Understanding Language Models:
Optimization, Architecture, and Emergent Abilities

by

Yingcong Li

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Electrical and Computer Engineering)
in the University of Michigan
2025

Doctoral Committee:

Professor Samet Oymak, Chair
Professor Wei Hu
Professor Liyue Shen
Professor Yixin Wang

Yingcong Li
yingcong@umich.edu
ORCID iD: 0009-0000-2667-6752

ACKNOWLEDGEMENTS

A PhD is a long journey that I began with very limited knowledge and a lot of uncertainty. It has not been easy, as I faced many challenges both academically and personally. However, along the way, I discovered deep academic interests, formed lasting friendships, and received invaluable support. I am sincerely grateful to everyone who has been part of this journey and who has brought me joy, guidance, and encouragement.

First and foremost, I would like to express my deepest gratitude to my advisor, Prof. Samet Oymak. Throughout my PhD, Dr. Oymak has provided unwavering support, clear guidance, and endless patience. Under his mentorship, I was able to develop my research interests, acquire essential skills, and grow as an independent thinker. He always believed in us and gave us the confidence to pursue ambitious problems. Beyond research, Dr. Oymak also cared deeply about our well-being, offering support through life's challenges, managing stress, and navigating future career choices. Thanks to his mentorship, I not only learned how to conduct research but also how to learn, how to stay resilient, and how to think about my long-term path. I am truly fortunate to have been his student.

I would also like to express my heartfelt gratitude to my parents and my younger sister. Being the only member of my family residing in the United States has not been easy, and without their constant support, encouragement, and companionship, I cannot imagine how much more lonely and difficult this journey would have been.

A special thanks goes to my two beloved cats, Lime and Amy, who have been with me for over four years. They have brought me more comfort, companionship, and emotional support than I ever expected. Though they occasionally test my patience, especially when waking me up at 5:00 AM or tapping the window at passing rabbits at midnight. Their presence has made my days warmer and my home more joyful.

I extend my sincere appreciation to my collaborators: Prof. Maryam Fazel, Dr. Ankit Singh Rawat, Prof. Dimitris Papailiopoulos, and Prof. Christos Thrampoulidis, for their insightful guidance and support throughout our projects. I am especially grateful to Prof. Thrampoulidis for generously sharing his personal experiences and thoughtful advice on academic careers and job searches.

I am also thankful to my committee members, Prof. Liyue Shen, Prof. Wei Hu, and Prof. Yixin Wang, for their time, support, and valuable feedback on my work. Their contributions have been essential in shaping my dissertation.

I would like to thank all of my colleagues and labmates, including (in alphabetical order): Xiangyu Chang, Alperen Gozeten, Emrullah Ildiz, Mingchen Li, Xiaofeng Liu, Yahya Sattar, Ege Onur Taga, and Xuechen Zhang, among many others. I am grateful to have met and worked alongside such a talented and supportive group of peers. In particular, I want to thank Davoud Ataee Tarzanagh for his time, effort, and patience during our close collaboration.

Finally, I would like to thank everyone else who has helped me along the way, even if your name does not appear above. Each act of kindness, each word of encouragement, and each moment of understanding has played a part in this journey.

Thank you all for being part of my PhD.

TABLE OF CONTENTS

LIST OF FIGURES

FIGURE

LIST OF APPENDICES

ABSTRACT

The remarkable success of large language models (LLMs) has led to significant advances across a wide range of tasks. However, their underlying mechanisms remain poorly understood, largely due to the complexity of their architectures (e.g., Transformers, Mamba) and the intricate ways in which predictions depend on data relationships. This thesis aims to uncover the fundamental principles behind the effectiveness of LLMs.

A central focus of this work is the optimization behavior of attention mechanisms, the core computational component of Transformer architectures. Unlike traditional neural networks, attention allows models to capture rich dependencies across sequences through token-to-token interactions. This thesis investigates the underlying mechanism of attention by analyzing its optimization dynamics. We show that optimized attention behaves similarly to a Support Vector Machine (SVM), effectively separating important tokens from less relevant ones using linear constraints on token-pair outer products. These selected tokens contribute most significantly to model performance. We further extend this analysis to next-token prediction, where we theoretically prove that a similar implicit bias holds.

While softmax attention has demonstrated strong empirical performance, its quadratic time and memory complexity limits its efficiency. To address this, recent architectures such as linear attention, state-space models, and gated linear attention have been proposed, achieving near-linear complexity per token via recurrent formulations. In addition to analyzing softmax attention, this thesis studies the optimization landscapes of these efficient alternatives in the context of in-context learning (ICL). We show that they implicitly perform variants of gradient descent over the in-context demonstrations, treating them as training data. We also investigate the role of model depth in leveraging unlabeled data. Our analysis reveals that while single-layer architectures fail to benefit from unlabeled in-context examples, multi-layer attention models can effectively exploit them, highlighting the importance of depth in semi-supervised in-context learning. Beyond architectural differences, this thesis explores optimization behavior across diverse problem settings, including retrieval-augmented generation (RAG), LoRA adaptation, and multitask prompting, providing insights that align more closely with real-world applications.

Finally, we examine the emergent abilities of LLMs through both theoretical and empirical

lenses. We formalize ICL as an algorithm learning problem, where the sequence model implicitly constructs a hypothesis function from the input prompt at inference time. We show numerically that sufficiently large, well-pretrained models can implement near-optimal algorithms. We also investigate chain-of-thought (CoT) reasoning, where models decompose complex tasks into simpler subproblems. We propose a two-stage interpretation of CoT: first, filtering and grouping relevant reasoning steps; second, performing in-context learning over each group. This framework explains the benefits of CoT reasoning in enhancing model expressivity and reducing in-context sample complexity.

Overall, this thesis aims to uncover the foundations of LLM effectiveness through the lens of optimization behavior, model architecture, and emergent capabilities.

CHAPTER 1

Introduction

1.1 Background

In recent decades, artificial intelligence (AI) has evolved rapidly, and is now widely used as an efficient tool to reduce human labor. However, the underlying reasons for its success remain underexplored, and significant efforts have been devoted to developing a deeper understanding of modern models.

Prior to the rise of large language models (LLMs), deep neural networks [10, 67, 75, 76, 97, 174, 183] had already achieved remarkable performance comparable to human abilities in domains such as computer vision (e.g., image classification, object detection), speech recognition, and machine translation. Beyond these applied successes, classical statistical learning theory and optimization theory have provided theoretical foundation for these achievements, grounding the study of neural networks within the broader framework of machine learning theory. For instance, studies have investigated data and compute efficiency by analyzing different data settings (e.g., coresets [9, 73]), model architectures (e.g., Mixture of Experts [115, 135]), and learning paradigms (e.g., transfer learning [15, 19, 191], continual learning [114, 120]). Additionally, research has focused on the optimization behavior of training deep models, examining phenomena such as benign overfitting [14, 25], the role of overparameterization [5, 48], and double-descent [17, 31, 139].

While traditional deep neural networks achieve strong performance on classical problems (e.g., regression, classification), they lack the capacity to effectively handle sequential data such as language, video, and time series. Unlike the conventional machine learning setting, where fixed-dimensional features are mapped to labels, sequence models map sequences of tokens to sequences of outputs. This greatly enhances their ability to model complex sequential data, and also brings up a variety of novel mathematical and algorithmic challenges:

- *The role of tokenization:* Tokenization is the crucial first step in processing raw sequential data, converting continuous or discrete signals into a format a model can un-

derstand. The choice of tokenization strategy regulates data representation, enabling models to process and understand sequential data more efficiently.

- *The role of different architectures:* To capture dependencies in sequential data, a variety of model architectures have been proposed. The attention mechanism has drawn the most focus and been most widely used, but other alternatives (e.g., linear attention [37, 92, 201], Mamba [40, 68] and gated linear attention [210]) also achieve competitive performance. Understanding how different architectures influence expressivity, efficiency, and generalization is crucial for advancing the effectiveness of sequence modeling.

- *The role of autoregression*: Since sequence models are often trained with a next-token prediction objective, they naturally acquire autoregressive capabilities. This structure enables novel learning paradigms beyond pretraining and fine-tuning, such as in-context learning, where models adapt to new tasks or domains through examples provided directly in the context window. Related techniques, such as prompt tuning, also enhance this adaptability by steering model behavior without modifying parameters directly.

In this thesis, I develop fundamental theories that address these problems, aiming to advance the foundations of language and sequence models.

The remainder of this chapter introduces the key background topics that motivate this work. We begin with an overview of the standard attention mechanism, where we focus on its implicit biases when applied to sequences of tokens. Next, we present efficient sequence-modeling architectures, including attention-based models as well as state-space models. We then discuss the emergence of in-context learning and the growing interest in chain-of-thought prompting. In the end, I provide a short summary of each chapters of the thesis. For a broader perspective, please refer to the conclusion in Chapter 11.

1.1.1 Attention Mechanism

Transformers have demonstrated exceptional performance across a wide array of domains, including natural language processing [22, 164], recommendation systems [34, 180, 221], and reinforcement learning [33, 85, 220]. At the heart of this success is the self-attention mechanism, first introduced by [193], which enables the model to capture global dependencies across input sequences. Unlike conventional models such as multilayer perceptrons (MLPs) or convolutional neural networks (CNNs), self-attention computes contextualized representations by modeling pairwise token interactions across the entire sequence [36, 121, 152, 153].

Formally, given an input sequence $\boldsymbol{X} \in \mathbb{R}^{T\times d}$ of length T with embedding dimension d, the self-attention mechanism is defined as:

$$\mathrm{att}(\boldsymbol{X}) := \mathbb{S}(\boldsymbol{X}\boldsymbol{W}_q\boldsymbol{W}_k^\top\boldsymbol{X}^\top)\boldsymbol{X}\boldsymbol{W}_v, \tag{1.1}$$

where $\boldsymbol{W}_q, \boldsymbol{W}_k, \boldsymbol{W}_v \in \mathbb{R}^{d\times d}$ are the trainable query, key, and value projection matrices. The softmax function $\mathbb{S}(\cdot)$ is applied row-wise to produce attention weights from the similarity scores in $\boldsymbol{X}\boldsymbol{W}_q\boldsymbol{W}_k^\top\boldsymbol{X}^\top$. The resulting weighted sum of values $\boldsymbol{X}\boldsymbol{W}_v$ yields the final output of the attention layer.

Despite their empirical success, the theoretical foundations of attention mechanisms are still being actively investigated. Prior research has examined several important aspects, including the Lipschitz properties of attention layers [94], the behavior of neural tangent kernels [79, 209], and the expressive power and Turing-completeness of Transformers [42, 47, 54, 62, 105, 213], along with statistical approximation guarantees [176, 203]. More recent work has shifted toward understanding the optimization and generalization dynamics of Transformers [58, 86, 107, 143, 144, 149, 189].

In this thesis, we contribute to this line of work by establishing a formal connection between the optimization geometry of self-attention and hard-margin support vector machines (SVMs). Specifically, we show that training a single-layer Transformer with gradient descent leads to an implicit separation of optimal versus non-optimal tokens, governed by linear constraints on token pair outer-products. This geometric perspective enables a precise characterization of the inductive bias induced by self-attention in shallow models.

1.1.2 Recurrent Alternatives

While Transformers have achieved widespread success, their standard (softmax) attention mechanism incurs quadratic time and memory complexity with sequence length, posing significant computational challenges for long-context applications. To address this limitation, a growing body of work has developed efficient alternatives that achieve near-linear complexity per token, largely by adopting recurrent formulations.

Early approaches include linear attention and state-space models (SSMs) (e.g. H3 [57]), both of which support $O(1)$ inference per token. These architectures exploit recurrent structures to achieve efficient sequence modeling.

Given an input sequence $\boldsymbol{X} \in \mathbb{R}^{T\times d}$ and projection matrices $\boldsymbol{W}_q, \boldsymbol{W}_k, \boldsymbol{W}_v \in \mathbb{R}^{d\times d}$ as in (1.1), linear attention is defined as:

$$\mathrm{att}_{\mathrm{lin}}(\boldsymbol{X}) := (\boldsymbol{X}\boldsymbol{W}_q\boldsymbol{W}_k^\top\boldsymbol{X}^\top)\boldsymbol{X}\boldsymbol{W}_v, \tag{1.2}$$

which replaces the softmax normalization in (1.1) with a linear operation, thus reducing the computational cost.

H3-like state-space models incorporate an implicit convolutional filter. A representative formulation is given by:

$$\mathrm{ssm}(\boldsymbol{X}) := (\boldsymbol{X}\boldsymbol{W}_q) \odot ((\boldsymbol{X}\boldsymbol{W}_k \odot \boldsymbol{X}\boldsymbol{W}_v) * \boldsymbol{f}), \tag{1.3}$$

where $\boldsymbol{f} \in \mathbb{R}^{n+1}$ is a 1D convolutional filter that mixes tokens across time. The element-wise (Hadamard) product $\odot$ serves as a gating mechanism [41] between query and key channels, and plays a critical role in enabling attention-like behavior.

Although these early models are significantly more efficient, they typically underperform compared to softmax-based attention in many real-world tasks. However, recent recurrent models such as RetNet [181], Mamba [40, 68], xLSTM [16], GLA Transformer [210], and RWKV [155] have closed this performance gap. As observed by [210], many of these models can be unified under the framework of Gated Linear Attention (GLA), which enhances linear attention with data-dependent gating within a recurrent formulation.

GLA is also closely related to implicit attention frameworks. For example, the Unified Implicit Attention model by [224] shows that Mamba [68] and RWKV [154] can be viewed as special cases under a shared gating-based attention principle. Furthermore, [225] demonstrate that gated cross-attention can improve multimodal fusion robustness, highlighting the broader potential of gated architectures for interpretability and stability.

The general form of GLA can be expressed as a recurrence. Let $\boldsymbol{X} = [\boldsymbol{x}_1\ \boldsymbol{x}_2\ \dots\ \boldsymbol{x}_T]^\top \in \mathbb{R}^{T\times d}$ be the input sequence, with associated query, key, and value embeddings defined as $\boldsymbol{q}_i = \boldsymbol{W}_q^\top \boldsymbol{x}_i$, $\boldsymbol{k}_i = \boldsymbol{W}_k^\top \boldsymbol{x}_i$, and $\boldsymbol{v}_i = \boldsymbol{W}_v^\top \boldsymbol{x}_i$ for $1 \le i \le T$. The GLA recurrence is:

$$\mathrm{gla}(\boldsymbol{X}) := [\boldsymbol{o}_1\ \boldsymbol{o}_2\ \dots\ \boldsymbol{o}_T]^\top, \text{ where } \quad \boldsymbol{o}_i = \boldsymbol{S}_i \boldsymbol{q}_i, \quad \boldsymbol{S}_i = \boldsymbol{G}_i \odot \boldsymbol{S}_{i-1} + \boldsymbol{v}_i \boldsymbol{k}_i^\top, \tag{1.4}$$

with $\boldsymbol{S}_0 = \boldsymbol{0}$ as the initial state. Here, $\boldsymbol{S}_i \in \mathbb{R}^{d\times d}$ is the 2D recurrent state, $\boldsymbol{o}_i \in \mathbb{R}^d$ is the output, and $\boldsymbol{G}_i = G(\boldsymbol{x}_i) \in \mathbb{R}^{d\times d}$ is the data-dependent gating matrix. When $\boldsymbol{G}_i$ is an all-one matrix, this recurrence reduces to causal linear attention [92].

Despite the recent empirical success of these recurrent architectures, their theoretical foundations remain limited. This thesis analyzes the optimization landscape of Linear Attention (cf. (1.2)), State-Space Models (cf. (1.3)), and Gated Linear Attention (cf. (1.4)) in the context of in-context learning (ICL). We show that these models implicitly perform variants of gradient descent over the input prompt, offering a unified perspective on their optimization behavior in ICL settings.

1.1.3 In-context Learning

Modern language models exhibit the remarkable ability to learn novel tasks or solve complex problems from the demonstrations provided within their context window [23, 61, 146, 190]. Such in-context learning (ICL) offers a novel and effective alternative to traditional fine-tuning techniques and has become an important feature of LLM with its applications spanning retrieval-augmented generation [106], and reasoning via advanced prompting techniques, such as chain-of-thought [205]. Specifically, ICL shows the ability of a language model to provide a prediction on an example given a few (input, output) examples within a prompt. For NLP, the examples may correspond to pairs of (question, answer)'s or translations.

There is growing interest in understanding the mechanisms behind ICL [23, 126, 165] due to its success in continuously enabling novel applications for LLMs [61, 146, 190]. In the previous work, [59] explored ICL ability of Transformers. In particular, they considered in-context prompts where each in-context example is labeled by a target function from a given function class, including linear models. A number of works have studied this and related settings to develop a theoretical understanding of ICL [197, 60, 38, 122, 117, 12, 4, 214, 50]. [4] focus on linear regression and provide a construction of Transformer weights that can enable a single step of GD based on in-context examples. Along the similar line, [196] provide a construction of weights in linear attention-only Transformers that can emulate GD steps on in-context examples for a linear regression task. Similar to this line of work, [39] argue that pre-trained language models act as meta-optimizer which utilize attention to apply meta-gradients to the original language model based on the in-context examples. [66] study the effectiveness of test-time training, theoretically and empirically demonstrating that a single gradient update at test time can enhance in-context performance.

While our work shares similarities with this line of research, we extend the theoretical understanding of ICL along several novel dimensions. These include the analysis of diverse model architectures (e.g., state-space models, gated linear attention) and non-isotropic data settings (e.g., retrieval-augmented generation, multitask prompting, semi-supervised learning). This broader perspective allows us to characterize the loss landscapes of different models and establish their connections to variants of online gradient descent algorithms.

1.1.4 Chain-of-thought Prompting and Reasoning Capabilities

Chain-of-thought (CoT) prompting has led to impressive improvements in complex reasoning tasks by decomposing them into step-by-step intermediate solutions [205, 140, 98, 204, 222, 145, 194, 99]. This generally showcases the ability of Transformers to model compositional functions. Several works have studied the advantages of this approach: [102, 53]

demonstrate that teaching arithmetic through intermediate steps allows small Transformers to learn more effectively; [108] show that student models can benefit from rationales generated by significantly larger teacher models. The broader idea of learning to compose skills has been explored in other settings as well [170, 124]. In parallel, the challenge of learning shallow neural networks with compositional structures has been shown to be theoretically hard [63, 64, 216], though recent work by [70] proves a formal learnability bound suggesting that compositional structure can enhance ICL. Although longer CoT traces have been shown to improve models' ability to solve challenging reasoning tasks, they are typically a feature of frontier models (e.g., OpenAI's o1 [84], DeepSeek-R1 [69], and s1 [136]). Reasoning with smaller language models remains difficult, as they often fail to complete key steps within the solution path. To address this limitation, [217, 218] combine supervised finetuning with reinforcement learning, enabling models to more effectively identify and solve critical but difficult steps along the reasoning trajectory. However, most prior studies focus on the empirical benefits and algorithmic design of CoT prompting, while relatively little is known about its underlying mechanisms.

From an expressivity standpoint, Transformers have been shown to be universal sequence-to-sequence approximators [213]. [62] provide an explicit construction demonstrating that even shallow Transformers can simulate general-purpose computation when their outputs are looped back into their inputs. More recently, [65] introduce continuous tokens that enhance model expressivity and inference efficiency via simultaneously tracking multiple reasoning traces in latent space. Other works have established the Turing-completeness of Transformer architectures, although these typically rely on infinite precision or recursion around attention layers [202, 161, 157, 125]. Closer to our line of work, [4, 196, 197] prove that constant-depth Transformers can implement gradient descent for solving linear regression. Similarly, [62] and [3] show that Transformers with looped inputs or linear attention can implement preconditioned gradient descent at convergence.

In this thesis, we build upon this foundation and propose a new interpretation of CoT reasoning as a two-step process: first, applying a filtering procedure to CoT prompts via a specific construction; and second, performing in-context learning over the filtered prompts.

1.2 Contributions and Thesis Outline

Chapter 2: Max-Margin Token Selection in Attention Mechanism

In this chapter, we explore the softmax-attention model $f(\boldsymbol{X}) = \langle \boldsymbol{X}\boldsymbol{v}, \texttt{softmax}(\boldsymbol{X}\boldsymbol{W}\boldsymbol{p})\rangle$, where $\boldsymbol{X}$ is the token sequence and $(\boldsymbol{v}, \boldsymbol{W}, \boldsymbol{p})$ are trainable parameters. We prove that running gradient descent on $\boldsymbol{p}$, or equivalently $\boldsymbol{W}$, converges in direction to a max-margin

solution that separates locally-optimal tokens from non-optimal ones. This clearly formalizes attention as an optimal token selection mechanism. Remarkably, our results are applicable to general data and precisely characterize optimality of tokens in terms of the value embeddings $\boldsymbol{X}\boldsymbol{v}$ and problem geometry. We also provide a broader regularization path analysis that establishes the margin maximizing nature of attention even for nonlinear prediction heads. When optimizing $\boldsymbol{v}$ and $\boldsymbol{p}$ simultaneously with logistic loss, we identify conditions under which the regularization paths directionally converge to their respective hard-margin SVM solutions where $\boldsymbol{v}$ separates the input features based on their labels. Interestingly, the SVM formulation of $\boldsymbol{p}$ is influenced by the support vector geometry of $\boldsymbol{v}$.

[187] Ataee Tarzanagh, Davoud, Yingcong Li, Xuechen Zhang, and Samet Oymak. "Max-margin token selection in attention mechanism." Advances in neural information processing systems 36 (2023): 48314-48362.

Chapter 3: Optimization of Projection Matrices: Softmax Attention as SVM

In this chapter, we study the self-attention mechanism where pairwise similarities are computed as $\texttt{softmax}(\boldsymbol{X}\boldsymbol{W}_q\boldsymbol{W}_k^\top\boldsymbol{X}^\top)$, with $(\boldsymbol{W}_q, \boldsymbol{W}_k)$ denoting the trainable query and key parameters. Similar to Chapter 2, we establish a formal equivalence between the optimization geometry of self-attention and a hard-margin SVM problem, which separates optimal input tokens from non-optimal ones using linear constraints on the outer products of token pairs. This formalism allows us to characterize the implicit bias of 1-layer transformers optimized with gradient descent, as follows. **(1)** Optimizing the attention layer, parameterized by $(\boldsymbol{W}_k, \boldsymbol{W}_Q)$, with vanishing regularization, converges in direction to an SVM solution minimizing the nuclear norm of the combined parameter $\boldsymbol{W} := \boldsymbol{W}_k\boldsymbol{W}_q^\top$. Instead, directly parameterizing by $\boldsymbol{W}$ minimizes a Frobenius norm SVM objective. We characterize this convergence, highlighting that it can occur in locally-optimal directions rather than global ones. **(2)** Complementing this, for $\boldsymbol{W}$-parameterization, we prove the local/global directional convergence of gradient descent under suitable geometric conditions. Importantly, we show that over-parameterization catalyzes global convergence by ensuring the feasibility of the SVM problem and by guaranteeing a benign optimization landscape devoid of stationary points. **(3)** While our theory applies primarily to linear prediction heads, we propose a more general SVM equivalence that predicts the implicit bias of 1-layer transformers with nonlinear heads/MLPs. Our findings apply to general datasets, trivially extend to cross-attention layer, and their practical validity is verified via thorough numerical experiments. We also introduce open problems and future research directions. We believe these findings inspire a new perspective, interpreting multilayer transformers as a hierarchy of SVMs that separates and selects optimal tokens.

[185] Tarzanagh, Davoud Ataee, Yingcong Li, Christos Thrampoulidis, and Samet Oymak. "Transformers as support vector machines." arXiv preprint arXiv:2308.16898 (2023).

Chapter 4: Convergence of Gradient Descent in Next-Token Prediction

In this chapter, we ask: *What does a single self-attention layer learn from next-token prediction?* We show that training self-attention with gradient descent learns an automaton which generates the next token in two distinct steps: (1) Hard retrieval: Given input sequence, self-attention precisely selects the high-priority input tokens associated with the last input token. (2) Soft composition: It then creates a convex combination of the high-priority tokens from which the next token can be sampled. Under suitable conditions, we rigorously characterize these mechanics through a directed graph over tokens extracted from the training data. We prove that gradient descent implicitly discovers the strongly-connected components (SCC) of this graph and self-attention learns to retrieve the tokens that belong to the highest-priority SCC available in the context window. Our theory relies on decomposing the model weights into a directional component and a finite component that correspond to hard retrieval and soft composition steps respectively. This also formalizes a related implicit bias formula conjectured in Chapters 2 and 3.

[112] Li, Yingcong, Yixiao Huang, Muhammed E. Ildiz, Ankit Singh Rawat, and Samet Oymak. "Mechanics of next token prediction with self-attention." In International Conference on Artificial Intelligence and Statistics, pp. 685-693. PMLR, 2024.

Chapter 5: Optimization Landscape of Linear Attention and State Space Models

In this chapter, we develop a stronger characterization of the optimization and generalization landscape of ICL through contributions on architectures, low-rank parameterization, and correlated designs: (1) We study the landscape of 1-layer linear attention and 1-layer H3, a state-space model. Under a suitable correlated design assumption, we prove that both implement 1-step preconditioned gradient descent. We show that thanks to its native convolution filters, H3 also has the advantage of implementing sample weighting and outperforming linear attention in suitable settings. (2) By studying correlated designs, we provide new risk bounds for retrieval augmented generation (RAG) and task-feature alignment which reveal how ICL sample complexity benefits from distributional alignment. (3) We derive the optimal risk for low-rank parameterized attention weights in terms of covariance spectrum. Through this, we also shed light on how LoRA can adapt to a new distribution by capturing the shift between task covariances. Experimental results corroborate our theoretical findings.

[116] Li, Yingcong, Ankit S. Rawat, and Samet Oymak. "Fine-grained analysis of in-context linear estimation: Data, architecture, and beyond." Advances in Neural Information Pro-

cessing Systems 37 (2024): 138324-138364.

Chapter 6: Optimization Landscape of Gated Linear Attention

In this chapter, we investigate the in-context learning capabilities of the GLA model and make the following contributions. We show that a multilayer GLA can implement a general class of Weighted Preconditioned Gradient Descent (WPGD) algorithms with data-dependent weights. These weights are induced by the gating mechanism and the input, enabling the model to control the contribution of individual tokens to prediction. To further understand the mechanics of this weighting, we introduce a novel data model with multitask prompts and characterize the optimization landscape of learning a WPGD algorithm. We identify mild conditions under which there exists a unique global minimum, up to scaling invariance, and the associated WPGD algorithm is unique as well. Finally, we translate these findings to explore the optimization landscape of GLA and shed light on how gating facilitates context-aware learning and when it is provably better than vanilla linear attention.

[118] Li, Yingcong, Davoud Ataee Tarzanagh, Ankit Singh Rawat, Maryam Fazel, and Samet Oymak. "Gating is weighting: Understanding gated linear attention through in-context learning." arXiv preprint arXiv:2504.04308 (2025).

Chapter 7: Task-specific Prompt Optimization in Linear Attention

In this chapter, we study in-context learning and consider a novel setting where the global task distribution can be partitioned into a union of conditional task distributions. We then examine the use of task-specific prompts and prediction heads for learning the prior information associated with the conditional task distribution using a one-layer attention model. Our results on loss landscape show that task-specific prompts facilitate a covariance-mean decoupling where prompt-tuning explains the conditional mean of the distribution whereas the variance is learned/explained through in-context learning. Incorporating task-specific head further aids this process by entirely decoupling estimation of mean and variance components. This covariance-mean perspective similarly explains how jointly training prompt and attention weights can provably help over fine-tuning after pretraining.

[30] Chang, Xiangyu, Yingcong Li, Muti Kara, Samet Oymak, and Amit K. Roy-Chowdhury. "Provable Benefits of Task-Specific Prompts for In-context Learning." arXiv preprint arXiv:2503.02102 (2025).

Chapter 8: Benefits of Unlabeled Data in Multilayer Linear Attention

In this chapter, we study in-context learning and examine a canonical setting where the demonstrations are drawn according to a binary Gaussian mixture model (GMM) and a

certain fraction of the demonstrations have missing labels. We provide a comprehensive theoretical study to show that: (1) The loss landscape of one-layer linear attention models recover the optimal fully-supervised estimator but completely fail to exploit unlabeled data; (2) In contrast, multilayer or looped transformers can effectively leverage unlabeled data by implicitly constructing estimators of the form $\sum_{i\geq 0} a_i(\boldsymbol{X}^\top \boldsymbol{X})^i \boldsymbol{X}^\top \boldsymbol{y}$ with $\boldsymbol{X}$ and $\boldsymbol{y}$ denoting features and partially-observed labels (with missing entries set to zero). We characterize the class of polynomials that can be expressed as a function of depth and draw connections to Expectation Maximization, an iterative pseudo-labeling algorithm commonly used in semi-supervised learning. Importantly, the leading polynomial power is exponential in depth, so mild amount of depth/looping suffices. As an application of theory, we propose looping off-the-shelf tabular foundation models to enhance their semi-supervision capabilities. Extensive evaluations on real-world datasets show that our method significantly improves the semisupervised tabular learning performance over the standard single pass inference.

[111] Li, Yingcong, Xiangyu Chang, Muti Kara, Xiaofeng Liu, Amit Roy-Chowdhury, and Samet Oymak. "When and How Unlabeled Data Provably Improve In-Context Learning." arXiv preprint arXiv:2506.15329 (2025).

Chapter 9: Generalization in In-context Learning

In this chapter, we formalize in-context learning as an algorithm learning problem where a transformer model implicitly constructs a hypothesis function from the input sequence at inference-time. We first explore the statistical aspects of this abstraction through the lens of multitask learning: We obtain generalization bounds for ICL when the input prompt is (1) a sequence of i.i.d. (input, label) pairs or (2) a trajectory arising from a dynamical system. The crux of our analysis is relating the excess risk to the stability of the algorithm implemented by the transformer, which holds under mild assumptions. Secondly, we use our abstraction to show that transformers can act as an adaptive learning algorithm and perform model selection across different hypothesis classes. We provide numerical evaluations that (1) demonstrate transformers can indeed implement near-optimal algorithms on classical regression problems with i.i.d. and dynamic data, (2) identify an inductive bias phenomenon where the transfer risk on unseen tasks is independent of the transformer complexity, and (3) empirically verify our theoretical predictions.

[113] Li, Yingcong, Muhammed Emrullah Ildiz, Dimitris Papailiopoulos, and Samet Oymak. "Transformers as algorithms: Generalization and stability in in-context learning." In International conference on machine learning, pp. 19565-19594. PMLR, 2023.

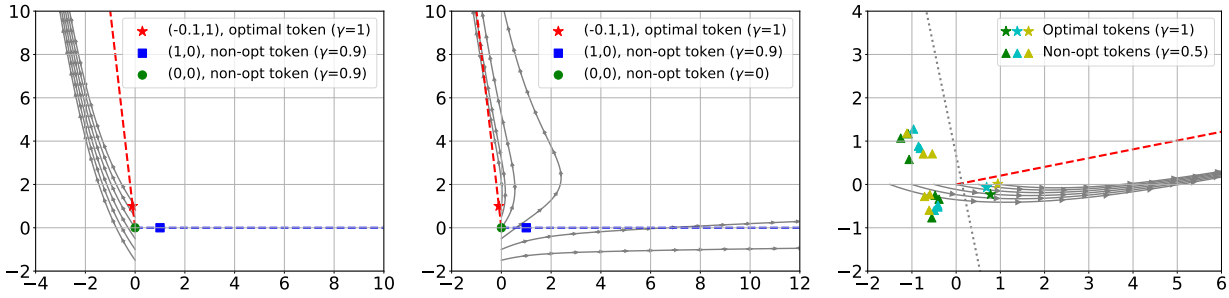
Chapter 10: Chain-of-thought and Its Emergent Abilities

This chapter investigates the impact of CoT on the ability of transformers to in-context learn a simple to study, yet general family of compositional functions: multi-layer perceptrons (MLPs). We find that the success of CoT can be attributed to breaking down in-context learning of a compositional function into two distinct phases: focusing on and filtering data related to each step of the composition and in-context learning the single-step composition function. Through both experimental and theoretical evidence, we demonstrate how CoT significantly reduces the sample complexity of ICL and facilitates the learning of complex functions that non-CoT methods struggle with.

[117] Li, Yingcong, Kartik Sreenivasan, Angeliki Giannou, Dimitris Papailiopoulos, and Samet Oymak. "Dissecting chain-of-thought: Compositionality through in-context filtering and learning." Advances in Neural Information Processing Systems 36 (2023): 22021-22046.

CHAPTER 2

Max-Margin Token Selection in Attention Mechanism



(a) Global convergence (b) Global and local convergence (c) Multiple inputs

Figure 2.1: The convergence behavior of the gradient descent on the attention weights $\boldsymbol{p}$ using the logistic loss in (ERM). The arrows (—>—) represent trajectories from different initializations. Here, (- - -) and (- - -) denote the **g**lobally- and **l**ocally-optimal **m**ax-**m**argin directions (`GMM`, `LMM`). γ denotes the *score* of a token per Definition 2.1. Discussion is provided under Theorems 2.2 and 2.3.

2.1 Introduction

The prominence of the attention mechanism motivates a fundamental theoretical understanding of its role in optimization and learning. While it is well-known that attention enables the model to focus on the relevant parts of the input sequence, the precise mechanism by which this is achieved is far from clear. To this end, we ask

Q: *What are the optimization behavior of the attention mechanism?*

In this chapter, we mainly focus on the optimization dynamics and inductive biases of prompt $\boldsymbol{p}$ and head $\boldsymbol{v}$. Specifically, we consider the fundamental attention model $f(\boldsymbol{X}) =$

$\langle \boldsymbol{X}\boldsymbol{v}, \mathbb{S}(\boldsymbol{X}\boldsymbol{W}^\top \boldsymbol{p})\rangle$. Here, $\boldsymbol{X}$ is the sequence of input tokens, $\boldsymbol{v}$ is the prediction head, $\boldsymbol{W}$ is the trainable key-query weights, and $\mathbb{S}$ denotes the softmax nonlinearity. For transformers, $\boldsymbol{p}$ corresponds to the `[CLS]` token or tunable prompt [104, 110, 149], whereas for RNN architectures [11], $\boldsymbol{p}$ corresponds to the hidden state.

Given training data $(\boldsymbol{X}_i, y_i)_{i=1}^n$ with labels $y_i \in \{-1, 1\}$ and inputs $\boldsymbol{X}_i \in \mathbb{R}^{T\times d}$, we consider the empirical risk minimization with a decreasing loss function $\ell(\cdot) : \mathbb{R} \to \mathbb{R}$,

$$\mathcal{L}(\boldsymbol{v}, \boldsymbol{p}, \boldsymbol{W}) = \frac{1}{n}\sum_{i=1}^{n} \ell(y_i \cdot f(\boldsymbol{X}_i)), \quad \text{where} \quad f(\boldsymbol{X}_i) = \mathbb{S}(\boldsymbol{p}^\top \boldsymbol{W}\boldsymbol{X}_i^\top)\boldsymbol{X}_i\boldsymbol{v}. \tag{2.1}$$

At a high-level, this work establishes fundamental equivalences between the optimization trajectories of (2.1) and hard-margin SVM problems. Our main contributions are as follows:

- ◇ **Optimization geometry of attention (§2.3):** We first show that gradient iterations of $\boldsymbol{p}$ and $\boldsymbol{W}$ admit a one-to-one mapping, thus we focus on optimizing $\boldsymbol{p}$ without losing generality. In Theorem 2.3, we prove that, under proper initialization:

  > Gradient descent on $\boldsymbol{p}$ converges in direction to a max-margin solution (see ($\boldsymbol{p}$-SVM)) that separates locally-optimal tokens from non-optimal ones.

  We call these **L**ocally-optimal **M**ax-**M**argin (`LMM`) directions and show that these thoroughly characterize the viable convergence directions of attention when the norm of its weights grows to infinity. We also identify conditions under which (algorithm-independent) regularization path and gradient descent path converge to **G**lobally-optimal **M**ax-**M**argin (`GMM`) direction in Theorems 2.1 and 2.2, respectively. A central feature of our results is precisely quantifying *optimality* in terms of *token scores* $\gamma_t = y \cdot \boldsymbol{v}^\top \boldsymbol{x}_t$ where $\boldsymbol{x}_t$ is the tth token of the input sequence $\boldsymbol{X}$. *Locally-optimal* tokens are those with higher scores than their nearest neighbors determined by the SVM solution. These are illustrated in Figure 2.1.

- ◇ **Optimize attention $\boldsymbol{p}$ and prediction-head $\boldsymbol{v}$ jointly (§2.4):** We study the joint problem under logistic loss function. We use regularization path analysis where (ERM) is solved under ridge constraints and we study the solution trajectory as the constraints are relaxed. Since the problem is linear in $\boldsymbol{v}$, if the attention features $\boldsymbol{x}_i^{\text{att}} = \boldsymbol{X}_i^\top \mathbb{S}(\boldsymbol{X}_i\boldsymbol{W}^\top \boldsymbol{p})$ are separable based on their labels y_i, $\boldsymbol{v}$ would implement a max-margin classifier. Building on this, we prove that $\boldsymbol{p}$ and $\boldsymbol{v}$ converges to their respective max-margin solutions under proper geometric conditions (Theorem 2.5). Relaxing these conditions, we obtain a more general solution where margin constraints

on $\boldsymbol{p}$ are relaxed on the inputs whose attention features are not support vectors of $\boldsymbol{v}$ (Theorem 2.6). Figure 2.4 illustrates these outcomes.

In Section 2.5, we extend our approach to the more general model $f(\boldsymbol{X}) = \psi(\boldsymbol{X}^\top \mathbb{S}(\boldsymbol{X}\boldsymbol{W}^\top \boldsymbol{p}))$ – where ψ is a nonlinear prediction head – to establish the margin maximizing nature of attention under broader conditions (Theorem 2.7). Overall, our results formalize the role of the attention mechanism as a token-selection & separation mechanism, characterize its geometry (when tokens are separable), and lay the groundwork for future research by connecting attention to a max-margin SVM formulation.

2.2 Preliminaries and Problem Setup

Notations. For any integer $N \geq 1$, let $[N] := \{1, \ldots, N\}$. We use lower-case and upper-case bold letters (e.g. $\boldsymbol{a}$ and $\boldsymbol{A}$) to represent vectors and matrices, respectively. The entries of $\boldsymbol{a}$ are denoted as $\boldsymbol{a}_i$. We use $\bar{\sigma}(\boldsymbol{A})$ to denote the maximum singular value of $\boldsymbol{A}$. We denote the minimum of two numbers a, b as $a \wedge b$, and the maximum as $a \vee b$. Big-O notation $\mathcal{O}(\cdot)$ hides the universal constants. Throughout, we will use $\mathcal{L}(\boldsymbol{p})$ and $\mathcal{L}(\boldsymbol{v}, \boldsymbol{p})$ to denote Objective (2.1) with fixed $(\boldsymbol{v}, \boldsymbol{W})$ and $\boldsymbol{W}$, respectively.

Optimization. Given an objective function $\mathcal{L} : \mathbb{R}^d \to \mathbb{R}$ and an ℓ_2-norm bound R, define the regularized solution as

$$\bar{\boldsymbol{p}}(R) := \arg \min_{\|\boldsymbol{p}\| \leq R} \mathcal{L}(\boldsymbol{p}). \tag{2.2}$$

Regularization path – the evolution of $\bar{\boldsymbol{p}}(R)$ as R grows – is known to capture the spirit of gradient descent as the ridge constraint R provides a proxy for the number of gradient descent iterations. For instance, [87, 169, 179] study the implicit bias of logistic regression and rigorously connect the directional convergence of regularization path (i.e. $\lim_{R\to\infty} \bar{\boldsymbol{p}}(R)/R$) and gradient descent. For gradient descent, we assume the objective $\mathcal{L}(\boldsymbol{p})$ is smooth and describe the gradient descent process as

$$\boldsymbol{p}(t+1) = \boldsymbol{p}(t) - \eta(t)\nabla\mathcal{L}(\boldsymbol{p}(t)), \tag{2.3}$$

where $\eta(t)$ is the stepsize at time t and $\nabla\mathcal{L}(\boldsymbol{p}(t))$ is the gradient of $\mathcal{L}$ at $\boldsymbol{p}(t)$.

Attention in Transformers. Next, we will discuss the connection between our model and the attention mechanism used in transformers. Our exposition borrows from [149], where

the authors analyze the same attention model using gradient-based techniques on specific contextual datasets.

- **Self-attention** is the core building block of transformers [193]. Given an input consisting of T tokens $\boldsymbol{X} = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_T]^\top \in \mathbb{R}^{T\times d}$, self-attention with key-query matrix $\boldsymbol{W} \in \mathbb{R}^{d\times d}$, and value matrix $\boldsymbol{W}_v \in \mathbb{R}^{d\times v}$, the self-attention model is defined as follows:

$$\mathrm{att}(\boldsymbol{X}) = \mathbb{S}(\boldsymbol{X}\boldsymbol{W}\boldsymbol{X}^\top)\boldsymbol{X}\boldsymbol{W}_v. \tag{2.4}$$

  Here, $\mathbb{S}(\cdot)$ is the softmax nonlinearity that applies row-wise on the similarity matrix $\boldsymbol{X}\boldsymbol{W}\boldsymbol{X}^\top$.

- **Tunable tokens: `[CLS]` and prompt-tuning.** In practice, we append additional tokens to the raw input features $\boldsymbol{X}$: For instance, a `[CLS]` token is used for classification purposes [44] and prompt vectors can be appended for adapting a pretrained model to new tasks [104, 110]. Let $\boldsymbol{p} \in \mathbb{R}^d$ be the tunable token (`[CLS]` or prompt vector) and concatenate it to $\boldsymbol{X}$ to obtain $\boldsymbol{X}_{\boldsymbol{p}} := [\boldsymbol{p}\ \boldsymbol{X}^\top]^\top \in \mathbb{R}^{(T+1)\times d}$. Consider the cross-attention features obtained from $\boldsymbol{X}_{\boldsymbol{p}}$ and $\boldsymbol{X}$ given by

$$\mathrm{att}(\boldsymbol{X}_{\boldsymbol{p}}) = \mathbb{S}(\boldsymbol{X}_{\boldsymbol{p}}\boldsymbol{W}\boldsymbol{X}^\top)\boldsymbol{X}\boldsymbol{W}_v = \begin{bmatrix}\mathbb{S}(\boldsymbol{p}^\top\boldsymbol{W}\boldsymbol{X}^\top)\\ \mathbb{S}(\boldsymbol{X}\boldsymbol{W}\boldsymbol{X}^\top)\end{bmatrix}\boldsymbol{X}\boldsymbol{W}_v.$$

The beauty of cross-attention is that it isolates the contribution of $\boldsymbol{p}$ under the upper term $\boldsymbol{V}^\top\boldsymbol{X}^\top\mathbb{S}(\boldsymbol{X}\boldsymbol{W}^\top\boldsymbol{p}) \in \mathbb{R}^v$. In this work, we use the value weights for classification, thus we set $v = 1$, and denote $\boldsymbol{v} = \boldsymbol{W}_v \in \mathbb{R}^d$. This brings us to our attention model of interest:

$$f(\boldsymbol{X}) = \langle \boldsymbol{X}\boldsymbol{v}, \mathbb{S}(\boldsymbol{K}\boldsymbol{p})\rangle, \quad \text{where} \quad \boldsymbol{K} = \boldsymbol{X}\boldsymbol{W}^\top. \tag{2.5}$$

Here, $(\boldsymbol{v}, \boldsymbol{W}, \boldsymbol{p})$ are the tunable model parameters and $\boldsymbol{K}$ is the key embeddings. Note that $\boldsymbol{W}$ and $\boldsymbol{p}$ are playing the same role within softmax, thus, it is intuitive that they exhibit similar optimization dynamics. Confirming this, the next lemma shows that gradient iterations of $\boldsymbol{p}$ (after setting $\boldsymbol{W} \leftarrow$ Identity) and $\boldsymbol{W}$ admit a one-to-one mapping.

Lemma 2.1 *Fix $\boldsymbol{u} \in \mathbb{R}^d \setminus \{\boldsymbol{0}\}$. Let $\psi : \mathbb{R}^d \to \mathbb{R}$ and $\ell : \mathbb{R} \to \mathbb{R}$ be differentiable functions. On the same training data $(y_i, \boldsymbol{X}_i)_{i=1}^n$, define $\tilde{\mathcal{L}}(\boldsymbol{p}) := 1/n \sum_{i=1}^n \ell(y_i \cdot \psi(\boldsymbol{X}_i^\top\mathbb{S}(\boldsymbol{X}_i\boldsymbol{p})))$ and $\mathcal{L}(\boldsymbol{W}) := 1/n \sum_{i=1}^n \ell(y_i \cdot \psi(\boldsymbol{X}_i^\top\mathbb{S}(\boldsymbol{X}_i\boldsymbol{W}^\top\boldsymbol{u})))$. Consider the gradient descent iterations on $\boldsymbol{p}$ and $\boldsymbol{W}$ with initial values $\boldsymbol{p}(0)$ and $\boldsymbol{W}(0) = \boldsymbol{u}\boldsymbol{p}(0)^\top/\|\boldsymbol{u}\|^2$ and stepsizes η and $\eta/\|\boldsymbol{u}\|^2$,*

respectively:

$$\boldsymbol{p}(t+1) = \boldsymbol{p}(t) - \eta \nabla \tilde{\mathcal{L}}(\boldsymbol{p}(t)),$$
$$\boldsymbol{W}(t+1) = \boldsymbol{W}(t) - \frac{\eta}{\|\boldsymbol{u}\|^2} \nabla \mathcal{L}(\boldsymbol{W}(t)).$$

We have that $\boldsymbol{W}(t) = \boldsymbol{u}\boldsymbol{p}(t)^\top / \|\boldsymbol{u}\|^2$ *for all* $t \geq 0$.

This lemma directly characterizes the optimization dynamics of $\boldsymbol{W}$ through the dynamics of $\boldsymbol{p}$, allowing us to reconstruct $\boldsymbol{W}$ from $\boldsymbol{p}$ using their gradient iterations. Therefore, we will fix $\boldsymbol{W}$ and concentrate on optimizing $\boldsymbol{p}$ in Section 2.3 and the joint optimization of $(\boldsymbol{v}, \boldsymbol{p})$ in Section 2.4.

Problem definition: Throughout, $(y_i, \boldsymbol{X}_i)_{i=1}^n$ denotes training dataset where $y_i \in \{-1, 1\}$ and $\boldsymbol{X}_i \in \mathbb{R}^{T \times d}$. We denote the key embeddings of $\boldsymbol{X}_i$ via $\boldsymbol{K}_i = \boldsymbol{X}_i \boldsymbol{W}^\top$ and explore the training risk

$$\mathcal{L}(\boldsymbol{v}, \boldsymbol{p}) = \frac{1}{n} \sum_{i=1}^{n} \ell\left(y_i \cdot \boldsymbol{v}^\top \boldsymbol{X}_i^\top \mathbb{S}(\boldsymbol{K}_i \boldsymbol{p})\right). \tag{ERM}$$

Importantly, our results apply to general tuples $(y_i, \boldsymbol{X}_i, \boldsymbol{K}_i)$ and do not assume that $(\boldsymbol{X}_i, \boldsymbol{K}_i)$ are tied via $\boldsymbol{W}$. Finally, the tth tokens of $\boldsymbol{X}_i, \boldsymbol{K}_i$ are denoted by $\boldsymbol{x}_{it}, \boldsymbol{k}_{it} \in \mathbb{R}^d$, respectively, for $t \in [T]$.

The highly nonlinear and nonconvex nature of the softmax operation makes the training problem in (ERM) a challenging nonconvex optimization problem for $\boldsymbol{p}$, even with a fixed $\boldsymbol{v}$. In the next section, we will introduce a set of assumptions to demonstrate the global and local convergence of gradient descent for margin maximization in the attention mechanism.

2.3 Margin Maximization with Attention

In this section, we present the main results (Theorems 2.2 and 2.3) by examining the implicit bias of gradient descent on learning $\boldsymbol{p} \in \mathbb{R}^d$ given a fixed choice of $\boldsymbol{v} \in \mathbb{R}^d$. Notably, our results apply to general *decreasing loss functions without requiring convexity*. This generality is attributed to margin maximization arising from the exponentially-tailed nature of softmax within attention, rather than ℓ. We maintain the following assumption on the loss function throughout this section.

Assumption 2.1 (Well-behaved Loss) *Over any bounded interval: (i) $\ell : \mathbb{R} \to \mathbb{R}$ is strictly decreasing. (ii) ℓ' is M_0-Lipschitz continuous and $|\ell'(u)| \leq M_1$.*

Assumption 2.1 includes many common loss functions, including the logistic loss $\ell(u) = \log(1 + e^{-u})$, exponential loss $\ell(u) = e^{-u}$, and correlation loss $\ell(u) = -u$. Assumption 2.1 implies that $\mathcal{L}(\boldsymbol{p})$ is L_p–smooth (see Lemma A.4 in Appendix), where

$$L_p := \frac{1}{n} \sum_{i=1}^{n} \left(M_0 \|\boldsymbol{v}\|^2 \|\boldsymbol{W}\|^2 \|\boldsymbol{X}_i\|^4 + 3M_1 \|\boldsymbol{v}\| \, \|\boldsymbol{W}\|^2 \|\boldsymbol{X}_i\|^3 \right). \tag{2.6}$$

We now introduce a convex hard-margin SVM problem that separates one token of the input sequence from the rest, jointly solved over all inputs. We will show that this problem captures the optimization properties of softmax-attention. Fix indices $\boldsymbol{\alpha} = (\alpha_i)_{i=1}^n$ and consider

$$\boldsymbol{p}^{\mathrm{mm}}(\boldsymbol{\alpha}) = \arg\min_{\boldsymbol{p}} \|\boldsymbol{p}\| \;\; \text{s.t.} \;\; \min_{t \neq \alpha_i} \boldsymbol{p}^\top (\boldsymbol{k}_{i\alpha_i} - \boldsymbol{k}_{it}) \geq 1, \;\; \text{for all} \;\; 1 \leq i \leq n. \qquad (\boldsymbol{p}\text{-SVM})$$

Note that existence of $\boldsymbol{p}^{\mathrm{mm}}(\boldsymbol{\alpha})$ implies the separability of tokens $\boldsymbol{\alpha}$ from the others. Specifically, choosing direction $\boldsymbol{p}^{\mathrm{mm}}(\boldsymbol{\alpha})$ will exactly select tokens $(\boldsymbol{x}_{i\alpha_i})_{i=1}^n$ at the attention output for each input sequence, that is, $\lim_{R\to\infty} \boldsymbol{X}_i^\top \mathbb{S}(R \cdot \boldsymbol{K}_i \boldsymbol{p}^{\mathrm{mm}}(\boldsymbol{\alpha})) = \boldsymbol{x}_{i\alpha_i}$. We are now ready to introduce our main results that characterize the global and local convergence of the attention weights $\boldsymbol{p}$ via ($\boldsymbol{p}$-SVM).

2.3.1 Global Convergence of Prompt $\boldsymbol{p}$

We first identify the conditions that guarantee the global convergence of gradient descent for $\boldsymbol{p}$. The intuition is that, in order for attention to exhibit implicit bias, the softmax nonlinearity should be forced to select the *optimal token within each input sequence*. Fortunately, the optimal tokens that achieve the smallest training objective under decreasing loss function $\ell(\cdot)$ have a clear definition.

Definition 2.1 (Token Scores, Optimality & `GMM`) *The score of token $\boldsymbol{x}_{it}$ of input $\boldsymbol{X}_i$ is defined as $\gamma_{it} := y_i \cdot \boldsymbol{v}^\top \boldsymbol{x}_{it}$. The optimal tokens for input $\boldsymbol{X}_i$ are those tokens with highest scores given by*

$$opt_i \in \arg\max_{t \in [T]} \gamma_{it}.$$

Globally-optimal max-margin (`GMM`) direction is defined as the solution of ($\boldsymbol{p}$-SVM) with optimal indices $(opt_i)_{i=1}^n$ by $\boldsymbol{p}^{mm\star}$.

It is worth noting that score definition simply uses the *value embeddings* $\boldsymbol{v}^\top \boldsymbol{x}_{it}$ of the tokens. Note that multiple tokens within an input might attain the same score, thus opt_i or $\boldsymbol{p}^{\text{mm}\star}$ may not be unique. The theorem below provides our regularization path guarantee on the global convergence of attention.

Theorem 2.1 (Regularization Path) *Suppose Assumption 2.1 on the loss function holds, and for all $i \in [n]$ and $t \neq opt_i$, the scores obey $\gamma_{it} < \gamma_{iopt_i}$. Then, the regularization path $\bar{\boldsymbol{p}}(R) = \arg\min_{\|\boldsymbol{p}\| \leq R} \mathcal{L}(\boldsymbol{p})$ converges to the* GMM *direction i.e. $\lim_{R\to\infty} \bar{\boldsymbol{p}}(R)/R = \boldsymbol{p}^{mm\star}/\|\boldsymbol{p}^{mm\star}\|$.*

Theorem 2.1 shows that as the regularization strength R increases towards the ridgeless problem $\min_{\boldsymbol{p}} \mathcal{L}(\boldsymbol{p})$, the optimal direction $\bar{\boldsymbol{p}}(R)$ aligns more closely with the max-margin solution $\boldsymbol{p}^{\text{mm}\star}$. Since this theorem allows for arbitrary token scores, it demonstrates that *max-margin token separation is an essential feature of the attention mechanism*. In fact, it is a corollary of Theorem 2.7, which applies to the generalized model $f(\boldsymbol{X}) = \psi(\boldsymbol{X}^\top \mathbb{S}(\boldsymbol{X}\boldsymbol{W}^\top \boldsymbol{p}))$ and accommodates multiple optimal tokens per input. However, while regularization path analysis captures the global behavior, gradient descent lacks general global convergence guarantees. In Section 2.3.2, we show that due to the nonconvex landscape and softmax nonlinearity, gradient descent often converges to local optima. We first establish that when (ERM) is trained with gradient descent, the norm of the parameters will diverge. For the restrictive setting of $n = 1$, gradient descent also exhibits a global convergence guarantee.

Assumption 2.2 *For all $i \in [n]$ and $t, \tau \neq opt_i$, the scores per Definition 2.1 obey $\gamma_{it} = \gamma_{i\tau} < \gamma_{iopt_i}$.*

Theorem 2.2 (Global Convergence of Gradient Descent) *Suppose Assumption 2.1 on the loss function ℓ and Assumption 2.2 on the tokens' score hold. Then, the gradient descent iterates $\boldsymbol{p}(t+1) = \boldsymbol{p}(t) - \eta \nabla \mathcal{L}(\boldsymbol{p}(t))$ on* (ERM)*, with the stepsize $\eta \leq 1/L_p$ and any starting point $\boldsymbol{p}(0)$ satisfy $\lim_{t\to\infty} \|\boldsymbol{p}(t)\| = \infty$. If $n = 1$, we also have $\lim_{t\to\infty} \boldsymbol{p}(t)/\|\boldsymbol{p}(t)\| = \boldsymbol{p}^{mm\star}/\|\boldsymbol{p}^{mm\star}\|$.*

Theorem 2.2 shows that gradient descent will diverge in norm, and when $n = 1$, the normalized predictor $\boldsymbol{p}(t)/\|\boldsymbol{p}(t)\|$ converges towards $\boldsymbol{p}^{\text{mm}\star}$, the separator of the globally optimal token. While $n = 1$ is a stringent condition, this requirement is in fact tight as discussed in Appendix A.5. To illustrate this theorem, we have conducted synthetic experiments. Let us first explain the setup used in Figure 2.1. We set $d = 3$ as the dimension, with each token having three entries $\boldsymbol{x} = [x_1, x_2, x_3]$. We reserve the first two coordinates as key embeddings $\boldsymbol{k} = [x_1, x_2, 0]$ by setting $\boldsymbol{W} = \text{diag}([1, 1, 0])$. This is what we display in our figures as token

positions. Finally, in order to assign scores to the tokens we use the last coordinate by setting $\boldsymbol{v} = [0, 0, 1]$. This way score becomes $y \cdot \boldsymbol{v}^\top \boldsymbol{x} = y \cdot x_3$, allowing us to assign any score (regardless of key embedding).

In Figure 2.1a, the gray paths represent gradient descent trajectories from different initializations. The points $(0, 0)$ and $(1, 0)$ correspond to non-optimal tokens, while the point $(-0.1, 1)$ represents the optimal token. Notably, gradient descent iterates with various starting points converge towards the direction of the max-margin solution $\boldsymbol{p}^{\text{mm}\star}$ (depicted by - - -). Moreover, as the iteration count t increases, the inner product $\langle \boldsymbol{p}(t)/\|\boldsymbol{p}(t)\|, \boldsymbol{p}^{\text{mm}\star}/\|\boldsymbol{p}^{\text{mm}\star}\|\rangle$ consistently increases. Figure 2.1c also depicts the directional convergence of gradient descent from various initializations on multiple inputs, with the gray dotted line representing the separating hyperplane. These emphasize the gradual alignment between the evolving predictor and the max-margin solution throughout the optimization.

`GMM` $\boldsymbol{p}^{\text{mm}\star}$ is guaranteed to exist when labels are 1: Our results focus on the directional convergence of attention, which corresponds to the scenarios ($\boldsymbol{p}$-SVM) is feasible. As demonstrated in Figure 2.3a numerically, we expect separability to hold if the problem has more parameters (large d) whereas for small d, `GMM` or `LMM` directions may not exist and gradient descent can instead find a finite minima to $\mathcal{L}(\boldsymbol{p})$. While we defer the statistical analysis of separability to future, we make the remark that if labels are always 1, the problem is guaranteed to be separable because value weights $\boldsymbol{v}$ provides the separating direction. This scenario can be interpreted as an *exact alignment* between values and

Lemma 2.2 *Suppose for all $i \in [n]$ and $t \neq opt_i$, $y_i = 1$ and $\gamma_{it} < \gamma_{iopt_i}$. Also assume $\boldsymbol{W} \in \mathbb{R}^{d\times d}$ is full-rank. Then $\boldsymbol{p}^{mm\star}$ exists – i.e. ($\boldsymbol{p}$-SVM) is feasible for optimal indices $\alpha_i \leftarrow opt_i$.*

2.3.2 Local convergence of the attention weights p

Theorem 2.2 on the global convergence of gradient descent serves as a prelude to the general behavior of the optimization. Once we relax Assumption 2.2 by allowing for arbitrary token scores, we will show that $\boldsymbol{p}$ can converge (in direction) to a locally-optimal solution. However, this locally-optimal solution is still characterized in terms of ($\boldsymbol{p}$-SVM) which separates *locally-optimal* tokens from the rest. Our theory builds on two new concepts: locally-optimal tokens and neighbors of these tokens.

Definition 2.2 (SVM-Neighbor and Locally-Optimal Tokens) *Fix token indices $\boldsymbol{\alpha} = (\alpha_i)_{i=1}^n$ for which ($\boldsymbol{p}$-SVM) is feasible to obtain $\boldsymbol{p}^{mm} = \boldsymbol{p}^{mm}(\boldsymbol{\alpha})$. Consider tokens*

$\mathcal{T}_i \subset [T]$ *such that* $(\boldsymbol{k}_{i\alpha_i} - \boldsymbol{k}_{it})^\top \boldsymbol{p}^{mm} = 1$ *for all* $t \in \mathcal{T}_i$. *We refer to* $\mathcal{T}_i$ *as support indices of* $\boldsymbol{k}_{i\alpha_i}$. *Additionally, tokens with indices* $\boldsymbol{\alpha} = (\alpha_i)_{i=1}^n$ *are called locally-optimal if for all* $i \in [n]$ *and* $t \in \mathcal{T}_i$ *scores per Definition 2.1 obey* $\gamma_{i\alpha_i} > \gamma_{it}$. *Associated* $\boldsymbol{p}^{mm}$ *is called a locally-optimal max-margin (*`LMM`*) direction.*

To provide a basis for discussing local convergence, we provide some preliminary definitions regarding cones. For a given $\boldsymbol{q}$ and a scalar $\mu > 0$, we define $\texttt{cone}_\mu(\boldsymbol{q})$ as the set of vectors $\boldsymbol{p} \in \mathbb{R}^d$ such that the correlation coefficient between $\boldsymbol{p}$ and $\boldsymbol{q}$ is at least $1 - \mu$:

$$\texttt{cone}_\mu(\boldsymbol{q}) := \left\{ \boldsymbol{p} \in \mathbb{R}^d \;\middle|\; \left\langle \frac{\boldsymbol{p}}{\|\boldsymbol{p}\|}, \frac{\boldsymbol{q}}{\|\boldsymbol{q}\|} \right\rangle \geq 1 - \mu \right\}. \quad (2.7)$$

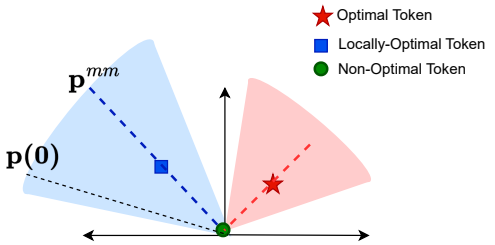


Figure 2.2: Gradient descent initialization $\boldsymbol{p}(0)$ inside the cone containing the locally-optimal solution $\boldsymbol{p}^{\mathrm{mm}}$.

Given $R > 0$, the intersection of $\texttt{cone}_\mu(\boldsymbol{q})$ and the set $\{\boldsymbol{p} \in \mathbb{R}^d | \; \|\boldsymbol{p}\| \geq R\}$ is denoted as $\mathcal{C}_{\mu,R}(\boldsymbol{q})$:

$$\mathcal{C}_{\mu,R}(\boldsymbol{q}) := \texttt{cone}_\mu(\boldsymbol{q}) \cap \left\{ \boldsymbol{p} \in \mathbb{R}^d \;\middle|\; \|\boldsymbol{p}\| \geq R \right\}. \quad (2.8)$$

Next, we demonstrate the existence of parameters $\mu = \mu(\boldsymbol{\alpha}) > 0$ and $R > 0$ such that when R is sufficiently large, there are no stationary points within $\mathcal{C}_{\mu,R}(\boldsymbol{p}^{\mathrm{mm}})$. Further, the gradient descent initialized within $\mathcal{C}_{\mu,R}(\boldsymbol{p}^{\mathrm{mm}})$ converges in direction to $\boldsymbol{p}^{\mathrm{mm}}/\|\boldsymbol{p}^{\mathrm{mm}}\|$; refer to Figure 2.2 for a visualization.

Theorem 2.3 (Local Convergence of Gradient Descent) *Suppose Assumption 2.1 on the loss function* ℓ *holds and assume* $\boldsymbol{\alpha} = (\alpha_i)_{i=1}^n$ *are indices of locally-optimal tokens per Definition 2.2. Then, there is a constant* $\mu = \mu(\boldsymbol{\alpha}) \in (0, 1)$ *and* $R > 0$ *such that* $\mathcal{C}_{\mu,R}(\boldsymbol{p}^{mm})$ *does not contain any stationary points. Further, for any starting point* $\boldsymbol{p}(0) \in \mathcal{C}_{\mu,R}(\boldsymbol{p}^{mm})$, *gradient descent iterates* $\boldsymbol{p}(t+1) = \boldsymbol{p}(t) - \eta \nabla \mathcal{L}(\boldsymbol{p}(t))$ *on* (ERM) *with stepsize* $\eta \leq 1/L_p$ *satisfies* $\lim_{t\to\infty} \|\boldsymbol{p}(t)\| = \infty$ *and* $\lim_{t\to\infty} \boldsymbol{p}(t)/\|\boldsymbol{p}(t)\| = \boldsymbol{p}^{mm}/\|\boldsymbol{p}^{mm}\|$.

To further illustrate Theorem 2.3, we can consider Figure 2.1b where $n = 1$ and $T = 3$. In this figure, the point $(0, 0)$ represents the non-optimal tokens, while $(1, 0)$ represents the locally optimal token. Additionally, the gray paths represent the trajectories of gradient descent initiated from different points. By observing the figure, we can see that gradient descent, when properly initialized, converges towards the direction of $\boldsymbol{p}^{\mathrm{mm}}$ (depicted by - - -). This direction of convergence effectively separates the locally optimal tokens $(1, 0)$ from the non-optimal token $(0, 0)$.

2.3.3 Regularization paths can only converge to locally-optimal max-margin directions

An important question arises regarding whether our definition of LMM (Definition 2.2) encompasses all possible convergence paths of the attention mechanism when $\|\boldsymbol{p}\| \to \infty$. To address this, we introduce the set of LMM directions as follows:

$$\mathcal{P}^{\mathrm{mm}} := \left\{ \frac{\boldsymbol{p}^{\mathrm{mm}}(\boldsymbol{\alpha})}{\|\boldsymbol{p}^{\mathrm{mm}}(\boldsymbol{\alpha})\|} \;\middle|\; \boldsymbol{\alpha} \text{ is locally-optimal per Definition 2.2} \right\}.$$

The following theorem establishes the tightness of these directions: It demonstrates that for any candidate $\boldsymbol{q} \notin \mathcal{P}^{\mathrm{mm}}$, its local regularization path within an arbitrarily small neighborhood will provably not converge in the direction of $\boldsymbol{q}$.

Theorem 2.4 *Fix $\boldsymbol{q} \notin \mathcal{P}^{mm}$ with unit ℓ_2 norm. Assume that token scores are distinct (namely $\gamma_{it} \neq \gamma_{i\tau}$ for $t \neq \tau$) and key embeddings $\boldsymbol{k}_{it}$ are in general position (see Theorem A.1). Fix arbitrary $\epsilon > 0, R_0 > 0$. Define the local regularization path of $\boldsymbol{q}$ as its (ϵ, R_0)-conic neighborhood:*

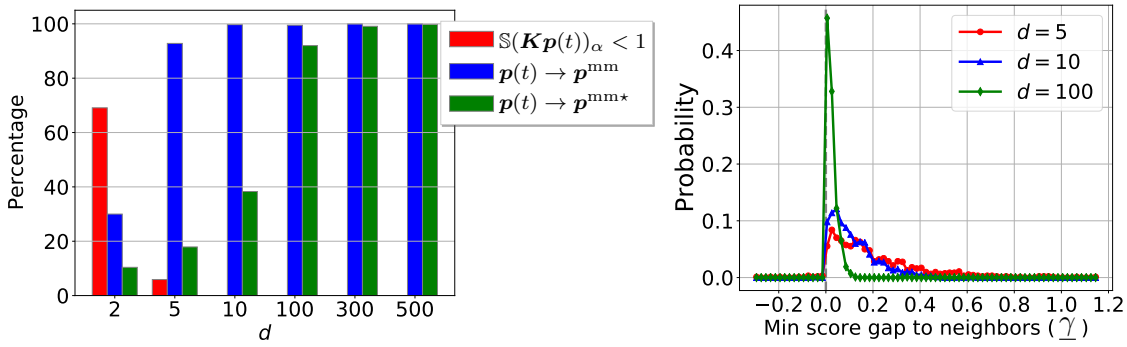
$$\bar{\boldsymbol{p}}(R) = \underset{\boldsymbol{p} \in \mathcal{C}_{\epsilon,R_0}(\boldsymbol{q}), \|\boldsymbol{p}\| \leq R}{\arg\min} \mathcal{L}(\boldsymbol{p}), \tag{2.9}$$
$$\text{where} \quad \mathcal{C}_{\epsilon,R_0}(\boldsymbol{q}) = \texttt{cone}_{\epsilon}(\boldsymbol{q}) \cap \left\{ \boldsymbol{p} \in \mathbb{R}^d \middle| \; \|\boldsymbol{p}\| \geq R_0 \right\}.$$

Then, either $\lim_{R\to\infty} \|\bar{\boldsymbol{p}}(R)\| < \infty$ or $\lim_{R\to\infty} \bar{\boldsymbol{p}}(R)/\|\bar{\boldsymbol{p}}(R)\| \neq \boldsymbol{q}$. In both scenarios $\lim_{R\to\infty} \bar{\boldsymbol{p}}(R)/R \neq \boldsymbol{q}$.

The result above nicely complements Theorem 2.3, which states that when gradient descent is initialized above a threshold ($\|\boldsymbol{p}(0)\| \geq R_0$) in an LMM direction, $\|\boldsymbol{p}(t)\|$ diverges but the direction converges to LMM. In contrast, Theorem 2.4 shows that regardless of how small the cone is (in terms of angle and norm lower bound $\|\boldsymbol{p}\| \geq R_0$), the optimal solution path will not converge along $\boldsymbol{q} \notin \mathcal{P}^{\mathrm{mm}}$.

To illustrate Theorems 2.3 & 2.4, we have investigated the convergence behavior of $\boldsymbol{p}(t)$ generated by gradient descent in Figure 2.3a, using $n = 4$, $T = 6$, and conducted 1,000 random trials for varying $d \in \{2, 5, 10, 100, 300, 500\}$. These experiments use normalized gradient descent with learning rate 1 for 1000 iterations. Inputs $\boldsymbol{x}_{it}$ and the linear head $\boldsymbol{v}$ are uniformly sampled from the unit sphere, while y_i is uniformly ± 1, and $\boldsymbol{W}$ is set to $\boldsymbol{I}$.

The bar plot in Figure 2.3a distinguishes between *non-saturated* softmax (red bars) and *saturated* softmax (other bars). Saturation is defined as average softmax probability over tokens selected by gradient descent are at least $1-10^{-5}$ and implies that attention selects one



(a) Perc. of different convergence scenarios

(b) Prob. of $\underline{\gamma} := \min_{i\in[n],t\in\mathcal{T}_i} \gamma_{i\alpha_i} - \gamma_{it}$

Figure 2.3: Convergence analysis of $\boldsymbol{p}(t)$ trained with random data using gradient descent. **(a)** shows three scenarios: (1) attention failing to select one token per input (i.e. softmax is not saturated); (2) $\boldsymbol{p}$ converging towards $\boldsymbol{p}^{\mathrm{mm}}$; and (3) $\boldsymbol{p}^{\mathrm{mm}}$ equating to $\boldsymbol{p}^{\mathrm{mm}\star}$ with red, blue, and green bars, respectively. Considering saturated softmax instances where $\boldsymbol{p}(t)$ selects one token α_i per-input, **(b)** presents histogram of the minimal score gap between α_i and its corresponding neighbors $\mathcal{T}_i$.

token per input. Note that, whenever the norm of gradient descent solution is finite, softmax will be non-saturated. For small d (e.g., $d = 2$), problem has small degrees of freedom to separate optimal tokens from the rest (i.e. no SVM solution for `LMM` directions) – especially due to label randomness. This results in a tall red bar capturing the finite-norm solutions. However, for larger d, we observe that softmax saturates (i.e. $\|\boldsymbol{p}(t)\| \to \infty$) and we observe that the selected tokens $\boldsymbol{\alpha}$ almost always converges to an `LMM` direction (blue bar) – this is in line with Theorems 2.3 & 2.4. We also study the convergence to the globally-optimal `GMM` which is represented by the green bar: `GMM` is a strict subset of `LMM` however as d increases, we observe that the probability of `GMM` convergence increases as well. This behavior is in line with what one would expect from over-parameterized deep learning theory [5, 48, 83, 151] and motivates future research. The average correlation coefficient between $\boldsymbol{p}(t)$ and its associated `LMM`/`GMM` direction is 0.997, suggesting that, whenever softmax saturates, gradient descent indeed directionally converges to a `LMM` solution $\boldsymbol{p} \in \mathcal{P}^{\mathrm{mm}}$, confirming Theorem 2.4.

Furthermore, we found that there exist problem instances, with saturated softmax and $\|\boldsymbol{p}(t)\| \to \infty$, that do not converge to either `LMM` or `GMM`. We analyzed this phenomenon using the minimum score gap, $\underline{\gamma} := \min_{i\in[n],t\in\mathcal{T}_i} \gamma_{i\alpha_i} - \gamma_{it}$, where $\mathcal{T}_i, i \in [n]$, represents the sets of support index tokens. Figure 2.3b provides the probability distribution of $\underline{\gamma}$ (with bins of width < 0.01) and demonstrates the rarity of such cases. Specifically, we found this happens less than 1% of the problems, that is, $\mathrm{Prob}(\underline{\gamma} < 0) < 0.01$. Figure 2.3b also reveals that, in these scenarios, even if $\underline{\gamma} < 0$, it is typically close to zero i.e. even if there exists a

support index with a higher score, it is only slightly so. This is not surprising since when token scores are close, we need a large number of gradient iterations to distinguish them. For all practical purposes, the optimization will treat both tokens equally and rather than solving ($\boldsymbol{p}$-SVM), the more refined formulation ($\boldsymbol{p}$-SVM') developed in Section 2.5 will be a better proxy. Confirming this intuition, we have verified that, over the instances $\underline{\gamma} < 0$, gradient descent solution is still > 0.99 correlated with the max-margin solution in average. Additional details are provided in the appendix.

2.4 Joint Convergence of Head $\boldsymbol{v}$ and Prompt $\boldsymbol{p}$

In this section, we extend the preceding results to the general case of joint optimization of head $\boldsymbol{v}$ and attention weights $\boldsymbol{p}$ using a logistic loss function. To this aim, we focus on regularization path analysis, which involves solving (ERM) under ridge constraints and examining the solution trajectory as the constraints are relaxed.

High-level intuition. Since the prediction is linear as a function of $\boldsymbol{v}$, logistic regression in $\boldsymbol{v}$ can exhibit its own implicit bias to a max-margin solution. Concretely, define the attention features $\boldsymbol{x}_i^{\boldsymbol{p}} = \boldsymbol{X}_i^\top \mathbb{S}(\boldsymbol{K}_i \boldsymbol{p})$ and define the dataset $\mathcal{D}^{\boldsymbol{p}} = (y_i, \boldsymbol{x}_i^{\boldsymbol{p}})_{i=1}^n$. If this dataset $\mathcal{D}^{\boldsymbol{p}}$ is linearly separable, then fixing $\boldsymbol{p}$ and optimizing only $\boldsymbol{v}$ will converge in the direction of the standard max-margin classifier

$$\boldsymbol{v}^{\mathrm{mm}} = \arg\min_{\boldsymbol{v} \in \mathbb{R}^d} \|\boldsymbol{v}\| \quad \text{subject to} \quad y_i \cdot \boldsymbol{v}^\top \boldsymbol{r}_i \geq 1, \quad \text{for all} \quad 1 \leq i \leq n, \qquad (\boldsymbol{v}\text{-SVM})$$

after setting inputs to the attention features $\boldsymbol{r}_i \leftarrow \boldsymbol{x}_i^{\boldsymbol{p}}$ [177]. This motivates a clear question:

Under what conditions, optimizing $\boldsymbol{v}, \boldsymbol{p}$ jointly will converge to their respective max-margin solutions?

We study this question in two steps. Loosely speaking: **(1)** We will first assume that, at the optimal tokens $\boldsymbol{x}_{i\alpha_i}, i \in [n]$ selected by $\boldsymbol{p}$, when solving ($\boldsymbol{v}$-SVM) with $\boldsymbol{r}_i \leftarrow \boldsymbol{x}_{i\alpha_i}$, all of these tokens become support vectors of ($\boldsymbol{v}$-SVM). **(2)** We will then relax this condition to uncover a more general implicit bias for $\boldsymbol{p}$ that distinguish support vs non-support vectors. Throughout, we assume that the joint problem is separable and there exists $(\boldsymbol{v}, \boldsymbol{p})$ asymptotically achieving zero training loss.

2.4.1 When all attention features are support vectors

In ($\boldsymbol{v}$-SVM), define *label margin* to be $1/\|\boldsymbol{v}^{\mathrm{mm}}\|$. Our first insight in quantifying the joint implicit bias is that, **optimal tokens** admit a natural definition: *Those that maximize the downstream label margin when selected.* This is formalized below where we assume that: (1) Selecting the token indices $\boldsymbol{\alpha} = (\alpha_i)_{i=1}^n$ from each input data achieves the largest label margin. (2) The optimality of the $\boldsymbol{\alpha}$ choice is strict in the sense that mixing other tokens will shrink the label margin in ($\boldsymbol{v}$-SVM).

Assumption 2.3 (Optimal Tokens) *Let $\Gamma > 0$ be the label margin when solving ($\boldsymbol{v}$-SVM) with $\boldsymbol{r}_i \leftarrow \boldsymbol{x}_{i\alpha_i}$. There exists $\nu > 0$ such that for all $\boldsymbol{p}$, solving ($\boldsymbol{v}$-SVM) with $\boldsymbol{r}_i \leftarrow \boldsymbol{x}_i^{\boldsymbol{p}}$ results in a label margin of at most $\Gamma - \nu \cdot \max_{i\in[n]}(1 - \boldsymbol{s}_{i\alpha_i})$ where $\boldsymbol{s}_i = \mathbb{S}(\boldsymbol{K}_i\boldsymbol{p})$.*

Example: To gain intuition, let us fix $\boldsymbol{a} \in \mathbb{R}^d$ and consider the dataset obeying $\boldsymbol{x}_{i1} = y_i \cdot \boldsymbol{a}$ and $\|\boldsymbol{x}_{it}\| < \|\boldsymbol{a}\|$ for all $t \geq 2$ and all $i \in [n]$. For this dataset, we can choose $\alpha_i = 1$, $\boldsymbol{v}^{\mathrm{mm}} = \boldsymbol{a}/\|\boldsymbol{a}\|^2$, $\Gamma = 1/\|\boldsymbol{v}^{\mathrm{mm}}\| = \|\boldsymbol{a}\|$ and $\nu = \|\boldsymbol{a}\| - \sup_{i\in[n],t\geq 2}\|\boldsymbol{x}_{it}\|$.

Theorem 2.5 *Consider the ridge-constrained solutions $(\boldsymbol{v}_r, \boldsymbol{p}_R)$ of (ERM) defined as*

$$(\boldsymbol{v}_r, \boldsymbol{p}_R) := \arg\min_{(\boldsymbol{v},\boldsymbol{p})} \mathcal{L}(\boldsymbol{v}, \boldsymbol{p}) \quad \text{s.t.} \quad \begin{aligned} &\|\boldsymbol{p}\| \leq R, \\ &\|\boldsymbol{v}\| \leq r := r(R). \end{aligned} \tag{2.10}$$

Suppose there are token indices $\boldsymbol{\alpha} = (\alpha_i)_{i=1}^n$ for which $\|\boldsymbol{p}^{mm}(\boldsymbol{\alpha})\|$ exists ($\boldsymbol{p}$-SVM is feasible) and Assumption 2.3 holds for some $\Gamma, \nu > 0$. Then, when $r(R) = \exp(\mathcal{O}(R))$, as $R \to \infty$:

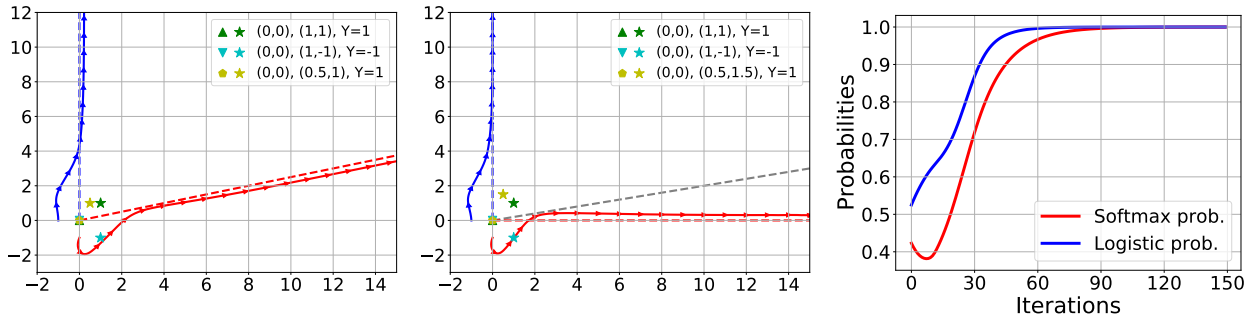
$$\lim_{R\to\infty} \frac{\boldsymbol{p}_R}{R} = \frac{\boldsymbol{p}^{mm}}{\|\boldsymbol{p}^{mm}\|}, \qquad \lim_{R\to\infty} \frac{\boldsymbol{v}_r}{r} = \frac{\boldsymbol{v}^{mm}}{\|\boldsymbol{v}^{mm}\|},$$

where $\boldsymbol{p}^{mm}$ is the solution of ($\boldsymbol{p}$-SVM) and $\boldsymbol{v}^{mm}$ is the solution of ($\boldsymbol{v}$-SVM) with $\boldsymbol{r}_i = \boldsymbol{x}_{i\alpha_i}$.

As further discussion, consider Figure 2.4a where we set $n = 3, T = d = 2$ and $\boldsymbol{W} =$ Identity. All three inputs share the point $(0, 0)$ which corresponds to their non-optimal tokens. The optimal tokens (denoted by $\star$) are all support vectors of the ($\boldsymbol{v}$-SVM) since $\boldsymbol{v}^{\mathrm{mm}} = [0, 1]$ is the optimal classifier direction (depicted by - - -). Because of this, $\boldsymbol{p}^{\mathrm{mm}}$ will separate optimal $\star$ tokens from tokens at the $(0, 0)$ coordinate via ($\boldsymbol{p}$-SVM) and its direction is dictated by yellow and teal colored $\star$'s which are the support vectors.

2.4.2 General solution when selecting one token per input

Can we relax Assumption 2.3, and if so, what is the resulting behavior? Consider the scenario where the optimal $\boldsymbol{p}$ diverges to ∞ and ends up selecting one token per input. Suppose this $\boldsymbol{p}$



(a) All are support vectors (b) (0.5,1.5) is not a support (c) Probability evolutions in **(a)**

Figure 2.4: **(a)** and **(b)** Joint convergence of attention weights $\boldsymbol{p}$ (—>—) and classifier head $\boldsymbol{v}$ (—>—) to max-margin directions. **(c)** Averaged softmax probability evolution of optimal tokens and logistic probability evolution of output in **(a)**.

selects some coordinates $\boldsymbol{\alpha} = (\alpha_i)_{i=1}^n$. Let $\mathcal{S} \subset [n]$ be the set of indices where the associated token $\boldsymbol{x}_{i\alpha_i}$ is a support vector when solving ($\boldsymbol{v}$-SVM). Set $\bar{\mathcal{S}} = [n] - \mathcal{S}$. Our intuition is as follows: Even if we slightly perturb this $\boldsymbol{p}$ choice and mix other tokens $t \neq \alpha_i$ over the input set $\bar{\mathcal{S}} \subset [n]$, since $\bar{\mathcal{S}}$ is not support vector for ($\boldsymbol{v}$-SVM), we can preserve the label margin (by only preserving the support vectors $\mathcal{S}$). This means that $\boldsymbol{p}$ may not have to enforce *max-margin* constraint over inputs $i \in \bar{\mathcal{S}}$, instead, it suffices to just select these tokens (asymptotically). This results in the following **relaxed SVM** problem:

$$\boldsymbol{p}^{\text{relax}} = \arg\min_{\boldsymbol{p}} \|\boldsymbol{p}\| \quad \text{such that} \quad \boldsymbol{p}^\top(\boldsymbol{k}_{i\alpha_i} - \boldsymbol{k}_{it}) \geq \begin{cases} 1 & \text{for all} \quad t \neq \alpha_i,\ i \in \mathcal{S} \\ 0 & \text{for all} \quad t \neq \alpha_i,\ i \in \bar{\mathcal{S}} \end{cases} . \tag{2.11}$$

Here, $\boldsymbol{p}^\top(\boldsymbol{k}_{i\alpha_i} - \boldsymbol{k}_{it}) \geq 0$ corresponds to the *selection* idea. Building on this intuition, the following theorem captures the generalized behavior of the joint regularization path.

Theorem 2.6 *Consider Problem* (2.10). *Suppose* $\mathbb{S}(\boldsymbol{K}_i \boldsymbol{p}_R)_{\alpha_i} \to 1$, *i.e., the tokens* $(\alpha_i)_{i=1}^n$ *are asymptotically selected. Let* $\boldsymbol{v}^{mm}$ *be the solution of* ($\boldsymbol{v}$-SVM) *with* $\boldsymbol{r}_i = \boldsymbol{x}_{i\alpha_i}$ *and* $\mathcal{S}$ *be its set of support vector indices. Suppose Assumption 2.3 holds over* $\mathcal{S}$ *i.e. having* $\boldsymbol{s}_{i\alpha_i} < 1$ *shrinks the margin when* ($\boldsymbol{v}$-SVM) *is only solved over* $\mathcal{S} \subset [n]$. *Then, when* $r(R) = \exp(\mathcal{O}(R))$, *as* $R \to \infty$, *we have*

$$\lim_{R\to\infty} \frac{\boldsymbol{v}_r}{r} = \frac{\boldsymbol{v}^{mm}}{\|\boldsymbol{v}^{mm}\|}, \qquad \lim_{R\to\infty} \frac{\boldsymbol{p}_R}{R} = \frac{\boldsymbol{p}^{relax}}{\|\boldsymbol{p}^{relax}\|},$$

where $\boldsymbol{p}^{relax}$ *is the solution of* (2.11) *with* $(\alpha_i)_{i=1}^n$ *choices.*

To illustrate this numerically, consider Figure 2.4b which modifies Figure 2.4a by pushing

the yellow $\star$ to the northern position $(0.5, 1.5)$. We still have $\boldsymbol{v}^{\mathrm{mm}} = [0, 1]$ however the yellow $\star$ is no longer a support vector of ($\boldsymbol{v}$-SVM). Thus, $\boldsymbol{p}$ solves the relaxed problem (2.11) which separates green and teal $\star$'s by enforcing the max-margin constraint on $\boldsymbol{p}$ (which is the red direction). Instead, yellow $\star$ only needs to achieve positive correlation with $\boldsymbol{p}$ (unlike Figure 2.4a where it dictates the direction). We also display the direction of $\boldsymbol{p}^{\mathrm{mm}}$ using a gray dashed line.

We further investigate the evolution of softmax and logistic output probabilities throughout the training process of Figure 2.4a, and the results are illustrated in Figure 2.4c. The averaged softmax probability of optimal tokens is represented by the red curve and is calculated as $\frac{1}{n}\sum_{i=1}^{n} \max_{t\in[T]} \mathbb{S}(\boldsymbol{K}_i\boldsymbol{p})_t$. An achievement of 1 for this probability indicates that the attention mechanism successfully selects the optimal tokens. On the other hand, the logistic probability of the output is represented by the blue curve and is determined by $1/n \sum_{i=1}^{n} 1/(1 + e^{-y_i \cdot f(\boldsymbol{X}_i)})$. This probability also reaches a value of 1, suggesting that the inputs are correctly classified.

2.5 Regularization Path of Attention with Nonlinear Head

So far our discussion has focused on the attention model with linear head. However, the conceptual ideas on optimal token selection via margin maximization also extends to a general nonlinear model under mild assumptions. The aim of this section is showcasing this generalization. Specifically, we consider the prediction model $f(\boldsymbol{X}) = \psi(\boldsymbol{X}^\top \mathbb{S}(\boldsymbol{K}\boldsymbol{p}))$ where $\psi(\cdot) : \mathbb{R}^d \to \mathbb{R}$ generalizes the linear head $\boldsymbol{v}$ of our attention model. For instance, following exposition in Section 2.2, $\psi(\cdot)$ can represent a multilayer transformer with $\boldsymbol{p}$ being a tunable prompt at the input layer. Recall that $(\boldsymbol{X}_i, \boldsymbol{K}_i, y_i)_{i=1}^n$ is the dataset of the input-key-label tuples. We consider the training risk

$$\mathcal{L}(\boldsymbol{p}) = \frac{1}{n}\sum_{i=1}^{n} \ell(y_i, \psi(\boldsymbol{X}_i^\top \boldsymbol{s}_i^{\boldsymbol{p}})), \quad \text{where} \quad \boldsymbol{s}_i^{\boldsymbol{p}} = \mathbb{S}(\boldsymbol{K}_i\boldsymbol{p}) \in \mathbb{R}^T. \tag{2.12}$$

The challenge with nonlinear $\psi(\cdot)$ is that, we lack a clear score function (Def. 2.1) unlike the previous sections. The assumption below introduces a generic condition that splits the tokens of each $\boldsymbol{X}_i$ into an *optimal* set $\mathcal{O}_i$ and *non-optimal* set $\bar{\mathcal{O}}_i = [T] - \mathcal{O}_i$. In words, non-optimal tokens are those that strictly increase the training risk $\mathcal{L}(\boldsymbol{p})$ if they are not fully suppressed by attention probabilities $\boldsymbol{s}_i^{\boldsymbol{p}}$.

Assumption 2.4 (Mixing non-optimal tokens hurt) *There exists sets $(\mathcal{O}_i)_{i=1}^n \subset [T]$ as follows. Let $q_i^{\boldsymbol{p}} = \sum_{t\in\bar{\mathcal{O}}_i} \boldsymbol{s}_{it}^{\boldsymbol{p}}$ be the sum of softmax similarities over the non-optimal set for $\boldsymbol{p}$. Set $q_{\max}^{\boldsymbol{p}} = \max_{i\in[n]} q_i^{\boldsymbol{p}}$. For any $\Delta > 0$, there exists $\rho < 0$ such that:*

$$\text{For all } \boldsymbol{p}, \boldsymbol{p}' \in \mathbb{R}^d, \quad \text{if} \quad \log(q_{\max}^{\boldsymbol{p}}) \le (1+\Delta)\log(q_{\max}^{\boldsymbol{p}'}) \wedge \rho, \quad \text{then } \mathcal{L}(\boldsymbol{p}) < \mathcal{L}(\boldsymbol{p}').$$

This assumption is titled *mixing hurts* because the attention output $\boldsymbol{X}_i^\top \boldsymbol{s}_i^{\boldsymbol{p}}$ is mixing the tokens of $\boldsymbol{X}_i$ and our condition is that, to achieve optimal risk, this mixture should not contain any non-optimal tokens. In particular, we require that, a model $\boldsymbol{p}$ that contains *exponentially less non-optimality* (quantified via $\log(q_{\max})$) compared to $\boldsymbol{p}'$ is strictly preferable. As we discuss in the supplementary material, Theorem 2.1 is in fact a concrete instance (with linear head $\boldsymbol{v}$) satisfying this condition.

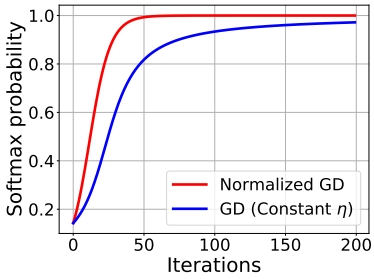
Before stating our generic theorem, we need to introduce the max-margin separator towards which regularization path of attention will converge. This is a slightly general version of Section 2.3's ($\boldsymbol{p}$-SVM) problem where we allow for a set of optimal tokens $\mathcal{O}_i$ for each input.

$$\boldsymbol{p}^{\mathrm{mm}} = \arg\min_{\boldsymbol{p}} \|\boldsymbol{p}\| \quad \text{s.t.} \quad \max_{\alpha\in\mathcal{O}_i} \min_{\beta\in\bar{\mathcal{O}}_i} \boldsymbol{p}^\top(\boldsymbol{k}_{i\alpha} - \boldsymbol{k}_{i\beta}) \ge 1, \quad \text{for all} \quad i \in [n]. \qquad (\boldsymbol{p}\text{-SVM'})$$

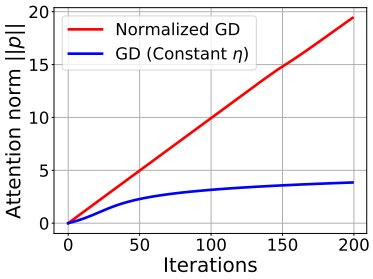
Unlike ($\boldsymbol{p}$-SVM), this problem is not necessarily convex when the optimal set $\mathcal{O}_i$ is not a singleton. To see this, imagine $n = d = 1$ and $T = 3$: Set the two optimal tokens as $\boldsymbol{k}_1 = 1$ and $\boldsymbol{k}_2 = -1$ and the non-optimal token as $\boldsymbol{k}_3 = 0$. The solution set of ($\boldsymbol{p}$-SVM') is $\boldsymbol{p}^{\mathrm{mm}} \in \{-1, 1\}$ whereas their convex combination $\boldsymbol{p} = 0$ violates the constraints. To proceed, our final result establishes the convergence of regularization path to the solution set of ($\boldsymbol{p}$-SVM') under Assumption 2.4.

Theorem 2.7 *Let $\mathcal{G}^{mm}$ be the set of global minima of ($\boldsymbol{p}$-SVM'). Suppose its margin $\Xi := 1/\|\boldsymbol{p}^{mm}\| > 0$ and Assumption 2.4 holds. Let $\texttt{dist}(\cdot,\cdot)$ denote the ℓ_2-distance between a vector and a set. Following (2.12), define $\bar{\boldsymbol{p}}(R) = \arg\min_{\|\boldsymbol{p}\|\le R} \mathcal{L}(\boldsymbol{p})$. We have that $\lim_{R\to\infty} \texttt{dist}\left(\frac{\bar{\boldsymbol{p}}(R)}{\Xi R}, \mathcal{G}^{mm}\right) = 0$.*

We note that Theorem 2.1 is a corollary of this result where $\mathcal{O}_i$'s and $\mathcal{G}^{\mathrm{mm}}$ are singleton. Based on this result, with multiple optimal tokens, Theorem 2.1 would gracefully generalize to solve ($\boldsymbol{p}$-SVM').



(a) Evolution of softmax prob.



(b) Evolution of $\|\boldsymbol{p}\|$

Figure 2.5: Evolution of softmax probability and attention weights when training with normalized gradient descent or constant step size η respectively.

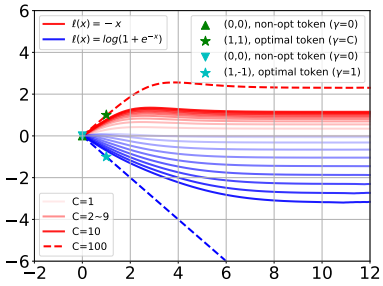


Figure 2.6: Trajectories of $\boldsymbol{p}$ with different loss functions and scores in Theorem 2.2.

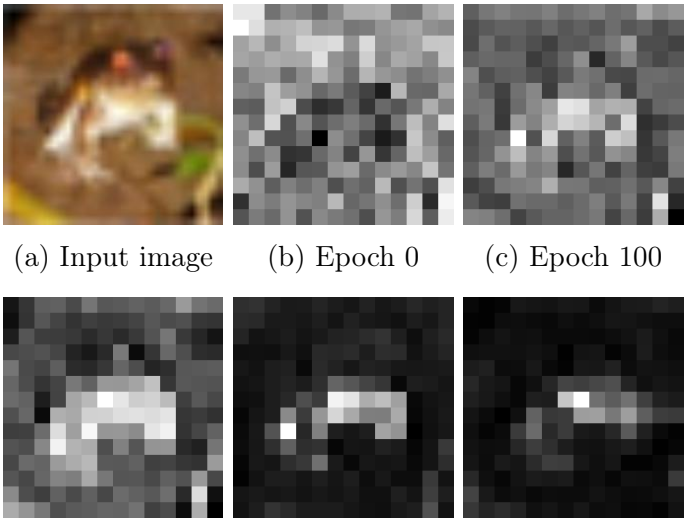
2.6 Experiments

Sparsity of softmax and evolution of attention weights. It is well known that, in practice, attention maps often exhibit sparsity and highlight salient tokens that aid inference. Our results provide a formal explanation of this when tokens are separable: Since attention selects a locally-optimal token within the input sequence and suppresses the rest, the associated attention map $\mathbb{S}(\boldsymbol{X}\boldsymbol{p})$ will (eventually) be a sparse vector. Additionally, the sparsity should arise in tandem with the increasing norm of attention weights. We provide empirical evidence to support these findings.

Synthetic experiments. Figures 2.5a and 2.5b show the evolution of the largest softmax probability and attention weights over time when using either normalized gradient or a fixed stepsize η for training. The dataset model follows Figure 2.1c. The softmax probability shown in Figure 2.5a is defined as $\frac{1}{n}\sum_{i=1}^{n}\max_{t\in[T]}\mathbb{S}(\boldsymbol{K}_i\boldsymbol{p})_t$. When this average probability reaches the value of 1, it means attention selects only a single token per input. The attention norm in Figure 2.5b, is simply equal to $\|\boldsymbol{p}\|$.

The red curves in both figures represent the normalized gradient method, which updates the model parameters $\boldsymbol{p}$ using $\boldsymbol{p}(t+1) = \boldsymbol{p}(t) - \eta\nabla\mathcal{L}(\boldsymbol{p}(t))/\|\nabla\mathcal{L}(\boldsymbol{p}(t))\|$ with $\eta = 0.1$. The blue curves correspond to gradient descent with constant learning rate given by $\boldsymbol{p}(t+1) = \boldsymbol{p}(t) - \eta\nabla\mathcal{L}(\boldsymbol{p}(t))$ with $\eta = 1$. Observe that the normalized gradient method achieves a softmax probability of 1 quicker as vanilla GD suffers from vanishing gradients. This is visible in Figure 2.5b where blue norm curve levels off.

Real experiments. To study softmax sparsity and the evolution of attention weights throughout training, we train a vision transformer (ViT-base) model [47] from scratch, utilizing the CIFAR-10 dataset [96] for 400 epochs with fixed learning rate 3×10^{-3}. ViT tokenizes an image into 16×16 patches, thus, its softmax attention maps can be easily visualized. We examine the average attention map – associated with the [CLS] token – computed from



(a) Input image (b) Epoch 0 (c) Epoch 100

(d) Epoch 200 (e) Epoch 300 (f) Epoch 400

Figure 2.7: Illustration of the progressive change in attention weights of the `[CLS]` token during training in the transformer model, using a specific input image shown in Figure 2.7a.

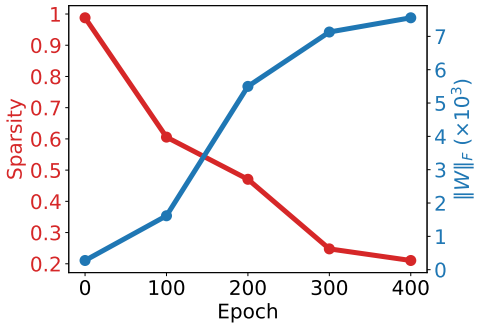


Figure 2.8: Red curve is the sparsity level $\widehat{\text{nnz}}(\boldsymbol{s})/\ T$ of the average attention map which takes values on [0,1]. A sparser vector implies that few key tokens receive significantly higher attention, while the majority of the tokens receive minimal attention. Blue curve is the Frobenius norm of attention weights $\|\boldsymbol{W}\|_F$ of the final layer. We display their evolutions over epochs.

all 12 attention heads within the model. Figure 2.7 provides a visual representation of the resulting attention weights (16×16 grids) corresponding to the original patch locations within the image. During the initial epochs of training, the attention weights are randomly distributed and exhibit a dense pattern. However, as the training progresses, the attention map gradually becomes sparser and the attention mechanism begins to concentrate on fewer salient patches within the image that possess distinct features that aid classification. This illustrates the evolution of attention from a random initial state to a more focused and sparse representation. These salient patches highlighted by attention conceptually corresponds to the optimal tokens within our theory.

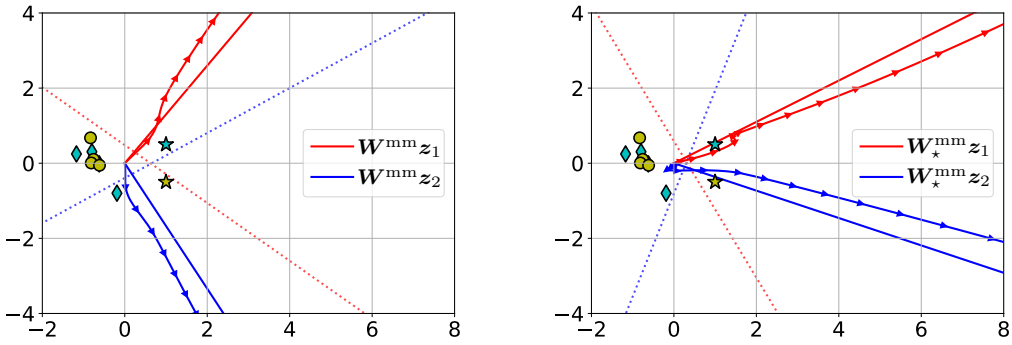
We quantify the sparsity of the attention map via a soft-sparsity measure, denoted by $\widehat{\text{nnz}}(\boldsymbol{s})$ where $\boldsymbol{s}$ is the softmax probability vector. The soft-sparsity is computed as the ratio of the ℓ_1–norm to the squared ℓ_2–norm, defined as $\widehat{\text{nnz}}(\boldsymbol{s}) = \|\boldsymbol{s}\|_1/\|\boldsymbol{s}\|^2$. $\widehat{\text{nnz}}(\boldsymbol{s})$ takes values between 1 to $T = 256$ and a smaller value indicates a sparser vector. Also note that $\|\boldsymbol{s}\|_1 = \sum_{t=1}^{T} \boldsymbol{s}_t = 1$. Together with sparsity, Figure 2.8 also displays the Frobenius norm of the combined key-query matrix $\boldsymbol{W}$ of the last attention layer over epochs. The theory suggests that the increase in sparsity is associated with the growth of attention weights – which converge directionally. The results in Figure 2.8 align with the theory, demonstrating the progressive sparsification of the attention map as $\|\boldsymbol{W}\|_F$ grows.

Transient optimization dynamics and the influence of the loss function. Theorem 2.2 shows that the asymptotic direction of gradient descent is determined by $\boldsymbol{p}^{\mathrm{mm}\star}$. However, it is worth noting that transient dynamics can exhibit bias towards certain input examples and their associated optimal tokens. We illustrate this idea in Fig 2.6, which displays the trajectories of the gradients for different scores and loss functions. We consider two optimal tokens ($\star$) with scores $\gamma_1 = 1$ and $\gamma_2 = C$, where C varies. For our analysis, we examine the correlation loss $\ell(x) = -x$ and the logistic loss $\ell(x) = \log(1 + e^{-x})$.

In essence, as C increases, we can observe that the correlation loss $\ell(x) = -x$ exhibits a bias towards the token with a high score, while the logistic loss is biased towards the token with a low score. The underlying reason for this behavior can be observed from the gradients of individual inputs: $\nabla \mathcal{L}_i(\boldsymbol{p}) = \ell_i' \cdot \boldsymbol{K}_i^\top \mathbb{S}'(\boldsymbol{X}\boldsymbol{p})\boldsymbol{X}\boldsymbol{v}$, where $\mathbb{S}'(\cdot)$ represents the derivative of the softmax function and $\ell_i' := \ell'(y_i \cdot \boldsymbol{v}^\top \boldsymbol{X}_i^\top \mathbb{S}(\boldsymbol{X}_i \boldsymbol{p}))$. Assuming that $\boldsymbol{p}$ (approximately) selects the optimal tokens, this simplifies to $\ell_i' \approx \ell'(\gamma_i)$ and $\|\nabla \mathcal{L}_i(\boldsymbol{p})\| \propto |\ell'(\gamma_i)| \cdot \gamma_i$. With the correlation loss, $|\ell'| = 1$, resulting in $\|\nabla \mathcal{L}_i(\boldsymbol{p})\| \propto \gamma_i$, meaning that a larger score induces a larger gradient. On the other hand, the logistic loss behaves similarly to the exponential loss under separable data, i.e., $|\ell'| = e^{-x}/(1 + e^{-x}) \approx e^{-x}$. Consequently, $\|\nabla \mathcal{L}_i(\boldsymbol{p})\| \propto \gamma_i e^{-\gamma_i} \approx e^{-\gamma_i}$, indicating that a smaller score leads to a larger gradient. These observations explain the empirical behavior we observe.

CHAPTER 3

Optimization of Projection Matrices: Softmax Attention as SVM



(a) $\boldsymbol{W}$-parameterization (b) $(\boldsymbol{W}_k, \boldsymbol{W}_q)$-parameterization

Figure 3.1: GD convergence during training of cross-attention weight $\boldsymbol{W}$ or $(\boldsymbol{W}_k, \boldsymbol{W}_q)$ with data. Teal and yellow markers represent tokens from $\boldsymbol{X}_1$ and $\boldsymbol{X}_2$, while stars mark optimal tokens. Solid lines in Figures **(a)** and **(b)** depict (W-SVM) and (W-SVM$_\star$) directions mapped to $\boldsymbol{z}_1$ (red) and $\boldsymbol{z}_2$ (blue), respectively. Arrows illustrating GD trajectories converging towards these SVM directions. Red and blue dotted lines represent the corresponding separating hyperplanes.

3.1 Introduction

In Chapter 2, we connected the optimization of prompts and prediction heads to the hard-margin SVM problem. In this chapter, we focus on the optimization behavior of projection matrices. Throughout, given input sequences $\boldsymbol{X}, \boldsymbol{Z} \in \mathbb{R}^{T \times d}$ with length T and embedding

dimension d, we study the core cross-attention and self-attention models:

$$f_{\texttt{cross}}(\boldsymbol{X}, \boldsymbol{Z}) := \mathbb{S}(\boldsymbol{Z}\boldsymbol{W}_q\boldsymbol{W}_k^\top\boldsymbol{X}^\top)\boldsymbol{X}\boldsymbol{W}_v, \tag{3.1a}$$

$$f_{\texttt{self}}(\boldsymbol{X}) := \mathbb{S}(\boldsymbol{X}\boldsymbol{W}_q\boldsymbol{W}_k^\top\boldsymbol{X}^\top)\boldsymbol{X}\boldsymbol{W}_v. \tag{3.1b}$$

Here, $\boldsymbol{W}_k, \boldsymbol{W}_q \in \mathbb{R}^{d\times m}$, $\boldsymbol{W}_v \in \mathbb{R}^{d\times v}$ are the trainable key, query, value matrices respectively; $\mathbb{S}(\cdot)$ denotes the softmax nonlinearity, which is applied row-wise on $\boldsymbol{X}\boldsymbol{W}_q\boldsymbol{W}_k^\top\boldsymbol{X}^\top$. Note that self-attention (3.1b) is a special instance of the cross-attention (3.1a) by setting $\boldsymbol{Z} \leftarrow \boldsymbol{X}$. To expose our main results, suppose the first token of $\boldsymbol{Z}$ – denoted by $\boldsymbol{z}$ – is used for prediction. Concretely, given a training dataset $(y_i, \boldsymbol{X}_i, \boldsymbol{z}_i)_{i=1}^n$ with labels $y_i \in \{-1, 1\}$ and inputs $\boldsymbol{X}_i \in \mathbb{R}^{T\times d}, \boldsymbol{z}_i \in \mathbb{R}^d$, we consider the empirical risk minimization with a decreasing loss function $\ell(\cdot) : \mathbb{R} \to \mathbb{R}$, represented as follows:

$$\begin{aligned} &\mathcal{L}(\boldsymbol{W}_k, \boldsymbol{W}_q) = \frac{1}{n}\sum_{i=1}^n \ell\left(y_i \cdot f(\boldsymbol{X}_i, \boldsymbol{z}_i)\right), \\ &\text{where} \quad f(\boldsymbol{X}_i, \boldsymbol{z}_i) = h\left(\boldsymbol{X}_i^\top \mathbb{S}\left(\boldsymbol{X}_i\boldsymbol{W}_k\boldsymbol{W}_q^\top\boldsymbol{z}_i\right)\right). \end{aligned} \tag{3.2}$$

Here, $h(\cdot) : \mathbb{R}^d \to \mathbb{R}$ is the prediction head that subsumes the value weights $\boldsymbol{W}_v$. In this formulation, the model $f(\cdot)$ precisely represents a one-layer transformer where an MLP follows the attention layer. Note that, we recover the self-attention in (3.2) by setting $\boldsymbol{z}_i \leftarrow \boldsymbol{x}_{i1}$, where $\boldsymbol{x}_{i1}$ denotes the first token of the sequence $\boldsymbol{X}_i$[1]. The softmax operation, due to its nonlinear nature, poses a significant challenge when optimizing (3.2). The problem is nonconvex and nonlinear even when the prediction head is fixed and linear. In this study, we focus on optimizing the attention weights ($\boldsymbol{W}_k, \boldsymbol{W}_q$ or $\boldsymbol{W}$) and overcome such challenges to establish a fundamental SVM equivalence.[2]

The main contributions are as follows:

- ◇ **Implicit bias of the attention layer (§3.2-3.3):** Optimizing the attention parameters ($\boldsymbol{W}_k, \boldsymbol{W}_q$) with vanishing regularization converges in direction towards a max-margin solution of (W-SVM$_\star$) with the nuclear norm objective of the combined parameter $\boldsymbol{W} := \boldsymbol{W}_k\boldsymbol{W}_q^\top$ (Thm 3.2). In the case of directly parameterizing cross-attention by the combined parameter $\boldsymbol{W}$, the regularization path (RP) directionally converges to (W-SVM) solution with the Frobenius norm objective. To our knowledge, this is the first result to formally distinguish the optimization dynamics of $\boldsymbol{W}$ vs $(\boldsymbol{W}_k, \boldsymbol{W}_q)$

[1] Note that for simplicity, we set $\boldsymbol{z}_i = \boldsymbol{x}_{i1}$, but it can be any other row of $\boldsymbol{X}_i$.

[2] We fix $h(\cdot)$ and only optimize the attention weights. This is partly to avoid the degenerate case where $h(\cdot)$ can be used to achieve zero training loss (e.g. via standard arguments like NTK [83]) without providing any meaningful insight into the functionality of the attention mechanism.

parameterizations, revealing the low-rank bias of the latter. Our theory clearly characterizes the *optimality* of selected tokens (Definition 3.1) and naturally extends to sequence-to-sequence or causal classification settings (see SAtt-SVM and Theorem B.2 in appendix).

- ◇ **Convergence of gradient descent (§3.4-3.5):** Gradient descent (GD) iterates for the combined key-query variable $\boldsymbol{W}$ converge in direction to a *locally-optimal* solution of (W-SVM) with appropriate initialization and a linear head $h(\cdot)$ (Sec. 3.5). For local optimality, selected tokens must have higher scores than their neighboring tokens. Locally-optimal directions are not necessarily unique and are characterized in terms of the problem geometry. As a key contribution, we identify geometric conditions that guarantee convergence to the globally-optimal direction (Sec. 3.4). Besides these, we show that over-parameterization (i.e. dimension d being large, and equivalent conditions) catalyzes global convergence by ensuring **(1)** feasibility of (W-SVM), and, **(2)** benign optimization landscape, in the sense that there are no stationary points and no spurious locally-optimal directions (see Sec. 3.5.2). These are illustrated in Figures 3.1 and 3.4.

- ◇ **Generality of SVM equivalence (§3.6):** When optimizing with linear $h(\cdot)$, the attention layer is inherently biased towards selecting a single token from each sequence (a.k.a. hard attention). This is reflected in (W-SVM) and arises from output tokens being convex combinations of the input tokens. In contrast, we show that nonlinear heads necessitate composing multiple tokens, highlighting their importance in the transformer's dynamics (Sec. 3.6.1). Using insights gathered from our theory, we propose a more general SVM equivalence. Remarkably, we demonstrate that our proposal accurately predicts the implicit bias of attention trained by gradient descent under general scenarios not covered by theory (e.g. $h(\cdot)$ being an MLP). Specifically, our general formulae decouple attention weights into two components: A **directional component** governed by SVM which selects the tokens by applying a 0-1 mask, and a **finite component** which dictates the precise composition of the selected tokens by adjusting the softmax probabilities.

An important feature of these findings is that they apply to arbitrary datasets (whenever SVM is feasible) and are numerically verifiable. We extensively validate the max-margin equivalence and implicit bias of transformers through enlightening experiments. We hold the view that these findings aid in understanding transformers as hierarchical max-margin token-selection mechanisms, and we hope that our outcomes will serve as a foundation for upcoming studies concerning their optimization and generalization dynamics.

3.2 Preliminaries and Problem Setup

Notation. For any integer $N \geq 1$, let $[N] := \{1, \ldots, N\}$. We use lowercase and uppercase bold letters (e.g., $\boldsymbol{a}$ and $\boldsymbol{A}$) to represent vectors and matrices, respectively. The entries of a vector $\boldsymbol{a}$ are denoted as $\boldsymbol{a}_i$. For a matrix $\boldsymbol{A}$, $\|\boldsymbol{A}\|$ denotes the spectral norm, i.e. maximum singular value, $\|\boldsymbol{A}\|_\star$ denotes the nuclear norm, i.e. summation of all singular values, and $\|\boldsymbol{A}\|_F := \sqrt{\mathtt{tr}\left(\boldsymbol{A}^\top \boldsymbol{A}\right)}$ denotes the Frobenius norm. $\mathtt{dist}\left(\cdot, \cdot\right)$ denotes the Euclidean distance between two sets. The minimum / maximum of two numbers a, b is denoted as $a \wedge b$ / $a \vee b$. The big-O notation $\mathcal{O}(\cdot)$ hides the universal constants.

Unveiling the relationship between attention and linear SVMs. For linear classification, it is well-established that GD iterations on logistic loss and separable datasets converge towards the hard margin SVM solution, which effectively separates the two classes within the data [169, 177, 215]. The softmax nonlinearity employed by the attention layer exhibits an exponentially-tailed behavior similar to the logistic loss; thus, attention may also be biased towards margin-maximizing solutions.

However, the attention layer operates on tokens within an input sequence, rather than performing classification directly. Therefore, its bias is towards an SVM, specifically (W-SVM), which aims to separate the tokens of input sequences by selecting the relevant ones and suppressing the rest. Nonetheless, formalizing this intuition is considerably more challenging: The presence of the highly nonlinear and nonconvex softmax operation renders the analysis of standard GD algorithms intricate. Additionally, the 1-layer transformer in (3.2) does not inherently exhibit a singular bias towards a single (W-SVM) problem, even when using a linear head h. Instead, it can result in multiple locally optimal directions induced by their associated SVMs. We emphasize that [186] is the first work to make this attention↔SVM connection. Here, we augment their framework to transformers by developing the first guarantees for self/cross-attention layer, nonlinear prediction heads, and global convergence.

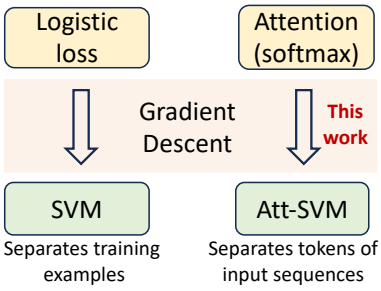


Figure 3.2: Implicit biases of the attention layer and logistic regression.

Optimization problem definition. We use a linear head $h(\boldsymbol{x}) = \boldsymbol{v}^\top \boldsymbol{x}$ for most of our theoretical exposition. Given dataset $(y_i, \boldsymbol{X}_i, \boldsymbol{z}_i)_{i=1}^n$, we minimize the empirical risk of an 1-layer transformer using combined weights $\boldsymbol{W} \in \mathbb{R}^{d \times d}$ or individual weights $\boldsymbol{W}_k, \boldsymbol{W}_q \in \mathbb{R}^{d \times m}$ for a fixed head and decreasing loss function:

$$\mathcal{L}(\boldsymbol{W}) = \frac{1}{n}\sum_{i=1}^{n} \ell\left(y_i \cdot \boldsymbol{v}^\top \boldsymbol{X}_i^\top \mathbb{S}(\boldsymbol{X}_i \boldsymbol{W} \boldsymbol{z}_i)\right), \tag{W-ERM}$$

$$\mathcal{L}(\boldsymbol{W}_k, \boldsymbol{W}_q) = \frac{1}{n}\sum_{i=1}^{n} \ell\left(y_i \cdot \boldsymbol{v}^\top \boldsymbol{X}_i^\top \mathbb{S}(\boldsymbol{X}_i \boldsymbol{W}_k \boldsymbol{W}_q^\top \boldsymbol{z}_i)\right). \tag{KQ-ERM}$$

We can recover the self-attention model by setting $\boldsymbol{z}_i$ to be the first token of $\boldsymbol{X}_i$, i.e., $\boldsymbol{z}_i \leftarrow \boldsymbol{x}_{i1}$. While the above formulation regresses a single label y_i per $(\boldsymbol{X}_i, \boldsymbol{z}_i)$, in Sections 3.6 and B.5, we show that our findings gracefully extend to the sequence-to-sequence classification setting where we output and classify T tokens per inputs $\boldsymbol{X}_i, \boldsymbol{Z}_i \in \mathbb{R}^{T\times d}$. See Sections 3.6 and B.5 for results on nonlinear prediction heads.

Optimization algorithms. Given a parameter $R > 0$, we consider an ℓ_2-norm bound R, and define the regularized path solution associated with Objectives (W-ERM) and (KQ-ERM), respectively as (W-RP) and (KQ-RP). These update rules allow us to find the solution within a constrained region defined by the norm bound. The RP illustrates the evolution of $\bar{\boldsymbol{W}}_R$ as R increases, capturing the essence of GD where the ridge constraint serves as an approximation for the number of iterations. Previous studies, including [87, 169, 179, 186], have examined the implicit bias of logistic regression and established a connection between the directional convergence of the RP (i.e., $\lim_{R\to\infty} \bar{\boldsymbol{W}}_R/R$) and GD. In [186], the concept of a local RP was also employed to investigate implicit bias along local directions. For GD, with appropriate initialization and step size $\eta > 0$, we describe the optimization process associated with (W-ERM) and (KQ-ERM) as (W-GD) and (KQ-GD), respectively.

Given $\boldsymbol{W}(0) \in \mathbb{R}^{d\times d}$, $\eta > 0$, for $k \geq 0$ do:	Given $\boldsymbol{W}_q(0), \boldsymbol{W}_k(0) \in \mathbb{R}^{d\times m}$, $\eta > 0$, for $k \geq 0$ do:
$\boldsymbol{W}(k+1) = \boldsymbol{W}(k) - \eta \nabla \mathcal{L}(\boldsymbol{W}(k))$. (W-GD)	$\begin{bmatrix}\boldsymbol{W}_k(k+1)\\ \boldsymbol{W}_q(k+1)\end{bmatrix} = \begin{bmatrix}\boldsymbol{W}_k(k)\\ \boldsymbol{W}_q(k)\end{bmatrix} - \eta \begin{bmatrix}\nabla_{\boldsymbol{W}_k}\mathcal{L}\left(\boldsymbol{W}_k(k), \boldsymbol{W}_q(k)\right)\\ \nabla_{\boldsymbol{W}_q}\mathcal{L}\left(\boldsymbol{W}_k(k), \boldsymbol{W}_q(k)\right)\end{bmatrix}$. (KQ-GD)
Given $R > 0$, find $d \times d$ matrix:	Given $R > 0$, find $d \times m$ matrices:
$\bar{\boldsymbol{W}}(R) = \underset{\|\boldsymbol{W}\|_F \leq R}{\arg\min}\ \mathcal{L}(\boldsymbol{W})$. (W-RP)	$(\bar{\boldsymbol{W}}_k(R), \bar{\boldsymbol{W}}_q(R)) = \underset{\|\boldsymbol{W}_k\|_F^2 + \|\boldsymbol{W}_q\|_F^2 \leq 2R}{\arg\min} \mathcal{L}(\boldsymbol{W}_k, \boldsymbol{W}_q)$. (KQ-RP)

3.2.1 Optimal tokens and hard-margin SVM problem for cross attention

Given $\boldsymbol{X}_i \in \mathbb{R}^{T\times d}, \boldsymbol{z}_i \in \mathbb{R}^d$, we present a convex hard-margin SVM problem, denoted as (W-SVM), that aims to separate a specific token from the remaining tokens in the input

sequence $\boldsymbol{X}_i$. This problem is jointly solved for all inputs, allowing us to examine the optimization properties of cross-attention. To delve deeper, we introduce the concept of optimal tokens, which are tokens that minimize the training objective under the decreasing loss function $\ell(\cdot)$. To illustrate this, the token at the output of the attention layer is given by $\boldsymbol{a}_i = \boldsymbol{X}_i^\top \boldsymbol{s}_i$, where $\mathbb{S}(\boldsymbol{X}_i \boldsymbol{W} \boldsymbol{z}_i) = \boldsymbol{s}_i$. Here, $\boldsymbol{a}_i$ can be written as $\boldsymbol{a}_i = \sum_{t\in[T]} \boldsymbol{s}_{it}\boldsymbol{x}_{it}$, where $\boldsymbol{s}_{it} \geq 0$ and $\sum_{t\in[T]} \boldsymbol{s}_{it} = 1$. Now, assuming the linearity of $h(\boldsymbol{x}) = \boldsymbol{v}^\top \boldsymbol{x}$ and that for a specific token $\{\text{opt}_i\}$, $\gamma_{it} = y_i \cdot h(\boldsymbol{x}_{it}) = y_i \cdot \boldsymbol{v}^\top \boldsymbol{x}_{it} \leq \gamma_{i\text{opt}_i}$, we get

$$\frac{1}{n}\sum_{i=1}^{n} \ell(y_i \cdot \sum_{t\in[T]} \boldsymbol{s}_{it} h(\boldsymbol{x}_{it})) \geq \frac{1}{n}\sum_{i=1}^{n} \ell(y_i \cdot h(\boldsymbol{x}_{i\text{opt}_i})),$$

which implies that

$$\begin{aligned}
\mathcal{L}(\boldsymbol{W}) &= \frac{1}{n}\sum_{i=1}^{n} \ell(y_i \cdot h(\boldsymbol{a}_i)) \\
&= \frac{1}{n}\sum_{i=1}^{n} \ell(y_i \cdot \sum_{t\in[T]} \boldsymbol{s}_{it} h(\boldsymbol{x}_{it})) \\
&\geq \frac{1}{n}\sum_{i=1}^{n} \ell(y_i \cdot h(\boldsymbol{x}_{i\text{opt}_i})) = \frac{1}{n}\sum_{i=1}^{n} \ell(\gamma_{i\text{opt}_i}) = \mathcal{L}_\star.
\end{aligned}$$

Here, the inequality follows since $\gamma_{it} = y_i \cdot h(\boldsymbol{x}_{it}) = y_i \cdot \boldsymbol{v}^\top \boldsymbol{x}_{it} \leq \gamma_{i\text{opt}_i}$ by Definition 3.1 and the strictly decreasing nature of the loss ℓ due to Assumption 3.1. As we can see, γ_{it} contributes to the objective decrease, and larger values of γ_{it} further minimize the training objective under the decreasing loss function $\ell(\cdot)$. This is further discussed in Lemma 3.2. This exploration introduces the notions of token scores and optimality, providing insights into the underlying principles of self-attention mechanisms [186].

Definition 3.1 (Token Score and Optimality) *Given a prediction head $\boldsymbol{v} \in \mathbb{R}^d$, the score of a token $\boldsymbol{x}_{it}$ of input $\boldsymbol{X}_i$ is defined as $\gamma_{it} = y_i \cdot \boldsymbol{v}^\top \boldsymbol{x}_{it}$. The optimal token for each input $\boldsymbol{X}_i$ is given by the index $opt_i \in \arg\max_{t\in[T]} \gamma_{it}$ for all $i \in [n]$.*

By introducing token scores and identifying optimal tokens, we can better understand the importance of individual tokens and their impact on the overall objective. The token score quantifies the contribution of a token to the prediction or classification task, while the optimal token represents the token that exhibits the highest relevance within the corresponding input sequence.

• **Hard-margin SVM for W-parameterization.** Equipped with the set of optimal indices $\text{opt} := (\text{opt}_i)_{i=1}^n$ as per Definition 3.1, we introduce the following SVM formulation associated to (W-ERM):

$$\boldsymbol{W}^{\text{mm}} = \arg\min_{\boldsymbol{W}} \|\boldsymbol{W}\|_F \quad \text{s.t.} \quad (\boldsymbol{x}_{i\text{opt}_i} - \boldsymbol{x}_{it})^\top \boldsymbol{W} \boldsymbol{z}_i \geq 1 \quad \forall\, t \neq \text{opt}_i,\ i \in [n]. \tag{W-SVM}$$

The existence of matrix $\boldsymbol{W}^{\text{mm}}$ implies the separability of tokens $(\text{opt}_i)_{i=1}^n$ from the others. The terms $a_{it} := \boldsymbol{x}_{it}^\top \boldsymbol{W} \boldsymbol{z}_i$ represent the dot product between the key-query features before applying the softmax nonlinearity. This dot product is a crucial characteristic of self-attention and we can express the SVM constraints in (W-SVM) as $a_{i\text{opt}_i} \geq a_{it} + 1$. Thus, (W-SVM) finds the most efficient direction that ensures the optimal token $\boldsymbol{x}_{i\text{opt}_i}$ achieves the highest similarity with query $\boldsymbol{z}_i$ among all key embeddings. Our first result shows that (W-SVM) is feasible under mild over-parameterization.

Theorem 3.1 *Suppose $d \geq \max(T-1, n)$. Then, almost all datasets $(y_i, \boldsymbol{X}_i, \boldsymbol{z}_i)_{i=1}^n$ – including the self-attention setting with $\boldsymbol{z}_i \leftarrow \boldsymbol{x}_{i1}$ – obey the following:* (W-SVM) *is feasible i.e., $\boldsymbol{W}^{mm}$ separates the desired tokens $opt = (opt_i)_{i=1}^n$.*

We note that the convex formulation (W-SVM) does not fully capture the GD geometry on (W-ERM). In a more general sense, GD can provably converge to an SVM solution over locally-optimal tokens, as detailed in Section 3.5.1. For deeper insight into (W-SVM), consider that the attention layer's output is a convex mixture of input tokens. Thus, if minimizing the training loss involves choosing the optimal token $\boldsymbol{x}_{i\text{opt}_i}$, the softmax similarities should eventually converge to a one-hot vector, precisely including $\boldsymbol{x}_{i\text{opt}_i}$ (assigned 1), while ignoring all other tokens (assigned 0s). This convergence requires the attention weights $\boldsymbol{W}$ to diverge in norm to saturate the softmax probabilities. Due to the exponential-tailed nature of the softmax function, the weights converge directionally to the max-margin solution. This phenomenon resembles the implicit bias of logistic regression on separable data [88, 177]. Lemma 3.2 formalizes this intuition and rigorously motivates optimal tokens.

• **Non-convex SVM for $(\boldsymbol{W}_k, \boldsymbol{W}_q)$-parameterization.** The objective function (KQ-ERM) has an extra layer of nonconvexity compared to (W-ERM) as $(\boldsymbol{W}_k, \boldsymbol{W}_q)$ corresponds to a matrix factorization of $\boldsymbol{W}$. To study this, we introduce the following nonconvex SVM problem over $(\boldsymbol{W}_k, \boldsymbol{W}_q)$ akin to (W-SVM):

$$\min_{\boldsymbol{W}_k, \boldsymbol{W}_q} \frac{1}{2}\left(\|\boldsymbol{W}_k\|_F^2 + \|\boldsymbol{W}_q\|_F^2\right) \quad \text{s.t.} \quad (\boldsymbol{x}_{i\text{opt}_i} - \boldsymbol{x}_{it})^\top \boldsymbol{W}_k \boldsymbol{W}_q^\top \boldsymbol{z}_i \geq 1 \quad \forall\, t \neq \text{opt}_i,\ i \in [n]. \tag{KQ-SVM}$$

Even if the direction of GD is biased towards the SVM solution, it does not have to converge to the global minima of (KQ-SVM). Instead, it can converge towards a Karush–Kuhn–Tucker (KKT) point of the max-margin SVM. Such KKT convergence has been studied by [91, 128] in the context of other nonconvex margin maximization problems. Fortunately, (KQ-ERM) may not be as daunting as it may initially seem: Our experiments in Figures 3.1 and 3.6 reveal that GD is indeed biased towards the global minima of (KQ-SVM). This global minima is achieved by setting $\boldsymbol{W} := \boldsymbol{W}_k \boldsymbol{W}_q^\top$ and finding the factorization of $\boldsymbol{W}$ that minimizes the quadratic objective, yielding the following $\boldsymbol{W}$-parameterized SVM with nuclear norm objective:

$$\boldsymbol{W}_\star^{\mathrm{mm}} \in \underset{\mathtt{rank}(\boldsymbol{W})\le m}{\arg\min} \|\boldsymbol{W}\|_\star \quad \text{s.t.} \quad (\boldsymbol{x}_{i\mathrm{opt}_i} - \boldsymbol{x}_{it})^\top \boldsymbol{W} \boldsymbol{z}_i \ge 1 \quad \forall\, t \ne \mathrm{opt}_i,\ i \in [n]. \tag{W-SVM$_\star$}$$

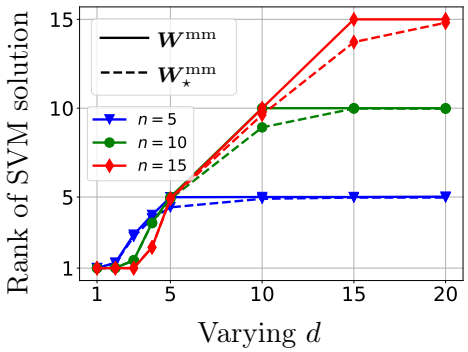
Above, the nonconvex rank constraint arises from the fact that the rank of $\boldsymbol{W} = \boldsymbol{W}_k \boldsymbol{W}_q^\top$ is at most m. However, under the condition of the full parameterization where $m \ge d$, the rank constraint disappears, leading to a convex nuclear norm minimization problem. Besides, the nuclear norm objective inherently encourages a low-rank solution [55, 166, 178]. Lemma 3.1, presented below, demonstrates that this guarantee holds whenever $n \le m$. This observation is further supported by our experiments (see Fig. 3.3). Thus, it offers a straightforward rationale for why setting $m < d$ is a reasonable practice, particularly in scenarios involving limited data.

Lemma 3.1 *Any optimal solution of* (W-SVM) *or* (W-SVM$_\star$) *is at most rank* n*. More precisely, the row space of* $\boldsymbol{W}^{mm}$ *or* $\boldsymbol{W}_\star^{mm}$ *lies within* $\mathtt{span}(\{\boldsymbol{z}_i\}_{i=1}^n)$.

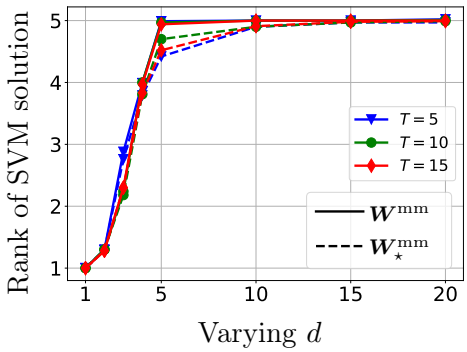
Figure 3.3 illustrates rank range of solutions for (W-SVM) and (W-SVM$_\star$), denoted as $\boldsymbol{W}^{\mathrm{mm}}$ and $\boldsymbol{W}_\star^{\mathrm{mm}}$, solved using optimal tokens $(\mathrm{opt}_i)_{i=1}^n$ and setting $m = d$ (the rank constraint is eliminated). Each result is averaged over 100 trials, and for each trial, $\boldsymbol{x}_{it}$, $\boldsymbol{z}_i$, and linear head $\boldsymbol{v}$ are randomly sampled from the unit sphere. In Fig. 3.3a, we fix $T = 5$ and vary n across $\{5, 10, 15\}$. Conversely, in Fig. 3.3b, we keep $n = 5$ constant and alter T across $\{5, 10, 15\}$. Both figures confirm rank of $\boldsymbol{W}^{\mathrm{mm}}$ and $\boldsymbol{W}_\star^{\mathrm{mm}}$ are bounded by $\max(n, d)$, validating Lemma 3.1.

3.3 Understanding the Implicit Bias of Self-Attention

We start by motivating the optimal token definition and establishing the global convergence of RPs which shed light on the implicit bias of attention parameterizations. Throughout, we maintain the following assumption regarding the loss function.



(a) Rank of attention SVM solutions ($T = 5$)



(b) Rank of attention SVM solutions ($n = 5$)

Figure 3.3: Rank range of solutions for (W-SVM) and (W-SVM$_\star$), denoted as $\boldsymbol{W}^{\mathrm{mm}}$ and $\boldsymbol{W}_\star^{\mathrm{mm}}$, solved using optimal tokens $(\mathrm{opt}_i)_{i=1}^n$ and setting $m = d$ (the rank constraint is eliminated). Both figures confirm ranks of $\boldsymbol{W}^{\mathrm{mm}}$ and $\boldsymbol{W}_\star^{\mathrm{mm}}$ are bounded by $\max(n, d)$, validating Lemma 3.1.

Assumption 3.1 *Over any bounded interval $[a, b]$: (i) $\ell : \mathbb{R} \to \mathbb{R}$ is strictly decreasing; (ii) The derivative ℓ' is bounded as $|\ell'(u)| \leq M_1$; (iii) ℓ' is M_0-Lipschitz continuous.*

Assumption 3.1 encompasses many common loss functions, e.g., logistic $\ell(u) = \log(1 + e^{-u})$, exponential $\ell(u) = e^{-u}$, and correlation $\ell(u) = -u$ losses.

Lemma 3.2 (Optimal Tokens Minimize Training Loss) *Suppose Assumption 3.1 (i)-(ii) hold, and not all tokens are optimal per Definition 3.1. Then, training risk obeys $\mathcal{L}(\boldsymbol{W}) > \mathcal{L}_\star := \frac{1}{n}\sum_{i=1}^n \ell(\gamma_{i\mathrm{opt}_i})$. Additionally, suppose there are optimal indices $(\mathrm{opt}_i)_{i=1}^n$ for which* (W-SVM) *is feasible, i.e. there exists a $\boldsymbol{W}$ separating optimal tokens. This $\boldsymbol{W}$ choice obeys $\lim_{R\to\infty} \mathcal{L}(R \cdot \boldsymbol{W}) = \mathcal{L}_\star$.*

The result presented in Lemma 3.2 originates from the observation that the output tokens of the attention layer constitute a convex combination of the input tokens. Consequently, when subjected to a strictly decreasing loss function, attention optimization inherently leans towards the selection of a singular token, specifically, the optimal token $(\mathrm{opt}_i)_{i=1}^n$. Our following theorem unveils the implicit bias ingrained within both attention parameterizations through RP analysis.

Theorem 3.2 *Suppose Assumptions 3.1 holds, optimal indices $(\mathrm{opt}_i)_{i=1}^n$ are unique, and* (W-SVM) *is feasible. Let $\boldsymbol{W}^{mm}$ be the unique solution of* (W-SVM)*, and let $\mathcal{W}_\star^{mm}$ be the solution set of* (W-SVM$_\star$) *with nuclear norm achieving objective $C_\star$. Then, Algorithms W-RP and KQ-RP, respectively, satisfy:*

- $\boldsymbol{W}$*-parameterization has Frobenius norm bias:* $\lim_{R\to\infty} \frac{\bar{\boldsymbol{W}}_R}{R} = \frac{\boldsymbol{W}^{mm}}{\|\boldsymbol{W}^{mm}\|_F}$.
- $(\boldsymbol{W}_k, \boldsymbol{W}_q)$*-parameterization has nuclear norm bias:* $\lim_{R\to\infty} \texttt{dist}\left(\frac{\bar{\boldsymbol{W}}_k(R)\bar{\boldsymbol{W}}_q(R)^\top}{R}, \frac{\mathcal{W}_\star^{mm}}{C_\star}\right) = 0$.
  - *Setting* $m = d$*:* (W-SVM$_\star$) *is a convex problem without rank constraints.*

Theorem 3.2 demonstrates that the RP of the $(\boldsymbol{W}_k, \boldsymbol{W}_q)$-parameterization converges to a max-margin solution of (W-SVM$_\star$) with nuclear norm objective on $\boldsymbol{W} = \boldsymbol{W}_k\boldsymbol{W}_q^\top$. When self-attention is directly parameterized by $\boldsymbol{W}$, the RP converges to the solution of (W-SVM) with a Frobenius norm objective. This result is the first to distinguish the optimization dynamics of $\boldsymbol{W}$ and $(\boldsymbol{W}_k, \boldsymbol{W}_q)$ parameterizations, revealing the low-rank bias of the latter. These findings also provide a clear characterization of token optimality (Definition 3.1) and extend naturally to the setting with multiple optimal tokens per sequence (Theorem B.2 in appendix). By definition, the RP captures the global geometry and cannot be used for the implicit bias of GD towards locally-optimal directions. Sections 3.4 and 3.5 accomplish this goal through gradient-descent and localized RP analysis to obtain locally-applicable SVM equivalences. Note that, this theorem requires each input sequence has a unique optimal token per Definition 3.1. Fortunately, this is a very mild condition as it holds for almost all datasets, namely, as soon as input features are slightly perturbed.

Theorem 3.2 establishes the implicit bias of attention from the perspective of RP analysis. This leads to the question: To what extent is this RP theory predictive of the implicit bias exhibited by GD? To delve into this, we examine the gradient paths of $\boldsymbol{W}(k)$ or $(\boldsymbol{W}_k(k), \boldsymbol{W}_q(k))$ and present the findings in Figure 3.1. We consider a scenario where $n = d = m = 2$ and $T = 5$, and employ cross-attention, where tokens $(\boldsymbol{z}_1, \boldsymbol{z}_2)$ are generated independently of the inputs $(\boldsymbol{X}_1, \boldsymbol{X}_2)$. The teal and yellow markers correspond to tokens from $\boldsymbol{X}_1$ and $\boldsymbol{X}_2$, respectively. The stars indicate the optimal token for each input. To provide a clearer view of the gradient convergence path, we illustrate the outcomes of training the attention weight $\boldsymbol{W}$ or $(\boldsymbol{W}_k, \boldsymbol{W}_q)$ in the form of $\boldsymbol{W}\boldsymbol{z}_i$ or $\boldsymbol{W}_k\boldsymbol{W}_q^\top\boldsymbol{z}_i$, where $i = 1, 2$. With reference to Equations (W-SVM) and (W-SVM$_\star$), the red and blue solid lines in Fig. 3.1a delineate the directions of $\boldsymbol{W}^{\mathrm{mm}}\boldsymbol{z}_1$ and $\boldsymbol{W}^{\mathrm{mm}}\boldsymbol{z}_2$, correspondingly. Conversely, the red and blue solid lines in Fig. 3.1b show the directions of $\boldsymbol{W}_\star^{\mathrm{mm}}\boldsymbol{z}_1$ and $\boldsymbol{W}_\star^{\mathrm{mm}}\boldsymbol{z}_2$. The red/blue arrows denote the corresponding directions of gradient evolution with the dotted lines representing the corresponding separating hyperplanes. Figure 3.1 provides a clear depiction of the incremental alignment of $\boldsymbol{W}(k)$ and $\boldsymbol{W}_k(k)\boldsymbol{W}_q(k)^\top$ with their respective attention SVM solutions as k increases. This strongly supports the assertions of Theorem 3.2.

It is worth noting that (W-SVM$_\star$) imposes a nonconvex rank constraint, i.e., $\texttt{rank}(\boldsymbol{W}) \leq m$. Nevertheless, this constraint becomes inconsequential if the unconstrained problem, with

m set to be greater than or equal to d, admits a low-rank solution, as demonstrated in Lemma 3.1. Consequently, in our experimental endeavors, we have the flexibility to employ the unconstrained attention SVM for predicting the implicit bias. This concept is succinctly summarized by the following lemma.

Lemma 3.3 *Let $\mathcal{W}_\star^{mm}$ be the solution set of* (W-SVM$_\star$) *with nuclear norm achieving objective $C_\star$. Further let $\mathcal{W}_{cvx}^{mm}$ be the solution set of* (W-SVM$_\star$) *with $m = d$ achieving objective C_{cvx}. If $\mathcal{W}_\star^{mm} \cap \mathcal{W}_{cvx}^{mm} \neq \emptyset$, then $C_\star = C_{cvx}$ and $\mathcal{W}_\star^{mm} \subseteq \mathcal{W}_{cvx}^{mm}$. Also, if the elements of $\mathcal{W}_{cvx}^{mm}$ have rank at most m, then, $\mathcal{W}_\star^{mm} = \mathcal{W}_{cvx}^{mm}$.*

3.4 Global Convergence of Gradient Descent

In this section, we will establish conditions that guarantee the global convergence of GD. Concretely, we will investigate when GD solution selects the *optimal token within each input sequence* through the softmax nonlinearity and coincides with the solution of the RP. Section 3.5 will complement this with showing that self-attention can more generally converge to locally-optimal max-margin directions. We identify the following conditions as provable catalysts for global convergence: (I) Initial gradient direction is favorable; (II) Overparameterization, i.e. d is appropriately large.

3.4.1 Properties of optimization landscape

We start by establishing some fundamental properties of Objectives (W-ERM) and (KQ-ERM).

Lemma 3.4 *Under Assumption 3.1, $\nabla\mathcal{L}(\boldsymbol{W})$, $\nabla_{\boldsymbol{W}_k}\mathcal{L}(\boldsymbol{W}_k, \boldsymbol{W}_q)$, and $\nabla_{\boldsymbol{W}_q}\mathcal{L}(\boldsymbol{W}_k, \boldsymbol{W}_q)$ are $L_{\boldsymbol{W}}$, $L_{\boldsymbol{W}_k}$, $L_{\boldsymbol{W}_q}$–Lipschitz continuous, respectively, where $a_i = \|\boldsymbol{v}\| \ \|\boldsymbol{z}_i\|^2\|\boldsymbol{X}_i\|^3$, $b_i = M_0\|\boldsymbol{v}\| \ \|\boldsymbol{X}_i\| + 3M_1$ for all $i \in [n]$,*

$$L_{\boldsymbol{W}} := \frac{1}{n}\sum_{i=1}^{n} a_i b_i, \quad L_{\boldsymbol{W}_k} := \|\boldsymbol{W}_q\| L_{\boldsymbol{W}}, \quad \text{and} \quad L_{\boldsymbol{W}_q} := \|\boldsymbol{W}_k\| L_{\boldsymbol{W}}. \tag{3.3}$$

The next assumption will play an important role ensuring the attention layer has a benign optimization landscape.

Assumption 3.2 *Optimal tokens' indices $(opt_i)_{i=1}^n$ are unique and one of the following conditions on the tokens holds:*

A1 *All tokens are support vectors, i.e., $(\boldsymbol{x}_{iopt_i} - \boldsymbol{x}_{it})^\top \boldsymbol{W}^{mm}\boldsymbol{z}_i = 1$ for all $t \neq opt_i$ and $i \in [n]$.*

A2 *The tokens' scores, as defined in Definition 3.1, satisfy* $\gamma_{it} = \gamma_{i\tau} < \gamma_{iopt_i}$, *for all* $t, \tau \neq opt_i$ *and* $i \in [n]$.

Assumption **A1** is directly linked to overparameterization and holds practical significance. In scenarios such as classical SVM classification, where the goal is to separate labels, overparameterization leads to the situation where *all training points become support vectors*. Consequently, the SVM solution aligns with the least-squares minimum-norm interpolation, a concept established in [80, 138] under broad statistical contexts. Assumption **A1** represents an analogous manifestation of this condition. Therefore, in cases involving realistic data distributions with sufficiently large d, we expect the same phenomena to persist, causing the SVM solution $\boldsymbol{W}^{\mathrm{mm}}$ to coincide with (W-SVM).

Drawing on insights from [80, Theorem 1] and our Theorem 3.1, we expect that the necessary degree of overparameterization remains moderate. Specifically, in instances where input sequences follow an independent and identically distributed (IID) pattern and tokens exhibit IID isotropic distributions, we posit that $d \gtrsim (T+n)\log(T+n)$ will suffice. More generally, the extent of required overparameterization will be contingent on the covariance of tokens [14, 138] and the distribution characteristics of input sequences [199].

Assumption **A2** stipulates that non-optimal tokens possess identical scores which constitutes a relatively stringent assumption that we will subsequently relax. Under Assumption 3.2, we establish that when optimization problem (W-ERM) is trained using GD, the norm of parameters will diverge.

Theorem 3.3 *Suppose Assumption 3.1 on the loss function* ℓ *and Assumption 3.2 on the tokens hold.*

T1 *For all* $\boldsymbol{W} \in \mathbb{R}^{d\times d}$, *the training loss* (W-ERM) *obeys* $\langle \nabla \mathcal{L}(\boldsymbol{W}), \boldsymbol{W}^{mm} \rangle < 0$.

T2 *Algorithm W-GD with the stepsize* $\eta \leq 1/L_{\boldsymbol{W}}$ *and any starting point* $\boldsymbol{W}(0)$ *satisfies* $\lim_{k\to\infty} \|\boldsymbol{W}(k)\|_F = 0$.

Proof Sketch. Let $\bar{\boldsymbol{h}}_i = \boldsymbol{X}_i \boldsymbol{W}^{\mathrm{mm}} \boldsymbol{z}_i$, $\gamma_i = y_i \cdot \boldsymbol{X}_i \boldsymbol{v}$, $\boldsymbol{h}_i = \boldsymbol{X}_i \boldsymbol{W} \boldsymbol{z}_i$, and $\boldsymbol{s}_i = \mathbb{S}(\boldsymbol{h}_i)$. We have

$$\begin{aligned} \langle \nabla \mathcal{L}(\boldsymbol{W}), \boldsymbol{W}^{\mathrm{mm}} \rangle &= \frac{1}{n} \sum_{i=1}^{n} \ell' \left(\gamma_i^\top \mathbb{S}(\boldsymbol{h}_i) \right) \cdot \left\langle \boldsymbol{X}_i^\top \mathbb{S}'(\boldsymbol{h}_i) \gamma_i \boldsymbol{z}_i^\top, \boldsymbol{W}^{\mathrm{mm}} \right\rangle \\ &= \frac{1}{n} \sum_{i=1}^{n} \ell'(\gamma_i^\top \mathbb{S}(\boldsymbol{h}_i)) \cdot \left(\bar{\boldsymbol{h}}_i^\top \mathrm{diag}(\boldsymbol{s}_i) \gamma_i - \bar{\boldsymbol{h}}_i^\top \boldsymbol{s}_i \boldsymbol{s}_i^\top \gamma_i \right). \end{aligned} \tag{3.4}$$

Let the equal score condition (Assumption **A2**) hold, i.e., $\gamma = \gamma_2 = \ldots = \gamma_T$ be a constant, and let γ_1 and $\bar{\boldsymbol{h}}_1$ denote the largest indices of γ and $\bar{\boldsymbol{h}}$. For any vector $\boldsymbol{s}$ with $\sum_{t\in[T]} \boldsymbol{s}_t = 1$

and $\boldsymbol{s}_t > 0$, we aim to show that $\bar{\boldsymbol{h}}^\top \mathrm{diag}(\boldsymbol{s})\gamma - \bar{\boldsymbol{h}}^\top \boldsymbol{s}\boldsymbol{s}^\top \gamma > 0$. We demonstrate this as follows:

$$\begin{aligned}
\bar{\boldsymbol{h}}^\top \mathrm{diag}(\boldsymbol{s})\gamma - \bar{\boldsymbol{h}}^\top \boldsymbol{s}\boldsymbol{s}^\top \gamma &= \sum_{t=1}^{T} \bar{\boldsymbol{h}}_t \gamma_t \boldsymbol{s}_t - \sum_{t=1}^{T} \bar{\boldsymbol{h}}_t \boldsymbol{s}_t \sum_{t=1}^{T} \gamma_t \boldsymbol{s}_t \\
&= \bar{\boldsymbol{h}}_1(\gamma_1 - \gamma)\boldsymbol{s}_1(1 - \boldsymbol{s}_1) - (\gamma_1 - \gamma)\boldsymbol{s}_1 \sum_{t\geq 2}^{T} \bar{\boldsymbol{h}}_t \boldsymbol{s}_t \\
&= (\gamma_1 - \gamma)(1 - \boldsymbol{s}_1)\boldsymbol{s}_1 \left[\bar{\boldsymbol{h}}_1 - \frac{\sum_{t\geq 2}^{T} \bar{\boldsymbol{h}}_t \boldsymbol{s}_t}{\sum_{t\geq 2}^{T} \boldsymbol{s}_t} \right] \\
&\geq (\gamma_1 - \gamma)(1 - \boldsymbol{s}_1)\boldsymbol{s}_1 (\bar{\boldsymbol{h}}_1 - \max_{t\geq 2} \bar{\boldsymbol{h}}_t). \qquad (3.5)
\end{aligned}$$

To proceed, define

$$\boldsymbol{\gamma}^i_{\mathrm{gap}} = \gamma_{i\mathrm{opt}_i} - \max_{t\neq \mathrm{opt}_i} \gamma_{it} \quad \text{and} \quad \bar{h}^i_{\mathrm{gap}} = \bar{\boldsymbol{h}}_{i\mathrm{opt}_i} - \max_{t\neq \mathrm{opt}_i} \bar{\boldsymbol{h}}_{it}.$$

With these, we obtain

$$\bar{\boldsymbol{h}}_i^\top \mathrm{diag}(\boldsymbol{s}_i)\gamma_i - \bar{\boldsymbol{h}}_i^\top \boldsymbol{s}_i \boldsymbol{s}_i^\top \gamma_i \geq \boldsymbol{\gamma}^i_{\mathrm{gap}} \bar{h}^i_{\mathrm{gap}} (1 - \boldsymbol{s}_{i\mathrm{opt}_i}) \boldsymbol{s}_{i\mathrm{opt}_i}.$$

Note that

$$\begin{aligned}
&\bar{h}^i_{\mathrm{gap}} = \min_{t\neq \mathrm{opt}_i} \ (\boldsymbol{x}_{i\mathrm{opt}_i} - \boldsymbol{x}_{it})^\top \boldsymbol{W}^{\mathrm{mm}} \boldsymbol{z}_i \geq 1, \\
&\boldsymbol{\gamma}^i_{\mathrm{gap}} = \min_{t\neq \mathrm{opt}_i} \ \gamma_{i\mathrm{opt}_i} - \gamma_{it} > 0, \\
&\boldsymbol{s}_{i\mathrm{opt}_i}(1 - \boldsymbol{s}_{i\mathrm{opt}_i}) > 0.
\end{aligned}$$

Hence,

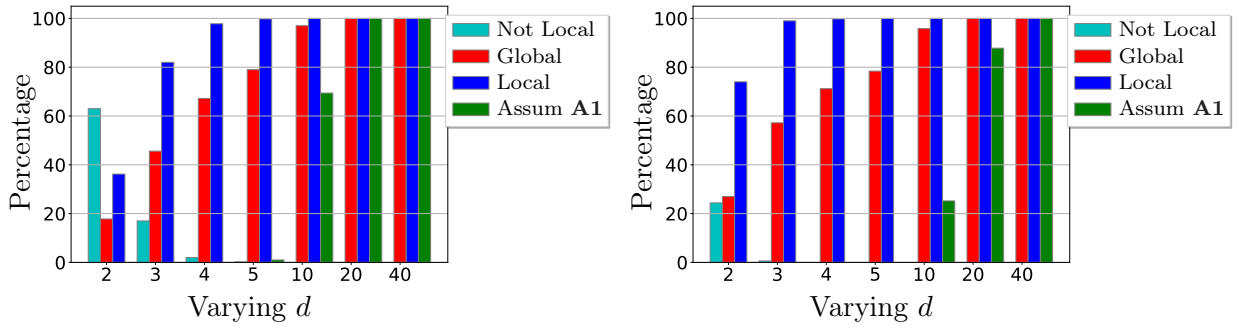
$$\min_{i\in[n]} \left\{ \bar{\boldsymbol{h}}_i^\top \mathrm{diag}(\boldsymbol{s}_i)\gamma_i - \bar{\boldsymbol{h}}_i^\top \boldsymbol{s}_i \boldsymbol{s}_i^\top \gamma_i \right\} > 0. \qquad (3.6)$$

Further, by Assumption 3.1, $\ell'_i < 0$, ℓ' is continuous and the domain is bounded, the maximum is attained and negative, and thus

$$\max_x \ell'(x) < 0. \qquad (3.7)$$

Hence, using (3.6) and (3.7) in (3.4), we obtain $\langle \nabla \mathcal{L}(\boldsymbol{W}), \boldsymbol{W}^{\mathrm{mm}} \rangle < 0$. In the scenario that Assumption **A1** holds (all tokens are support), $\bar{\boldsymbol{h}}_t = \boldsymbol{x}_{it}^\top \boldsymbol{W}^{\mathrm{mm}} \boldsymbol{z}_i$ is constant for all $t \geq 2$. Hence, following similar steps as in (3.5) completes the proof of **T1**.

Under Assumption 3.1, if the step size $\eta \leq 1/L_{\boldsymbol{W}}$ (where $L_{\boldsymbol{W}}$ is the Lipschitz constant), then for any initial point $\boldsymbol{W}(0)$, Algorithm W-GD satisfies the descent property



(a) $\boldsymbol{W}$-parameterization (b) $(\boldsymbol{W}_k, \boldsymbol{W}_q)$-parameterization

Figure 3.4: Convergence behavior of GD for $\boldsymbol{W}$- and $(\boldsymbol{W}_k, \boldsymbol{W}_q)$-parameterizations. Percentage of different convergence types of GD when training cross-attention weights **(a)**: $\boldsymbol{W}$ or **(b)**: $(\boldsymbol{W}_k, \boldsymbol{W}_q)$ with varying d. In both figures, red, blue, and teal bars represent the percentages of Global, Local (including Global), and Not Local convergence, respectively. The green bar corresponds to Assumption **A1** where all tokens act as support vectors. Larger overparameterization (d) relates to a higher percentage of globally-optimal SVM convergence.

$\mathcal{L}(\boldsymbol{W}(k+1)) - \mathcal{L}(\boldsymbol{W}(k)) \leq -\frac{\eta}{2}\|\nabla\mathcal{L}(\boldsymbol{W}(k))\|_F^2$, for all $k \geq 0$ [186, Lemma 6], implying $\|\nabla\mathcal{L}(\boldsymbol{W}(k))\|_F^2 \to 0$.

Furthermore, under Assumptions 3.1 and 3.2, for all $\boldsymbol{W} \in \mathbb{R}^{d\times l}$, the training loss (W-ERM) satisfies $\langle \nabla\mathcal{L}(\boldsymbol{W}), \boldsymbol{W}^{mm}\rangle < 0$, which by **T1** holds for all $\boldsymbol{W} \in \mathbb{R}^{d\times d}$. Therefore, $\langle \nabla\mathcal{L}(\boldsymbol{W}), \boldsymbol{W}^{mm}\rangle$ cannot be zero for any finite $\boldsymbol{W}$, contradicting the $\lim_{k\to\infty} \|\nabla\mathcal{L}(\boldsymbol{W}(k))\|_F^2 = 0$. This implies $\|\boldsymbol{W}(k)\|_F \to \infty$, proving **T2**.

The feasibility of SVM (per Theorem 3.1) is a necessary condition for the convergence of GD to the $\boldsymbol{W}^{\text{mm}}$ direction. However, it does not inform us about the optimization landscape. Two additional criteria are essential for convergence: the absence of stationary points $\nabla\mathcal{L}(\boldsymbol{W}) = 0$ and divergence of parameter norm to infinity. Theorem 3.3 above precisely guarantees both of these criteria under Assumption 3.2.

3.4.2 Provable global convergence of 1-layer transformer

In the quest for understanding the global convergence of a 1-layer transformer, [186, Theorem 2] provided the first global convergence analysis of the attention in a restrictive scenario where $n = 1$ and under the Assumption **A2**. Here, we present two new conditions for achieving global convergence towards the max-margin direction $\boldsymbol{W}^{\text{mm}}$ based on: **(I)** the initial gradient direction, and **(II)** over-parameterization. For the first case, we provide precise theoretical guarantees. For the second, we offer strong empirical evidence, supported by Theorem 3.3,

and a formal conjecture described in Section 3.5.2. We remind the reader that we optimize the attention weights $\boldsymbol{W}$ while fixing the linear prediction head $h(\boldsymbol{x}) = \boldsymbol{v}^\top \boldsymbol{x}$. This approach avoids trivial convergence guarantees where an over-parameterized $h(\cdot)$—whether it is a linear model or an MLP—can be used to achieve zero training loss without providing any meaningful insights into the functionality of the attention mechanism.

(I) Global convergence under good initial gradient. To ensure global convergence, we identify an assumption that prevents GD from getting trapped at suboptimal tokens that offer no scoring advantage compared to other choices. To establish a foundation for providing the convergence of GD to the globally optimal solution $\boldsymbol{W}^{\mathrm{mm}}$, we need the following definitions. For parameters $\mu > 0$ and $R > 0$, define

$$\bar{\mathcal{C}}_{\mu,R} := \left\{ \|\boldsymbol{W}\|_F \geq R \;\middle|\; \left\langle (\boldsymbol{x}_{i\mathrm{opt}_i} - \boldsymbol{x}_{it})\boldsymbol{z}_i^\top, \frac{\boldsymbol{W}}{\|\boldsymbol{W}\|_F} \right\rangle \geq \mu \quad \text{for all} \quad t \neq \mathrm{opt}_i,\ i \in [n] \right\}. \tag{3.8}$$

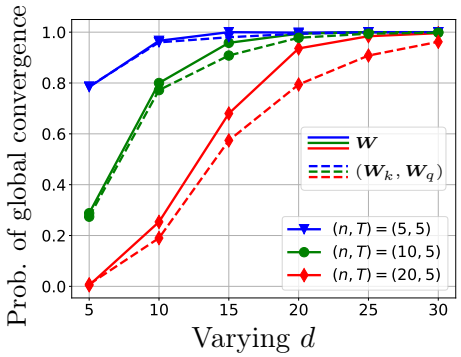
This is the set of all $\boldsymbol{W}$s that separate the optimal tokens from the rest with margin μ. We will show that, for any $\mu > 0$, the optimization landscape of this set is favorable and, if the updates remain in the set, the gradient descent will maximize the margin and find $\boldsymbol{W}^{\mathrm{mm}}$.

Assumption 3.3 (First GD Step is Separating) *For some $\iota > 0$ and all $t \neq opt_i,\ i \in [n]$: $(\boldsymbol{x}_{it} - \boldsymbol{x}_{iopt_i})^\top \nabla \mathcal{L}(0) \boldsymbol{z}_i \geq \iota$.*

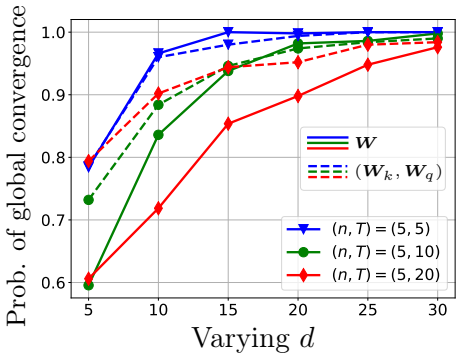
Theorem 3.4 *Suppose Assumption 3.1 on the loss function ℓ and Assumption 3.3 on the initial gradient hold.*

- *For any $\mu > 0$, there exists $R > 0$ such that $\bar{\mathcal{C}}_{\mu,R}$ does not contain any stationary points.*
- *Fix any $\mu \in (0, \iota/\|\nabla \mathcal{L}(0)\|_F)$. Consider GD iterations with $\boldsymbol{W}(0) = 0$, $\boldsymbol{W}(1) = -R\nabla \mathcal{L}(0)/\|\nabla \mathcal{L}(0)\|_F$, and $\boldsymbol{W}(k+1) = \boldsymbol{W}(k) - \eta \nabla \mathcal{L}(\boldsymbol{W}(k))$ for $k \geq 1$, where $\eta \leq 1/L_{\boldsymbol{W}}$ and R sufficiently large. If all iterates remain within $\bar{\mathcal{C}}_{\mu,R}$, then $\lim_{k\to\infty} \|\boldsymbol{W}(k)\|_F = \infty$ and $\lim_{k\to\infty} \frac{\boldsymbol{W}(k)}{\|\boldsymbol{W}(k)\|_F} = \frac{\boldsymbol{W}^{mm}}{\|\boldsymbol{W}^{mm}\|_F}$.*

Note that the second result of Theorem 3.4, i.e., the divergence of parameter norm to infinity and directional convergence requires that all GD iterations remain within $\bar{\mathcal{C}}_{\mu,R}$ defined in (3.8). In Appendix B.3.2, we show that if for all $\boldsymbol{W} \in \bar{\mathcal{C}}_{\mu,R}(\boldsymbol{W}^{\mathrm{mm}})$, $\min_{i\in[n]} \left\langle (\boldsymbol{x}_{i\mathrm{opt}_i} - \boldsymbol{x}_{it})\boldsymbol{z}_i^\top, \boldsymbol{W} - \eta \nabla \mathcal{L}(\boldsymbol{W}) \right\rangle - \min_{i\in[n]} \left\langle (\boldsymbol{x}_{i\mathrm{opt}_i} - \boldsymbol{x}_{it})\boldsymbol{z}_i^\top, \boldsymbol{W} \right\rangle$ is lower bounded by $(2\eta\mu/\|\boldsymbol{W}^{\mathrm{mm}}\|_F) \langle -\nabla \mathcal{L}(\boldsymbol{W}), \boldsymbol{W}^{\mathrm{mm}} \rangle$, then all GD iterations $\boldsymbol{W}(k)$ remain within $\bar{\mathcal{C}}_{\mu,R}$.



(a) Global convergence for varying n, d



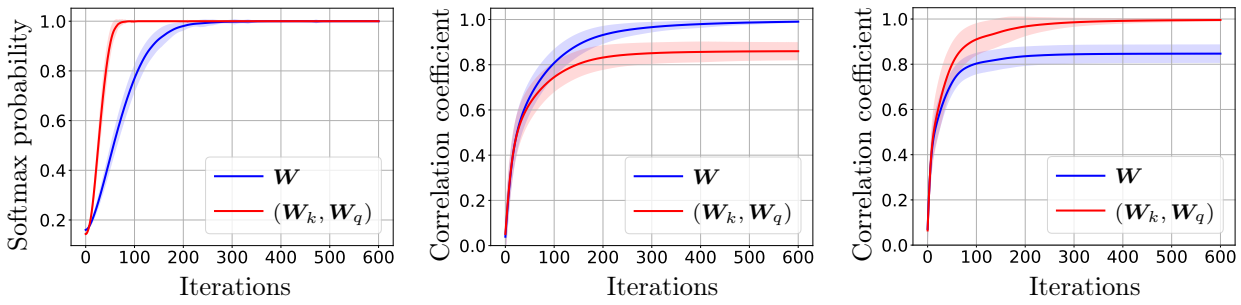
(b) Global convergence for varying T, d

Figure 3.5: Global convergence behavior of GD when training cross-attention weights $\boldsymbol{W}$ (solid) or $(\boldsymbol{W}_k, \boldsymbol{W}_q)$ (dashed) with random data. The blue, green, and red curves represent the probabilities of global convergence for **(a)**: fixing $T = 5$ and varying $n \in \{5, 10, 20\}$ and **(b)**: fixing $n = 5$ and varying $T \in \{5, 10, 20\}$. Results demonstrate that for both attention models, as d increases (due to over-parameterization), attention weights tend to select optimal tokens $(\text{opt}_i)_{i=1}^n$.

While this condition may appear complicated, it is essentially a tight requirement for updates to remain within $\bar{\mathcal{C}}_{\mu,R}$. Finally, it is worth mentioning that, if a stronger correlation condition between initial gradient $\nabla\mathcal{L}(0)$ and $\boldsymbol{W}^{\text{mm}}$ holds, one can also prove that updates remain within a tighter cone around $\boldsymbol{W}^{\text{mm}}$ through ideas developed in Theorem 3.5 by landing $\boldsymbol{W}(1)$ around $\boldsymbol{W}^{\text{mm}}$ direction. However, we opt to state here the result for the milder condition $\bar{\mathcal{C}}_{\mu,R}$.

(II) Global convergence via overparameterization. In the context of standard neural networks, overparameterization has been recognized as a pivotal factor for the global convergence of GD [5, 49, 119, 150]. However, conventional approaches like the neural tangent kernel [83] do not seamlessly apply to our scenario, given our assumption on fixed h and the avoidance of achieving zero loss by trivially fitting h. Furthermore, even when we train h to achieve zero loss, it doesn't provide substantial insights into the implicit bias of the attention weights $\boldsymbol{W}$. Conversely, Theorem 3.3 illustrates the benefits of over-parameterization in terms of convergence. Considering that Assumption **A1** is anticipated to hold as the dimension d increases, the norm of the GD solution is bound to diverge to infinity. This satisfies a prerequisite for converging towards the globally-optimal SVM direction $\boldsymbol{W}^{\text{mm}}$.

The trend depicted in Figure 3.4, where the percentage of global convergence (red bars) approaches 100% and Assumption **A1** holds with higher probability (green bars) as d grows, reinforces this insight. Specifically, green bars representing the percentage of the scenarios



(a) Evolution of softmax prob. (b) Corr. coef. of GD and $\boldsymbol{W}_{\boldsymbol{\alpha}}^{\mathrm{mm}}$ (c) Corr. coeff. of GD and $\boldsymbol{W}_{\star,\boldsymbol{\alpha}}^{\mathrm{mm}}$

Figure 3.6: Local convergence behaviour of GD when training cross-attention weights $\boldsymbol{W}$ (blue) or $(\boldsymbol{W}_k, \boldsymbol{W}_q)$ (red) with random data: **(a)** displays the largest entry of the softmax outputs averaged over the dataset; **(b&c)** display the Pearson correlation coefficients of GD trajectories and the SVM solutions **(b)** with the Frobenius norm objective $\boldsymbol{W}_{\boldsymbol{\alpha}}^{\mathrm{mm}}$ (solution of (W-SVM)) and **(c)** with the nuclear norm objective $\boldsymbol{W}_{\star,\boldsymbol{\alpha}}^{\mathrm{mm}}$ (solution of (W-SVM$_\star$)). These demonstrate the Frobenius norm bias of $\boldsymbol{W}(k)$ and the nuclear norm bias of $\boldsymbol{W}_k(k)\boldsymbol{W}_q(k)^\top$.

where almost all tokens act as support vectors (Assumption **A1**). In both experiments, and for each chosen d value, a total of 500 random instances are conducted under the conditions of $n = T = 5$. The outcomes are reported in terms of the percentages of Not Local, Local, and Global convergence, represented by the teal, blue, and red bars, respectively. We validate Assumption **A1** as follows: Given a problem instance, we compute the average margin over all non-optimal tokens of all inputs and declare that problem satisfies Assumption **A1**, if the average margin is below 1.1 (where 1 is the minimum).

Furthermore, the observations in Figure 3.5 regarding the percentages of achieving global convergence reaching 100 with larger d reaffirm that overparameterization leads the attention weights to converge directionally towards the optimal max-margin direction outlined by (W-SVM) and (W-SVM$_\star$).

In the upcoming section, we will introduce locally-optimal directions, to which GD can be proven to converge when appropriately initialized. We will then establish a condition that ensures the *globally-optimal direction is the sole viable locally-optimal direction*. This culmination will result in a formal conjecture detailed in Section 3.5.2.

3.5 Local Convergence of 1-Layer Transformer

So far, we have primarily focused on the convergence to the global direction dictated by (W-SVM). In this section, we investigate and establish the local directional convergence of GD as well as RP.

3.5.1 Local convergence of gradient descent

To proceed, we introduce locally-optimal directions by adapting Definition 2 of [186].

Definition 3.2 (Support Indices and Locally-Optimal Direction) *Fix token indices* $\boldsymbol{\alpha} = (\alpha_i)_{i=1}^n$. *Solve* (W-SVM) *with* $(opt_i)_{i=1}^n$ *replaced with* $\boldsymbol{\alpha} = (\alpha_i)_{i=1}^n$ *to obtain* $\boldsymbol{W}_{\boldsymbol{\alpha}}^{mm}$. *Consider the set* $\mathcal{T}_i \subset [T]$ *such that* $(\boldsymbol{x}_{i\alpha_i} - \boldsymbol{x}_{it})^\top \boldsymbol{W}_{\boldsymbol{\alpha}}^{mm} \boldsymbol{z}_i = 1$ *for all* $t \in \mathcal{T}_i$. *We refer to* $(\mathcal{T}_i)_{i=1}^n$ *as the support indices of* $\boldsymbol{\alpha}$. *Additionally, if for all* $i \in [n]$ *and* $t \in \mathcal{T}_i$ *scores per Definition 3.1 obey* $\gamma_{i\alpha_i} > \gamma_{it}$, *indices* $\boldsymbol{\alpha} = (\alpha_i)_{i=1}^n$ *are called* locally-optimal *and* $\boldsymbol{W}_{\boldsymbol{\alpha}}^{mm}$ *is called a* locally-optimal direction.

In words, the concept of local optimality requires that the selected tokens denoted as $\boldsymbol{\alpha}$ should have scores that are higher than the scores of their neighboring tokens referred to as support indices. It is important to observe that the tokens defined as $\text{opt} = (\text{opt}_i)_{i=1}^n$, which we term as the optimal tokens, inherently satisfy the condition of local optimality. Moving forward, we will provide Theorem 3.5 which establishes that when the process of GD is initiated along a direction that is locally optimal, it gradually converges in that particular direction, eventually aligning itself with $\boldsymbol{W}_{\boldsymbol{\alpha}}^{\text{mm}}$. This theorem immediately underscores the fact that if there exists a direction of local optimality (apart from the globally optimal direction $\boldsymbol{W}^{\text{mm}}$), then when GD commences from any arbitrary starting point, it does not achieve global convergence towards $\boldsymbol{W}^{\text{mm}}$.

To provide a basis for discussing local convergence of GD, we establish a cone centered around $\boldsymbol{W}_{\boldsymbol{\alpha}}^{\text{mm}}$ using the following construction. For parameters $\mu \in (0,1)$ and $R > 0$, we define $\mathcal{C}_{\mu,R}(\boldsymbol{W}_{\boldsymbol{\alpha}}^{\text{mm}})$ as the set of matrices $\boldsymbol{W} \in \mathbb{R}^{d\times d}$ such that $\|\boldsymbol{W}\|_F \geq R$ and the correlation coefficient between $\boldsymbol{W}$ and $\boldsymbol{W}_{\boldsymbol{\alpha}}^{\text{mm}}$ is at least $1-\mu$:

$$\mathcal{C}_{\mu,R}(\boldsymbol{W}_{\boldsymbol{\alpha}}^{\text{mm}}) = \left\{ \|\boldsymbol{W}\|_F \geq R \;\middle|\; \left\langle \frac{\boldsymbol{W}}{\|\boldsymbol{W}\|_F}, \frac{\boldsymbol{W}_{\boldsymbol{\alpha}}^{\text{mm}}}{\|\boldsymbol{W}_{\boldsymbol{\alpha}}^{\text{mm}}\|_F} \right\rangle \geq 1-\mu \right\}. \tag{3.9}$$

Theorem 3.5 *Suppose Assumption 3.1 on the loss* ℓ *holds, and let* $\boldsymbol{\alpha} = (\alpha_i)_{i=1}^n$ *be locally optimal tokens according to Definition 3.2. Let* $\boldsymbol{W}_{\boldsymbol{\alpha}}^{mm}$ *denote the SVM solution obtained via* (W-SVM) *by replacing* $(opt_i)_{i=1}^n$ *with* $\boldsymbol{\alpha} = (\alpha_i)_{i=1}^n$.

- *There exist parameters* $\mu = \mu(\boldsymbol{\alpha}) \in (0,1)$ *and* $R > 0$ *such that* $\mathcal{C}_{\mu,R}(\boldsymbol{W}_{\boldsymbol{\alpha}}^{mm})$ *does not contain any stationary points.*
- *Algorithm W-GD with* $\eta \leq 1/L_{\boldsymbol{W}}$ *and any* $\boldsymbol{W}(0) \in \mathcal{C}_{\mu,R}(\boldsymbol{W}_{\boldsymbol{\alpha}}^{mm})$ *satisfies* $\lim_{k\to\infty} \|\boldsymbol{W}(k)\|_F = \infty$ *and* $\lim_{k\to\infty} \frac{\boldsymbol{W}(k)}{\|\boldsymbol{W}(k)\|_F} = \frac{\boldsymbol{W}_{\boldsymbol{\alpha}}^{mm}}{\|\boldsymbol{W}_{\boldsymbol{\alpha}}^{mm}\|_F}$.

This theorem establishes the existence of positive parameters $\mu = \mu(\boldsymbol{\alpha}) > 0$ and $R > 0$ such that there are no stationary points within $\mathcal{C}_{\mu,R}(\boldsymbol{W}_{\boldsymbol{\alpha}}^{\mathrm{mm}})$. Furthermore, if gradient descent (GD) is initiated within $\mathcal{C}_{\mu,R}(\boldsymbol{W}_{\boldsymbol{\alpha}}^{\mathrm{mm}})$, it will converge in the direction of $\boldsymbol{W}_{\boldsymbol{\alpha}}^{\mathrm{mm}}/\|\boldsymbol{W}_{\boldsymbol{\alpha}}^{\mathrm{mm}}\|$. It is worth mentioning that the stronger Theorem 3.3 (e.g., global absence of stationary points) is applicable whenever all tokens are supported, i.e., $\bar{\mathcal{T}}_i = \emptyset$ for all $i \in [n]$. Figure 3.7 illustrates Theorem 3.5. The blue cone represents $\mathcal{C}_{\mu,R}(\boldsymbol{W}_{\boldsymbol{\alpha}}^{\mathrm{mm}})$, while the orange cone illustrates the cone surrounding the globally optimal token. Gradient descent initiated within $\mathcal{C}_{\mu,R}(\boldsymbol{W}_{\boldsymbol{\alpha}}^{\mathrm{mm}})$ converges in the direction of $\boldsymbol{W}_{\boldsymbol{\alpha}}^{\mathrm{mm}}$.

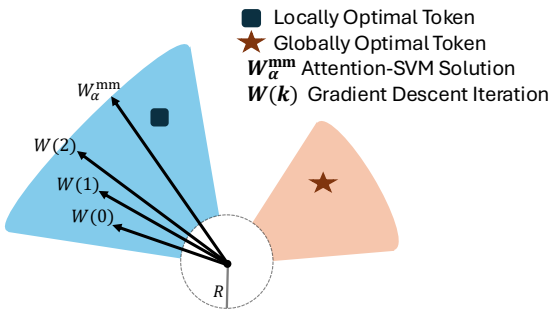


Figure 3.7: The blue cone represents $\mathcal{C}_{\mu,R}(\boldsymbol{W}_{\boldsymbol{\alpha}}^{\mathrm{mm}})$, while the orange illustrates the cone surrounding the globally optimal token. Gradient descent initiated within $\mathcal{C}_{\mu,R}(\boldsymbol{W}_{\boldsymbol{\alpha}}^{\mathrm{mm}})$ converges in the direction of $\boldsymbol{W}_{\boldsymbol{\alpha}}^{\mathrm{mm}}$.

In Figure 3.6, we consider setting where $n = 6$, $T = 8$, and $d = 10$. The displayed results are averaged from 100 random trials. We train cross-attention models with $\boldsymbol{x}_{it}, \boldsymbol{z}_i, \boldsymbol{v} \in \mathbb{R}^d$ randomly sampled from unit sphere, and apply the normalized GD approach with fixed step size $\eta = 0.1$. In Figure 3.6a we calculate the softmax probability via $\frac{1}{n}\sum_{i=1}^{n} \max_{t\in[T]} \mathbb{S}(\boldsymbol{X}_i \tilde{\boldsymbol{W}}(k)\boldsymbol{z}_i)_t$ for either $\tilde{\boldsymbol{W}} = \boldsymbol{W}$ or $\boldsymbol{W}_k\boldsymbol{W}_q^\top$ at each iteration. Both scenarios result in probability 1, which indicates that attention weights succeed in selecting one token per input. Then following Definition 3.2 let $\boldsymbol{\alpha} = (\alpha_i)_{i=1}^n$ be the token indices selected by GD and denote $\boldsymbol{W}_{\star,\boldsymbol{\alpha}}^{\mathrm{mm}}$ as the corresponding SVM solution of (W-SVM$_\star$). Define the correlation coefficient of two matrices as $\texttt{corr\_coef}(\boldsymbol{W}_1, \boldsymbol{W}_2) := \langle \boldsymbol{W}_1, \boldsymbol{W}_2\rangle / \|\boldsymbol{W}_1\|_F \|\boldsymbol{W}_2\|_F$. Figures 3.6b and 3.6c illustrate the correlation coefficients of attention weights ($\boldsymbol{W}(k)$ and $\boldsymbol{W}_k(k)\boldsymbol{W}_q(k)^\top$) with respect to $\boldsymbol{W}_{\boldsymbol{\alpha}}^{\mathrm{mm}}$ and $\boldsymbol{W}_{\star,\boldsymbol{\alpha}}^{\mathrm{mm}}$. The results demonstrate that $\boldsymbol{W}$ ($\boldsymbol{W}_k\boldsymbol{W}_q^\top$) ultimately reaches a 1 correlation with $\boldsymbol{W}_{\boldsymbol{\alpha}}^{\mathrm{mm}}$ ($\boldsymbol{W}_{\star,\boldsymbol{\alpha}}^{\mathrm{mm}}$), which suggests that $\boldsymbol{W}$ ($\boldsymbol{W}_k\boldsymbol{W}_q^\top$) converges in the direction of $\boldsymbol{W}_{\boldsymbol{\alpha}}^{\mathrm{mm}}$ ($\boldsymbol{W}_{\star,\boldsymbol{\alpha}}^{\mathrm{mm}}$). This further validates Theorem 3.5.

3.5.2 Overparameterization conjecture: When local-optimal directions disappear

In Section 3.4 we demonstrated that larger d serves as a catalyst for global convergence to select the optimal indices $\mathrm{opt} = (\mathrm{opt}_i)_{i=1}^n$. However, Section 3.5.1 shows that the convergence can be towards locally-optimal directions rather than global ones. How do we reconcile these? Under what precise conditions, can we expect global convergence?

The aim of this section is gathering these intuitions and stating a concrete conjecture

on the global convergence of the attention layer under geometric assumptions related to overparameterization. To recap, Theorem 3.1 characterizes when (W-SVM) is feasible and Theorem 3.3 characterizes when the parameter norm provably diverges to infinity, i.e. whenever all tokens are support vectors of (W-SVM) (Assumption **A1** holds). On the other hand, this is not sufficient for global convergence, as GD can converge in direction to locally-optimal directions per Section 3.5.1. Thus, to guarantee global convergence, we need to ensure that **globally-optimal direction is the only viable one**. Our next assumption is a fully-geometric condition that precisely accomplishes this.

Assumption 3.4 (There is always an optimal support index) *For any choice of* $\boldsymbol{\alpha} = (\alpha_i)_{i=1}^n$ *with* $\boldsymbol{\alpha} \neq$ *opt when solving* (W-SVM) *with* opt $\leftarrow \boldsymbol{\alpha}$*, there exists* $i \in [n]$ *such that* $\alpha_i \neq opt_i$ *and* $opt_i \in \mathcal{T}_i$*.*

This guarantees that no $\boldsymbol{\alpha} \neq$ opt can be locally-optimal because it has a support index with higher score at the input i with $\alpha_i \neq \text{opt}_i$. Thus, this ensures that global direction $\boldsymbol{W}^{\text{mm}}$ is the unique locally-optimal direction obeying Def. 3.2. Finally, note that local-optimality in Def. 3.2 is one-sided: GD can provably converge to locally-optimal directions, while we do not provably exclude the existence of other directions. Yet, Theorem 4 of [186] shows that local RPs (see Section 3.5.4 for details) can only converge to locally-optimal directions for almost all datasets[3]. This and Figure 3.4 provide strong evidence that Def. 3.2 captures all possible convergence directions of GD, and as a consequence, that Assumption 3.4 guarantees that $\boldsymbol{W}^{\text{mm}}$ is the only viable direction to converge.

➡ **Integrating the results and global convergence conjecture.** Combining Assumptions **A1** and 3.4, we have concluded that gradient norm diverges and $\boldsymbol{W}^{\text{mm}}$ is the only viable direction to converge. Thus, we conclude this section with the following **conjecture**: Suppose $\text{opt} = (\text{opt}_i)_{i=1}^n$ are the unique optimal indices with strictly highest score per sequence and Assumptions **A1** and 3.4 hold. Then, for almost all datasets (e.g. add small IID gaussian noise to input features), GD with a proper constant learning rate converges to $\boldsymbol{W}^{\text{mm}}$ of (W-SVM) in direction.

To gain some intuition, consider Figure 3.4: Here, red bars denote the frequency of global convergence whereas green bars denote the frequency of Assumption **A1** holding over random problem instances. In short, this suggests that Assumption **A1** occurs less frequently than global convergence, which is consistent with our conjecture. On the other hand, verifying Assumption 3.4 is more challenging due to its combinatorial nature. A stronger condition that implies Assumption 3.4 is when *all optimal indices* $(opt_i)_{i=1}^n$ *are support vectors of the*

[3]To be precise, they prove this for their version of Def. 3.2, which is stated for an attention model $f(\boldsymbol{p}) = \boldsymbol{v}^\top \boldsymbol{X}^\top \mathbb{S}(\boldsymbol{X}\boldsymbol{W}\boldsymbol{p})$ admitting an analogous SVM formulation.

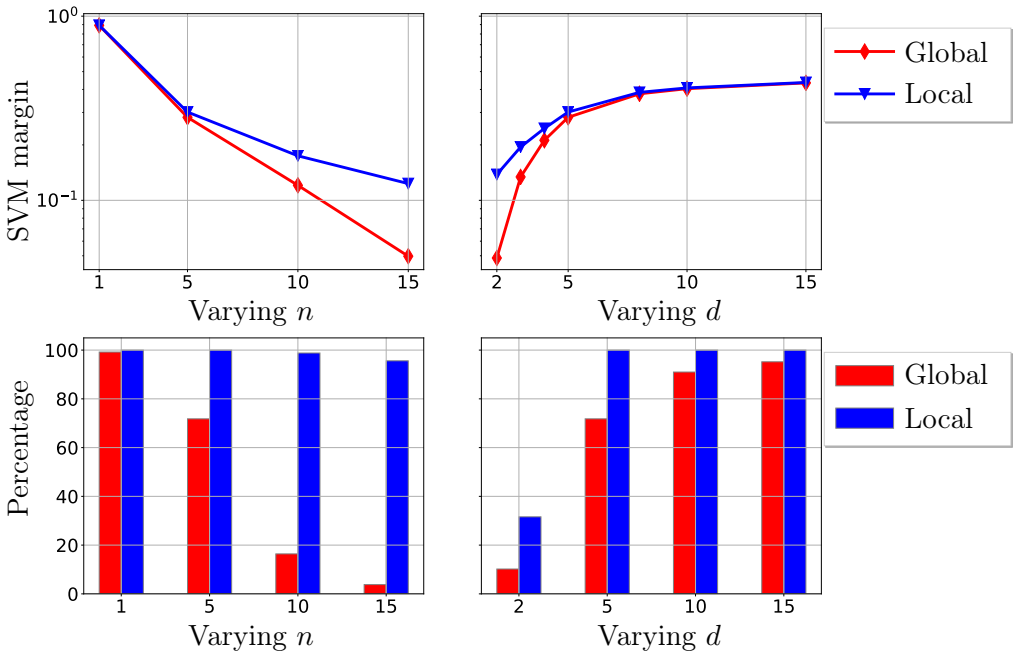


Figure 3.8: Performance of GD convergence and corresponding SVM margin. **Upper:** The SVM margins correspond to globally-optimal (red) and locally-optimal (blue) token indices, denoted as $1/\|\boldsymbol{W}^{\mathrm{mm}}\|_F$ and $1/\|\boldsymbol{W}_{\boldsymbol{\alpha}}^{\mathrm{mm}}\|_F$, respectively. **Lower:** Percentages of global convergence (when $\boldsymbol{\alpha} = \mathrm{opt}$, red) and local convergence (when $\boldsymbol{\alpha} \neq \mathrm{opt}$, blue).

SVM. That is, either $\mathrm{opt}_i = \alpha_i$ *or* $\mathrm{opt}_i \in \mathcal{T}_i$, $\forall\, i \in [n]$. When the data follows a statistical model, this stronger condition could be verified through probabilistic tools building on our earlier "all training points are support vectors" discussion [138]. More generally, we believe a thorough statistical investigation of (W-SVM) is a promising direction for future work.

3.5.3 Investigation on SVM objectives and GD convergence

Until now, we have discussed the global and local convergence performances of gradient descent (GD). Theorem 3.5 suggests that, without specific restrictions on tokens, when training with GD, the attention weight $\boldsymbol{W}$ converges towards $\boldsymbol{W}_{\boldsymbol{\alpha}}^{\mathrm{mm}}$. Here, the selected token indices $\boldsymbol{\alpha} = (\alpha_i)_{i=1}^n$ may not necessarily be identical to $\mathrm{opt} = (\mathrm{opt}_i)_{i=1}^n$. Experiments presented in Figures 3.4 and 3.5 also support this observation. In this section, we focus on scenarios where $\boldsymbol{\alpha} \neq \mathrm{opt}$ (e.g., when $\boldsymbol{W}^{\mathrm{mm}}$ is not feasible) and investigate the question: Towards which local direction is GD most likely to converge?

To this goal, in Figure 3.8, we consider SVM margin, which is defined by $1/\|\boldsymbol{W}_{\boldsymbol{\alpha}}^{\mathrm{mm}}\|_F$, and investigate its connection to the convergence performance of GD. On the left, we set $T = d = 5$ and vary n among $1, 5, 10, 15$; on the right, we fix $T = n = 5$ and change d to $2, 5, 10, 15$.

All tokens are randomly generated from the unit sphere. The SVM margins corresponding to the selected tokens $\boldsymbol{\alpha}$ are depicted as blue curves in the upper subfigures, while the SVM margins corresponding to the globally–optimal token indices ($\boldsymbol{\alpha} = \text{opt}$) are shown as red curves. The red and blue bars in the lower subfigures represent the percentages of global and local convergence, respectively. Combining all these findings empirically demonstrates that when the global SVM objective yields a solution with a small margin (i.e., $1/\|\boldsymbol{W}^{\text{mm}}\|_F$ is small, and 0 when global SVM is not feasible), GD tends to converge towards a local direction with a comparatively larger margin.

3.5.4 Guarantees on local regularization path

In this section, we provide a *localized* regularization path analysis for general objective functions. As we shall see, this will also allow us to predict the local solutions of gradient descent described in Section 3.5.1. Let $\diamond$ denote a general norm objective. Given indices $\boldsymbol{\alpha} = (\alpha_i)_{i=1}^n$, consider the formulation

$$\boldsymbol{W}_{\diamond,\alpha}^{\text{mm}} = \underset{\texttt{rank}(W)\le m}{\arg\min} \|\boldsymbol{W}\|_\diamond \quad \text{s.t.} \quad (\boldsymbol{x}_{i\alpha_i} - \boldsymbol{x}_{it})^\top \boldsymbol{W} \boldsymbol{z}_i \ge 1 \quad \text{for all} \quad t \ne \alpha_i,\ i \in [n]. \quad (\diamond\text{-SVM})$$

In this section, since $\diamond$ is clear from the context, we will use the shorthand $\boldsymbol{W}_{\alpha}^{\text{mm}} := \boldsymbol{W}_{\diamond,\alpha}^{\text{mm}}$ and denote the optimal solution set of ($\diamond$-SVM) as $\mathcal{W}^{\text{mm}} := \mathcal{W}_{\diamond,\alpha}^{\text{mm}}$. It is important to note that if the $\diamond$-norm is not strongly convex, $\mathcal{W}^{\text{mm}}$ may not be a singleton. Additionally, when $m = d$, the rank constraint becomes vacuous, and the problem becomes convex. The following result is a slight generalization of Theorem 3.1 and demonstrates that choosing a large d ensures the feasibility of ($\diamond$-SVM) uniformly over all choices of $\boldsymbol{\alpha}$. The proof is similar to that of Theorem 3.1, as provided in Appendix B.1.

Theorem 3.6 *Suppose $d \ge \max(T-1, n)$ and $m = d$. Then, almost all datasets*[4] *$(y_i, \boldsymbol{X}_i, \boldsymbol{z}_i)_{i=1}^n$ – including the self-attention setting with $\boldsymbol{z}_i \leftarrow \boldsymbol{x}_{i1}$ – obey the following: For any choice of indices $\boldsymbol{\alpha} = (\alpha_i)_{i=1}^n \subset [T]$, ($\diamond$-SVM) is feasible, i.e. the attention layer can separate and select indices $\boldsymbol{\alpha}$.*

To proceed, we define the *local regularization path*, which is obtained by solving the $\diamond$-norm-constrained problem over a $\boldsymbol{\alpha}$-dependent cone denoted as $\texttt{cone}_\epsilon(\boldsymbol{\alpha})$. This cone has a simple interpretation: it prioritizes tokens with a lower score than $\boldsymbol{\alpha}$ over tokens with a higher score than $\boldsymbol{\alpha}$. This interpretation sheds light on the convergence towards locally

[4]Here, *"almost all datasets"* means that adding i.i.d. gaussian noise, with arbitrary nonzero variance, to the input features will almost surely result in SVM's feasibility.

optimal directions: lower-score tokens create a barrier for $\boldsymbol{\alpha}$ and prevent optimization from moving towards higher-score tokens.

Definition 3.3 (Low&High Score Tokens and Separating Cone) *Given $\alpha \in [T]$, input sequence $\boldsymbol{X}$ with label y, $h(\cdot) : \mathbb{R}^d \to \mathbb{R}$, and score $\gamma_t = y \cdot h(\boldsymbol{x}_t)$ for all $t \in [T]$, define the low and high score tokens as*

$$\texttt{low}^\alpha(\boldsymbol{X}) := \left\{t \in [T] \;\middle|\; \gamma_t < \gamma_\alpha\right\}, \quad \texttt{high}^\alpha(\boldsymbol{X}) := \left\{t \in [T] - \{\alpha\} \;\middle|\; \gamma_t \geq \gamma_\alpha\right\}.$$

For input $\boldsymbol{X}_i$ and index α_i, we use the shorthand notations $\texttt{low}_i^\alpha$ and $\texttt{high}_i^\alpha$. Finally define $\texttt{cone}_\epsilon(\boldsymbol{\alpha})$ as

$$\texttt{cone}_\epsilon(\boldsymbol{\alpha}) := \left\{\texttt{rank}(\boldsymbol{W}) \leq m \;\middle|\; \min_{i\in[n]} \max_{t\in \texttt{low}_i^\alpha} \min_{\tau\in\texttt{high}_i^\alpha} (\boldsymbol{x}_{it} - \boldsymbol{x}_{i\tau})^\top \boldsymbol{W} \boldsymbol{z}_i \geq \epsilon \|\boldsymbol{W}\|_F\right\}. \tag{3.10}$$

Our next lemma relates this cone definition to locally-optimal directions of Definition 3.2.

Lemma 3.5 *Suppose ($\diamond$-SVM) is feasible. If indices $\boldsymbol{\alpha}$ are locally-optimal, $\boldsymbol{W}_{\boldsymbol{\alpha}}^{mm} \in \texttt{cone}_\epsilon(\boldsymbol{\alpha})$ for all sufficiently small $\epsilon > 0$. Otherwise, $\boldsymbol{W}_{\boldsymbol{\alpha}}^{mm} \notin \texttt{cone}_\epsilon(\boldsymbol{\alpha})$ for all $\epsilon > 0$. Additionally, suppose optimal indices $opt_i \in \arg\max_{t\in[T]} \gamma_{it}$ are unique and set $\boldsymbol{\alpha} \leftarrow opt$. Then, $\texttt{cone}_\epsilon(opt)$ is the set of all rank-$\leq m$ matrices (i.e. global set).*

Lemma 3.5 can be understood as follows: Among the SVM solutions $\boldsymbol{W}_{\boldsymbol{\alpha}}^{\mathrm{mm}}$, only those that are locally optimal demonstrate a barrier of low-score tokens, effectively acting as a protective shield against higher-score tokens. Moreover, in the case of globally optimal tokens (with the highest scores), the global set $\texttt{cone}_\epsilon(\mathrm{opt})$ can be chosen, as they inherently do not require protective measures. The subsequent result introduces our principal theorem, which pertains to the regularization path converging towards the locally-optimal direction over $\texttt{cone}_\epsilon(\boldsymbol{\alpha})$ whenever $\boldsymbol{\alpha}$ is locally optimal.

Theorem 3.7 (Convergence of Local Regularization Path) *Suppose Assumption 3.1 holds. Fix locally-optimal token indices $\boldsymbol{\alpha} = (\alpha_i)_{i=1}^n$ and $R_0, \epsilon > 0$. Consider the norm-constrained variation of (3.10) defined as*

$$\mathcal{C}_{\epsilon,R_0}^\diamond := \texttt{cone}_\epsilon(\boldsymbol{\alpha}) \bigcap \left\{\boldsymbol{W} \;\middle|\; \|\boldsymbol{W}\|_\diamond \geq R_0\right\}.$$

Define local RP as $\bar{\boldsymbol{W}}_R = \min_{\mathcal{C}_{\epsilon,R_0}^\diamond, \|\boldsymbol{W}\|_\diamond \leq R} \mathcal{L}(\boldsymbol{W})$ where $\mathcal{L}(\boldsymbol{W})$ is given by (W-ERM). Let $\mathcal{W}^{mm}$ be the set of minima for ($\diamond$-SVM) and $\Xi_\diamond > 0$ be the associated margin i.e. $\Xi_\diamond = 1/\|\boldsymbol{W}_{\boldsymbol{\alpha}}^{mm}\|_\diamond$. For any sufficiently small $\epsilon > 0$ and sufficiently large $R_0 = \mathcal{O}(1/\epsilon) > 0$, $\lim_{R\to\infty} \texttt{dist}\left(\frac{\bar{\boldsymbol{W}}_R}{R\Xi_\diamond}, \mathcal{W}^{mm}\right) = 0$. Additionally, suppose optimal indices $opt = (opt_i)_{i=1}^n$ are

unique and set $\boldsymbol{\alpha} \leftarrow$ *opt. Then, the same convergence guarantee on regularization path holds by setting* $\mathcal{C}^{\diamond}_{\epsilon,R_0}$ *as the set of rank-*$\leq m$ *matrices.*

Note that when setting $m = d$, the rank constraint is eliminated. Consequently, specializing this theorem to the Frobenius norm aligns it with Theorem 3.5. On the other hand, by assigning $\diamond$ as the nuclear norm and $\boldsymbol{\alpha} \leftarrow$ opt, the global inductive bias of the nuclear norm is recovered, as stated in Theorem 3.2.

We would like to emphasize that both this theorem and Theorem 3.2 are specific instances of Theorem B.2 found in Appendix B.5. It is worth noting that, within this appendix, we establish all regularization path results for sequence-to-sequence classification, along with a general class of *monotonicity-preserving* prediction heads outlined in Assumption B.1. The latter significantly generalizes linear heads, highlighting the versatility of our theory. The following section presents our discoveries concerning general nonlinear heads.

3.6 Toward a More General SVM Equivalence for Nonlinear Prediction Heads

So far, our theory has focused on the setting where the attention layer selects a single optimal token within each sequence. As we have discussed, this is theoretically well-justified under linear head assumption and certain nonlinear generalizations. On the other hand, for arbitrary nonconvex $h(\cdot)$ or multilayer transformer architectures, it is expected that attention will select multiple tokens per sequence. This motivates us to ask:

> ***Q:*** *What is the implicit bias and the form of* $\boldsymbol{W}(k)$ *when the GD solution is composed by multiple tokens?*

In this section, our goal is to derive and verify the generalized behavior of GD. Let $\boldsymbol{o}_i = \boldsymbol{X}_i^\top \boldsymbol{s}_i^{\boldsymbol{W}}$ denote the composed token generated by the attention layer where $\boldsymbol{s}_i^{\boldsymbol{W}} = \mathbb{S}(\boldsymbol{X}_i \boldsymbol{W} \boldsymbol{z}_i)$ are the softmax probabilities corresponding to $\boldsymbol{W}$. Suppose GD trajectory converges to achieve the risk $\mathcal{L}_\star = \min_{\boldsymbol{W}} \mathcal{L}(\boldsymbol{W})$, and the eventual token composition achieving $\mathcal{L}_\star$ is given by

$$\boldsymbol{o}_i^\star = \boldsymbol{X}_i^\top \boldsymbol{s}_i^\star,$$

where $\boldsymbol{s}_i^\star$ are the eventual softmax probability vectors that dictate the token composition. Since attention maps are sparse in practice, we are interested in the scenario where $\boldsymbol{s}_i^\star$ is sparse i.e. it contains some zero entries. This can only be accomplished by letting $\|\boldsymbol{W}\|_F \to \infty$. However, unlike the earlier sections, we wish to allow for arbitrary $\boldsymbol{s}_i^\star$ rather than a one-hot vector which selects a single token.

To proceed, we aim to understand the form of GD solution $\boldsymbol{W}(k)$ responsible for composing $\boldsymbol{o}_i^\star$ via the softmax map $\boldsymbol{s}_i^\star$ as $\|\boldsymbol{W}\|_F \to \infty$. Intuitively, $\boldsymbol{W}(k)$ should be decomposed into two components via

$$\boldsymbol{W}(k) \approx \boldsymbol{W}^{\text{fin}} + \|\boldsymbol{W}(k)\|_F \cdot \bar{\boldsymbol{W}}^{\text{mm}}, \tag{3.11}$$

where $\boldsymbol{W}^{\text{fin}}$ is the finite component and $\bar{\boldsymbol{W}}^{\text{mm}}$ is the directional component with $\|\bar{\boldsymbol{W}}^{\text{mm}}\|_F = 1$. Define the selected set $\mathcal{O}_i \subseteq [T]$ to be the indices $\boldsymbol{s}_{it}^\star \neq 0$ and the masked (i.e. suppressed) set as $\bar{\mathcal{O}}_i = [T] - \mathcal{O}_i$ where softmax entries are zero. In the context of earlier sections, we could also call these the *optimal set* and the *non-optimal set*, respectively.

• **Finite component:** The job of $\boldsymbol{W}^{\text{fin}}$ is to assign nonzero softmax probabilities within each $\boldsymbol{s}_i^\star$. This is accomplished by ensuring that, $\boldsymbol{W}^{\text{fin}}$ induces the probabilities of $\boldsymbol{s}_i^\star$ over $\mathcal{O}_i$ by satisfying the softmax equations

$$\frac{e^{\boldsymbol{x}_{it}^\top \boldsymbol{W}^{\text{fin}} \boldsymbol{z}_i}}{e^{\boldsymbol{x}_{i\tau}^\top \boldsymbol{W}^{\text{fin}} \boldsymbol{z}_i}} = e^{(\boldsymbol{x}_{it} - \boldsymbol{x}_{i\tau})^\top \boldsymbol{W}^{\text{fin}} \boldsymbol{z}_i} = \frac{\boldsymbol{s}_{it}^\star}{\boldsymbol{s}_{i\tau}^\star},$$

for $t, \tau \in \mathcal{O}_i$. Consequently, this $\boldsymbol{W}^{\text{fin}}$ should satisfy the following linear constraints

$$(\boldsymbol{x}_{it} - \boldsymbol{x}_{i\tau})^\top \boldsymbol{W}^{\text{fin}} \boldsymbol{z}_i = \log(\boldsymbol{s}_{it}^\star / \boldsymbol{s}_{i\tau}^\star) \quad \text{for all} \quad t, \tau \in \mathcal{O}_i,\ i \in [n]. \tag{3.12}$$

• **Directional component:** While $\boldsymbol{W}^{\text{fin}}$ creates the composition by allocating the nonzero softmax probabilities, it does not explain sparsity of attention map. This is the role of $\bar{\boldsymbol{W}}^{\text{mm}}$, which is responsible for selecting the selected tokens $\mathcal{O}_i$ and suppressing the masked ones $\bar{\mathcal{O}}_i$ by assigning zero softmax probability to them. To predict direction component, we build on the theory developed in earlier sections. Concretely, there are two constraints $\bar{\boldsymbol{W}}^{\text{mm}}$ should satisfy

1. **Equal similarity over selected tokens:** For all $t, \tau \in \mathcal{O}_i$, we have that $(\boldsymbol{x}_{it} - \boldsymbol{x}_{i\tau})^\top \boldsymbol{W} \boldsymbol{z}_i = 0$. This way, softmax scores assigned by $\boldsymbol{W}^{\text{fin}}$ are not disturbed by the directional component and $\boldsymbol{W}^{\text{fin}} + R \cdot \bar{\boldsymbol{W}}^{\text{mm}}$ will still satisfy the softmax equations (3.12).
2. **Max-margin against masked tokens:** For all $t \in \mathcal{O}_i, \tau \in \bar{\mathcal{O}}_i$, enforce the margin constraint $(\boldsymbol{x}_{it} - \boldsymbol{x}_{i\tau})^\top \boldsymbol{W} \boldsymbol{z}_i \geq 1$ subject to minimum norm $\|\boldsymbol{W}\|_F$.

Combining these yields the following convex generalized SVM formulation

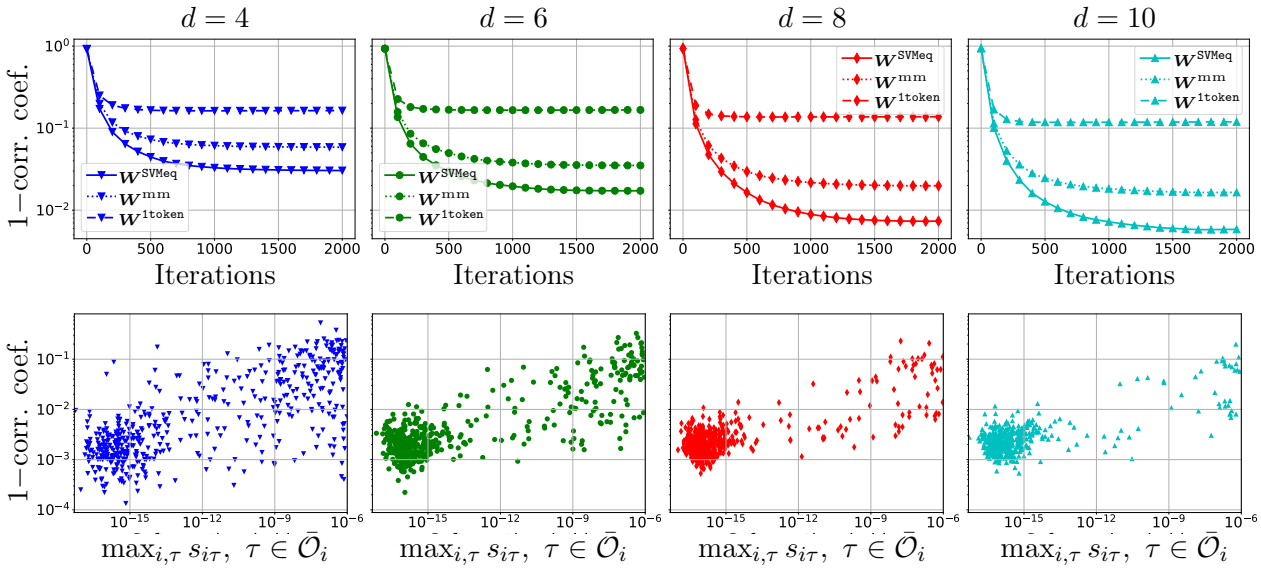


Figure 3.9: Behavior of GD with nonlinear nonconvex prediction head and multi-token compositions. **Upper:** The correlation between GD solution and three distinct baselines: $(\cdots)$ $\boldsymbol{W}^{\texttt{mm}}$ obtained from (Gen-SVM); (—) $\boldsymbol{W}^{\texttt{SVMeq}}$ obtained by calculating $\boldsymbol{W}^{\texttt{fin}}$ and determining the best linear combination $\boldsymbol{W}^{\texttt{fin}} + \gamma \bar{\boldsymbol{W}}^{\texttt{mm}}$ that maximizes correlation with the GD solution; and (- -) $\boldsymbol{W}^{\texttt{1token}}$ obtained by solving (W-SVM) and selecting the highest probability token from the GD solution. **Lower:** Scatterplot of the largest softmax probability over masked tokens (per our $s_{i\tau} \leq 10^{-6}$ criteria) vs correlation coefficient.

$$\boldsymbol{W}^{\texttt{mm}} = \arg\min_{\boldsymbol{W}} \|\boldsymbol{W}\|_F \ \text{ s.t. } \begin{cases} \forall\ t \in \mathcal{O}_i, \tau \in \bar{\mathcal{O}}_i: & (\boldsymbol{x}_{it} - \boldsymbol{x}_{i\tau})^\top \boldsymbol{W} \boldsymbol{z}_i \geq 1, \\ \forall\ t, \tau \in \mathcal{O}_i: & (\boldsymbol{x}_{it} - \boldsymbol{x}_{i\tau})^\top \boldsymbol{W} \boldsymbol{z}_i = 0, \end{cases} \forall\ i \in [n]. \qquad \text{(Gen-SVM)}$$

and set the normalized direction in (3.11) to $\bar{\boldsymbol{W}}^{\texttt{mm}} = \boldsymbol{W}^{\texttt{mm}} / \|\boldsymbol{W}^{\texttt{mm}}\|_F$.

It is important to note that (Gen-SVM) offers a substantial generalization beyond the scope of the previous sections, where the focus was on selecting a single token from each sequence, as described in the main formulation (W-SVM). This broader solution class introduces a more flexible approach to the problem.

We present experiments showcasing the predictive power of the (Gen-SVM) equivalence in nonlinear scenarios. We conducted these experiments on random instances using an MLP denoted as $h(\cdot)$, which takes the form of $\mathbf{1}^\top \texttt{ReLU}(\boldsymbol{x})$. We begin by detailing the preprocessing step and our setup. For the attention SVM equivalence analytical prediction, clear definitions of the selected and masked sets are crucial. These sets include token indices with nonzero and zero softmax outputs, respectively. However, practically, reaching a precisely zero output is not feasible. Hence, we define the selected set as tokens with softmax outputs exceeding 10^{-3},

and the masked set as tokens with softmax outputs below 10^{-6}. We also excluded instances with softmax outputs falling between 10^{-6} and 10^{-3} to distinctly separate the concepts of *selected* and *masked* sets, thereby enhancing the predictive accuracy of the attention SVM equivalence. In addition to the filtering process, we focus on scenarios where the label $y = -1$ exists to enforce *non-convexity* of prediction head $y_i \cdot h(\cdot)$. It is worth mentioning that when all labels are 1, due to the convexity of $y_i \cdot h(\cdot)$, GD tends to select one token per input, and Equations (Gen-SVM) and (W-SVM) yield the same solutions. The results are displayed in Figure 3.9, where $n = 3$, $T = 4$, and d varies within $4, 6, 8, 10$. We conduct 500 random trials for different choices of d, each involving $\boldsymbol{x}_{it}$, $\boldsymbol{z}_i$, and $\boldsymbol{v}$ randomly sampled from the unit sphere. We apply normalized GD with a step size $\eta = 0.1$ and run 2000 iterations for each trial.

• Figure 3.9 (upper) illustrates the correlation evolution between the GD solution and three distinctive baselines: $(\cdots)$ $\boldsymbol{W}^{\text{mm}}$ obtained from (Gen-SVM); (—) $\boldsymbol{W}^{\texttt{SVMeq}}$ obtained by calculating $\boldsymbol{W}^{\text{fin}}$ and determining the best linear combination $\boldsymbol{W}^{\text{fin}} + \gamma \bar{\boldsymbol{W}}^{\text{mm}}$ that maximizes correlation with the GD solution; and (- -) $\boldsymbol{W}^{\texttt{1token}}$ obtained by solving (W-SVM) and selecting the highest probability token from the GD solution. For clearer visualization, the logarithmic scale of correlation misalignment is presented in Figure 3.9. In essence, our findings show that $\boldsymbol{W}^{\texttt{1token}}$ yields unsatisfactory outcomes, whereas $\boldsymbol{W}^{\text{mm}}$ attains a significant correlation coefficient in alignment with our expectations. Ultimately, our comprehensive SVM-equivalence $\boldsymbol{W}^{\texttt{SVMeq}}$ further enhances correlation, lending support to our analytical formulas. It's noteworthy that SVM-equivalence displays higher predictability in a larger d regime (with an average correlation exceeding 0.99). This phenomenon might be attributed to more frequent directional convergence in higher dimensions, with overparameterization contributing to a smoother loss landscape, thereby expediting optimization.

• Figure 3.9 (lower) offers a scatterplot overview of the 500 random problem instances that were solved. The x-axis represents the largest softmax probability over the masked set, denoted as $\max_{i,\tau} s_{i\tau}$ where $\tau \in \bar{\mathcal{O}}_i$. Meanwhile, the y-axis indicates the predictivity of the SVM-equivalence, quantified as $1 - \texttt{corr\_coef}(\boldsymbol{W}, \boldsymbol{W}^{\texttt{SVMeq}})$. From this analysis, two significant observations arise. Primarily, there exists an inverse correlation between softmax probability and SVM-predictivity. This correlation is intuitive, as higher softmax probabilities signify a stronger divergence from our desired *masked set* state (ideally set to 0). Secondly, as dimensionality (d) increases, softmax probabilities over the masked set tend to converge towards the range of 10^{-15} (effectively zero). Simultaneously, attention SVM-predictivity improves, creating a noteworthy correlation.

3.6.1 When does attention select multiple tokens?

In this section, we provide a concrete example where the optimal solution indeed requires combining multiple tokens in a nontrivial fashion. Here, by nontrivial we mean that, we select more than 1 tokens from an input sequence but we don't select all of its tokens. Recall that, for linear prediction head, attention will ideally select the single token with largest score for almost all datasets. Perhaps not surprisingly, this behavior will not persist for nonlinear prediction heads. For instance in Figure 3.9, the GD output $\boldsymbol{W}$ aligned better in direction with $\boldsymbol{W}^{\mathrm{mm}}$ than $\boldsymbol{W}^{\texttt{1token}}$. Specifically, here we prove that if we make the function $h_y(\boldsymbol{x}) := y \cdot h(\boldsymbol{x})$ concave, then optimal softmax map can select multiple tokens in a controllable fashion. $h_y(\boldsymbol{x})$ can be viewed as generalization of the linear score function $y \cdot \boldsymbol{v}^\top \boldsymbol{x}$. In the example below, we induce concavity by incorporating a small $-\lambda\|\boldsymbol{x}\|^2$ term within a linear prediction head and setting $h(\boldsymbol{x}) = \boldsymbol{v}^\top \boldsymbol{x} - \lambda\|\boldsymbol{x}\|^2$ with $y = 1$.

Lemma 3.6 *Given $\boldsymbol{v} \in \mathbb{R}^d$, recall the score vector $\gamma = \boldsymbol{X}\boldsymbol{v}$. Without losing generality, assume γ is non-increasing. Define the vector of score gaps $\gamma^{gap} \in \mathbb{R}^{T-1}$ with entries $\gamma_t^{gap} = \gamma_t - \gamma_{t+1}$. Suppose all tokens within the input sequence are orthonormal and for some $\tau \geq 2$, we have that*

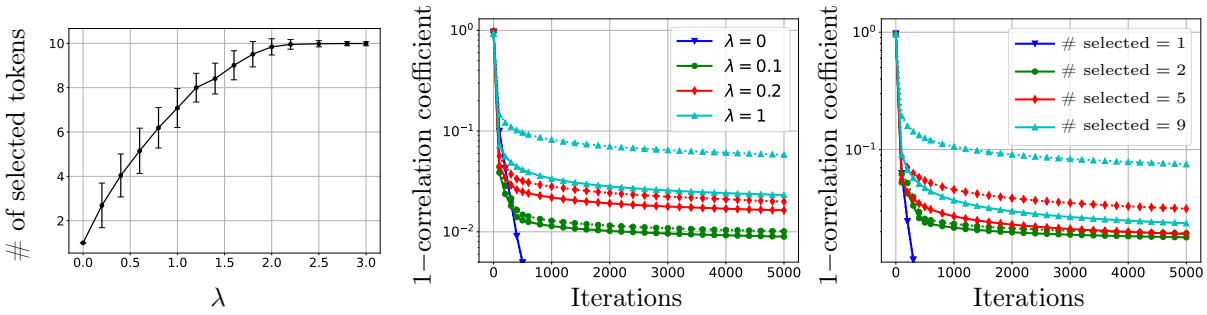
$$\tau\gamma_\tau^{gap}/2 > \gamma_1^{gap}. \tag{3.13}$$

Set $h(\boldsymbol{x}) = \boldsymbol{v}^\top \boldsymbol{x} - \lambda\|\boldsymbol{x}\|^2$ where $\tau\gamma_\tau^{gap}/2 > \lambda > \gamma_1^{gap}$, $\ell(x) = -x$, and $y = 1$. Let $\boldsymbol{\Delta}_T$ denote the T-dimensional simplex. Define the unconstrained softmax optimization associated to the objective h where we make $\boldsymbol{s} := \mathbb{S}(\boldsymbol{X}\boldsymbol{W}\boldsymbol{z})$ a free variable, namely,

$$\min_{\boldsymbol{s}\in\boldsymbol{\Delta}_T} \ell(h(\boldsymbol{X}\boldsymbol{s})) = \min_{\boldsymbol{s}\in\boldsymbol{\Delta}_T} \lambda\|\boldsymbol{X}^\top \boldsymbol{s}\|^2 - \boldsymbol{v}^\top \boldsymbol{X}^\top \boldsymbol{s}. \tag{3.14}$$

Then, the optimal solution $\boldsymbol{s}^\star$ contains at least 2 and at most τ nonzero entries.

Figure 3.10 presents experimental findings concerning Lemma 3.6 across random problem instances. For this experiment, we set $n = 1$, $T = 10$, and $d = 10$. The results are averaged over 100 random trials, with each trial involving the generation of randomly orthonormal vectors $\boldsymbol{x}_{1t}$ and the random sampling of vector $\boldsymbol{v}$ from the unit sphere. Similar to the processing step in Figure 3.9, and following Figure 3.9 (lower) which illustrates that smaller softmax outputs over masked sets correspond to higher correlation coefficients, we define the selected and masked token sets. Specifically, tokens with softmax outputs $> 10^{-3}$ are considered selected, while tokens with softmax outputs $< 10^{-8}$ are masked. Instances with softmax outputs between 10^{-8} and 10^{-3} are filtered out.



(a) λ vs # selected tokens (b) λ vs correlation coefficient (c) # selected vs corr. coef.

Figure 3.10: Behavior of GD when selecting multiple tokens. **(a)** The number of selected tokens increases with λ. **(b)** Predictivity of attention SVM solutions for varying λ; Dotted curves depict the correlation corresponding to $\boldsymbol{W}^{\mathrm{mm}}$ calculated via (Gen-SVM) and solid curves represent the correlation to $\boldsymbol{W}^{\texttt{SVMeq}}$, which incorporates the $\boldsymbol{W}^{\mathrm{fin}}$ correction. **(c)** Similar to (b), but evaluating correlations over different numbers of selected tokens.

Figure 3.10a shows that the number of selected tokens grows alongside λ, a prediction consistent with Lemma 3.6. When $\lambda = 0$, the head $h(\boldsymbol{x}) = \boldsymbol{v}^\top \boldsymbol{x}$ is linear, resulting in the selection of only one token per input. Conversely, as λ exceeds a certain threshold (e.g., $\lambda > 2.0$ based on our criteria), the optimization consistently selects all tokens. Figure 3.10b and 3.10c delve into the predictivity of attention SVM solutions for varying λ and different numbers of selected tokens. The dotted curves in both figures represent $1 - \texttt{corr\_coef}(\boldsymbol{W}, \boldsymbol{W}^{\mathrm{mm}})$, while solid curves indicate $1 - \texttt{corr\_coef}(\boldsymbol{W}, \boldsymbol{W}^{\texttt{SVMeq}})$, where $\boldsymbol{W}$ denotes the GD solution. Overall, the SVM-equivalence demonstrates a strong correlation with the GD solution (consistently above 0.95). However, selecting more tokens (aligned with larger λ values) leads to reduced predictivity.

To sum up, we have showcased the predictive capacity of the generalized SVM equivalence regarding the inductive bias of 1-layer transformers with nonlinear heads. Nevertheless, it's important to acknowledge that this section represents an initial approach to a complex problem, with certain caveats requiring further investigation (e.g., the use of filtering in Figures 3.9 and 3.10, and the presence of imperfect correlations). We aspire to conduct a more comprehensive investigation, both theoretically and empirically, in forthcoming work.

3.7 Extending the Theory to Sequential and Causal Predictions

While our formulations (W-ERM & KQ-ERM) regress a single label y_i per $(\boldsymbol{X}_i, \boldsymbol{z}_i)$, we extend in Appendix B.5 our findings to the sequence-to-sequence classification setting, where we output and classify all T tokens per input $\boldsymbol{X}_i, \boldsymbol{Z}_i \in \mathbb{R}^{T\times d}$. In this scenario, we prove that all of our RP guarantees remain intact after introducing a slight generalization of (W-SVM). Concretely, consider the following ERM problems for sequential and causal settings:

$$\begin{aligned}\mathcal{L}^{\texttt{seq}}(\boldsymbol{W}) &= \frac{1}{n}\sum_{i=1}^{n}\sum_{k=1}^{T}\ell(y_{ik}\cdot h(\boldsymbol{X}_i^\top \mathbb{S}(\boldsymbol{X}_i\boldsymbol{W}\boldsymbol{z}_{ik}))), \quad \text{and}\\ \mathcal{L}^{\texttt{csl}}(\boldsymbol{W}) &= \frac{1}{n}\sum_{i=1}^{n}\sum_{k=1}^{T}\ell(y_{ik}\cdot h(\boldsymbol{X}_i^\top \mathbb{S}_k(\boldsymbol{X}_i\boldsymbol{W}\boldsymbol{z}_{ik}))).\end{aligned} \tag{3.15}$$

Both equations train T tokens per input $(\boldsymbol{X}_i, \boldsymbol{Z}_i)$ and, as usual, we recover self-attention via $\boldsymbol{Z}_i \leftarrow \boldsymbol{X}_i$. For the causal setting, we use the masked attention $\mathbb{S}_k(\cdot)$ which calculates the softmax probabilities over the first k entries of its input and sets the remaining $T-k$ entries to zero. This way, the k'th prediction of the transformer only utilizes tokens from 1 to k and not the future tokens.

Let $\boldsymbol{\alpha} = (\alpha_{ik})_{ik=(1,1)}^{(n,k)}$ be tokens to be selected by attention (e.g. locally-optimal indices, see Def. B.1). Then, the sequential generalization of (W-SVM) corresponding to $\mathcal{L}^{\texttt{seq}}(\boldsymbol{W})$ is given by

$$\min_{\boldsymbol{W}} \|\boldsymbol{W}\|_F \quad \text{subj. to} \quad (\boldsymbol{x}_{i\alpha_{ik}} - \boldsymbol{x}_{it})^\top \boldsymbol{W}\boldsymbol{z}_{ik} \geq 1 \quad \text{for all} \quad t \neq \alpha_{ik},\ \ k\in[T],\ \ i\in[n]\ . \tag{3.16}$$

We refer the reader to Appendix B.5 which rigorously establishes all RP results for this sequential classification setting. On the other hand, for the causal inference setting SVM should reflect the fact that the model is not allowed to make use of future tokens. Note that the SVM constraints directly arise from softmax calculations. Thus, since attention is masked over the indices $t \leq k$ and $k \in [T]$, the SVM constraints should apply over the same mask. Thus, we can consider the straightforward generalization of global-optimality where opt_{ik} is the token index with highest score over indices $t \leq k$ and introduce an analogous definition for local-optimality. This leads to the following variation of (3.16), which aims to select indices $\alpha_{ik} \in [k]$ from the first k tokens

$$\min_{\boldsymbol{W}} \|\boldsymbol{W}\|_F \quad \text{subj. to} \quad (\boldsymbol{x}_{i\alpha_{ik}} - \boldsymbol{x}_{it})^\top \boldsymbol{W}\boldsymbol{z}_{ik} \geq 1 \quad \text{for all} \quad t \neq \alpha_{ik},\ \ t\leq k,\ \ k\in[T],\ \ i\in[n].$$

Causal attention is a special case of a general attention mask which can restrict softmax to arbitrary subset of the tokens. Such general masking can be handled similar to (3.7) by enforcing SVM constraints over the nonzero support of the mask. Finally, the discussion so far extends our main theoretical results and focuses on selecting single token per sequence. It can further be enriched by the generalized SVM equivalence developed in Section 3.6 to select and compose multiple tokens by generalizing (Gen-SVM).

CHAPTER 4

Convergence of Gradient Descent in Next-Token Prediction

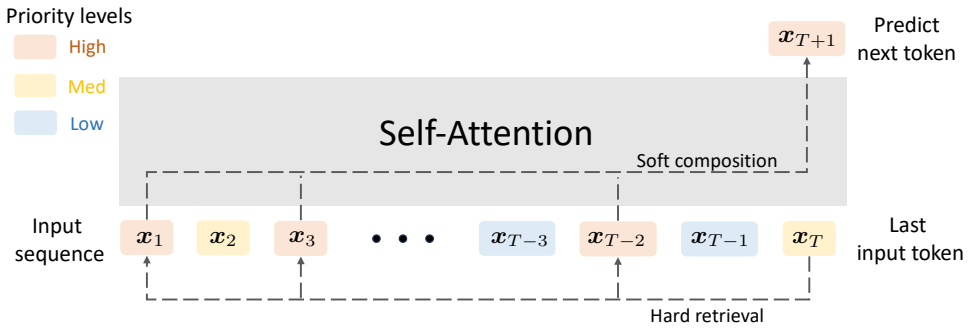


Figure 4.1: Overview of our result on next-token prediction. We study the implicit bias of gradient descent where a 1-layer self-attention model is trained until convergence. We prove that, during test-time, this model implements a hard retrieval to precisely select the high-priority tokens and then outputs a convex combination of these as the output from which the next token can be sampled. The notion of *high-priority* is formalized through the strongly-connected components of a directed graph associated to the last input token.

4.1 Introduction

This work aims to bridge this gap between the empirical success and principled understanding of Transformer-based language modeling by shedding light on the optimization landscape and key implicit biases faced by the self-attention mechanism in solving the next-token prediction task. In particular, focusing on a *single-layer* self-attention model with linear classification head, and solving the next-token prediction task, we consider the following questions:

Q1: *What relationships in training data are captured by the single-layer self-attention model?*

Q2: *How exactly do these relationships dictate the optimization geometry of natural algorithms such as gradient descent?*

We show that the answers to both of these questions are intertwined which we achieve by significantly expanding the recently proposed framework that connects learning with Transformers to the celebrated support vector machines (SVMs) [185, 186].

As illustrated in Figure 4.1, given training data as a collection of (input sequence, next token) pairs, self-attention model learns to (1) retrieve the high-priority tokens (highlighted with red color) to the last input token; and then (2) build a convex combination of these high-priority tokens. The notion of *high-priority* is dictated by a directed graph learned from training data. SGD training accomplishes this by learning hard and soft components of the attention weights $\boldsymbol{W}$ to execute (1) and (2) respectively. Concretely, the following theorem dictates the evolution of attention weights during gradient descent.

Theorem 4.1 (*informal*) *Consider training a single-layer self-attention model with gradient descent. The combined attention weights $\boldsymbol{W} := \boldsymbol{W}_k \boldsymbol{W}_q^\top$ evolve as*

$$\boldsymbol{W}_{gd} \approx C \cdot \boldsymbol{W}_{hard} + \boldsymbol{W}_{soft},$$

where $C \cdot \boldsymbol{W}_{hard}$ is the hard retrieval component selecting the high-priority tokens when $C \to \infty$; and $\boldsymbol{W}_{soft}$ is the soft composition component allocating nonzero softmax probabilities over selected tokens.

To capture the priority order among different tokens as observed in Figure 4.1, we construct directed graphs among the tokens in the vocabulary, namely *token-priority graphs* (TPGs). An illustration is provided in Figure 4.2, where a *strongly connected component* (SCC, highlighted as dashed black rectangles) in a TPG corresponds to the tokens that are reachable from each other, indicating the absence of a strict priority among those tokens. The hard retrieval component $\boldsymbol{W}_{\text{hard}}$ captures the topological order of different SCCs (orange arrows) whereas the soft composition component $\boldsymbol{W}_{\text{soft}}$ captures the relationships of different tokens within each SCC (black arrows). These TPGs will geometrically capture the learning dynamics of self-attention. Specifically, we propose the SVM problem (Graph-SVM), solution of which describes the direction gradient descent converges to. This way, SGD asymptotically enforces the topological order between SCCs i.e. the $C \cdot \boldsymbol{W}_{\text{hard}}$ term in Theorem 4.1 as $C \to \infty$. In practice, this implies that self-attention model favors suppressing lower priority tokens in favour of sampling higher priority tokens.

A conjecture on the decomposition in Theorem 4.1 was first proposed in [185][1]. This

[1]Their conjecture aims to characterize the impact of the MLP layer that follow self-attention in a binary classification setting. However, the high-level claim is same.

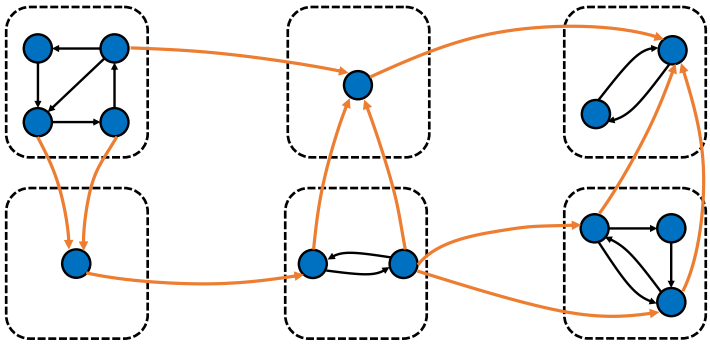


Figure 4.2: A token-priority graph (TPG) is a directed graph derived from training data (see Sec 4.2.1 for definition). The edges in TPG capture the input-output relationships between different tokens. A TPG can be partitioned into several SCCs depicted as dashed black squares. In light of Theorem 4.1, black intra-SCC edges within each SCC induce the soft-composition component of the attention weights whereas the orange edges induce the hard-retrieval component enforcing the priority orders among various SCCs.

decomposition is also related to the implicit bias of logistic regression on non-separable data [89]. Our theory fully formalizes this decomposition under the next-token prediction setting and reveals fundamental connections to graphical structure in data (e.g. through SCCs, TPGs).

Overall, we carefully study the gradient descent and regularization path algorithms for attention-based next-token prediction and make the following contributions:

- ◇ We study the optimization landscape of self-attention with log-loss and show that the problem is convex under suitable assumptions. We then establish a global convergence result to fully formalize Theorem 4.1 in terms of a directional component (Graph-SVM) and a finite component (see Sec 4.3). Notably, results apply to arbitrary datasets as we don't require distributional assumptions.

- ◇ Our theory reveals insightful connections between continuous and discrete optimization, namely: *Self-attention implicitly discovers the strongly-connected components of the TPGs during training.*

- ◇ Under a general setting, we establish the implicit bias of the solution obtained by vanishing regularization [87, 169, 179]. This yields a result similar to the gradient descent theory (see Sec 4.4). We also show that, in general, gradient descent can exhibit local directional convergence rather than global. We characterize these local directions through the SVM solutions of *pseudo TPGs* (see Sec 4.5).

4.2 Preliminaries and Problem Setup

Notation. Let $[n]$ denote the set $\{1, \cdots, n\}$. For a space $\mathcal{S}$, let $\mathcal{S}^\perp$ denote the orthogonal complement of $\mathcal{S}$ and $\mathbf{\Pi}_{\mathcal{S}}$ denote the projection operator on $\mathcal{S}$ with respect to Euclidean distance.

Next-token prediction problem. Let K be the vocabulary size with $\boldsymbol{E} = [\boldsymbol{e}_1 \ \ldots \ \boldsymbol{e}_K]^\top \in \mathbb{R}^{K\times d}$ denoting the embedding matrix consisting of d-dimensional token embeddings for the K tokens in the vocabulary. The next-token prediction is a multi-class classification problem and the goal is to predict the ID $y \in [K]$ of the next token given an input sequence $\boldsymbol{X} = [\boldsymbol{x}_1 \ \ldots \ \boldsymbol{x}_T]^\top \in \mathbb{R}^{T\times d}$, where $\boldsymbol{x}_t \in \boldsymbol{E}$ for all $t \in [T]$.

Suppose that we have a training dataset $\texttt{DSET} = \big\{(\boldsymbol{X}_i, y_i) \in \mathbb{R}^{T_i\times d} \times [K]\big\}_{i=1}^n$ consisting of n sequences where we allow the sequences to have different lengths T_i, $i \in [n]$. Throughout this paper, we use $x_{it} \in [K]$ to denote the scalar token ID corresponding to the t-th token $\boldsymbol{x}_{it} \in \mathbb{R}^d$ of the input sequence $\boldsymbol{X}_i$, i.e., $\boldsymbol{x}_{it} = \boldsymbol{e}_{x_{it}}$.

Self-attention model. We consider a *single-layer* self-attention model when making a prediction on a given input sequence $\boldsymbol{X} \in \mathbb{R}^{T\times d}$. Following the previous work [185], we denote the combined key-query weights by a trainable $\boldsymbol{W} \in \mathbb{R}^{d\times d}$ matrix, and assume identity value matrix. Let $\bar{\boldsymbol{x}} := \boldsymbol{x}_T$ be the last token of the input sequence $\boldsymbol{X}$. Then, the single-layer self-attention outputs the following embedding to predict the next-token ID y:

$$f_{\boldsymbol{W}}(\boldsymbol{X}) = \boldsymbol{X}^\top \mathbb{S}(\boldsymbol{X}\boldsymbol{W}\bar{\boldsymbol{x}}), \tag{4.1}$$

where $\mathbb{S}(\cdot)$ denotes the softmax operation which facilitates weighing tokens of $\boldsymbol{X}$ based on the data-dependent probabilities $\mathbb{S}(\boldsymbol{X}\boldsymbol{W}\bar{\boldsymbol{x}})$. Note that the output embedding $f_{\boldsymbol{W}}(\boldsymbol{X}) \in \mathbb{R}^d$ in (4.1) is a weighted linear combination of the input token embeddings in $\boldsymbol{X}$.

Empirical risk minimization (ERM) problem. Let $\ell : \mathbb{R} \to \mathbb{R}$ be the loss function. Given training dataset DSET, we consider the ERM problem with the following objective:

$$\mathcal{L}(\boldsymbol{W}) = \frac{1}{n}\sum_{i=1}^n \ell(\boldsymbol{c}_{y_i}^\top \boldsymbol{X}_i^\top \mathbb{S}(\boldsymbol{X}_i\boldsymbol{W}\bar{\boldsymbol{x}}_i)). \tag{ERM}$$

Throughout this paper, we fix the linear classification head $\boldsymbol{c}_k$[2] and assume $\|\boldsymbol{c}_k\|$ is bounded for all $k \in [K]$. Note that even though the classification head is linear, the problem of learning

[2]Specifically, we assume well-pretrained head $\boldsymbol{c}_1, \cdots, \boldsymbol{c}_K$ such that $\ell(\boldsymbol{c}_y^\top \boldsymbol{e}_k)$ returns the minimal risk when $k = y$.

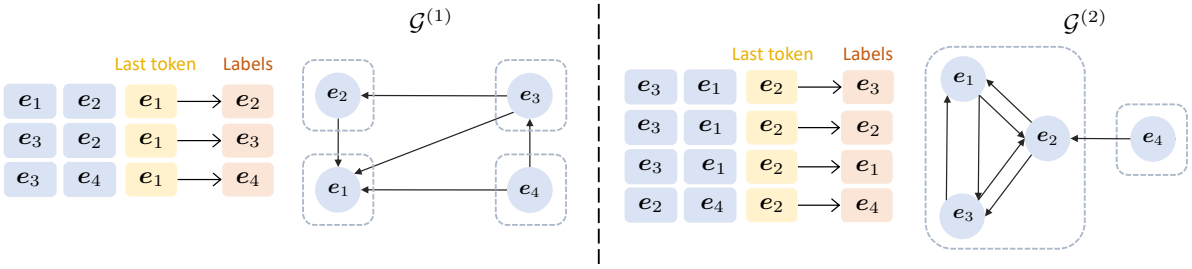


Figure 4.3: Illustration of token-priority graph (TPG). Given the input sequences and labels (next tokens), we construct the TPGs $\{\mathcal{G}^{(k)}\}_{k=1}^{K}$ according to the last token. Two TPGs $\mathcal{G}^{(1)}$ (left) and $\mathcal{G}^{(2)}$ (right) are constructed using the samples with $\boldsymbol{e}_1$ and $\boldsymbol{e}_2$ as the last tokens, respectively. In each graph, directed edges (label token→input token) are added between tokens/nodes. Based on these directed edges, each graph can be partitioned into its strongly-connected components (SCCs, highlighted as dashed grey rectangles). Each SCC is a set of tokens where each token is reachable from every other token within that SCC. Further details are deferred to Section 4.2.1.

attention parameters $\boldsymbol{W}$ via ERM is not necessarily convex due to the softmax operator. In this work, we focus on this exact problem and consider the following two algorithms to optimze for $\boldsymbol{W}$:

1. Gradient descent: Given starting point $\boldsymbol{W}(0) \in \mathbb{R}^{d\times d}$ and step size $\eta > 0$, for $\tau \geq 0$,

$$\boldsymbol{W}(\tau+1) = \boldsymbol{W}(\tau) - \eta\nabla\mathcal{L}(\boldsymbol{W}(\tau)). \tag{Algo-GD}$$

2. Regularization path: Given $R > 0$, $\boldsymbol{W} \in \mathbb{R}^{d\times d}$,

$$\bar{\boldsymbol{W}}(R) = \arg\min_{\|\boldsymbol{W}\|_F \leq R} \mathcal{L}(\boldsymbol{W}). \tag{Algo-RP}$$

The next-token prediction task aims to capture various patterns present in the underlying dataset. Towards this, we introduce *token-priority graph* (TPG) in the following section that summarizes the sequential priority orders presented in the training data. As we will see later, TPGs play a crucial role in characterizing the optimization geometry for both (Algo-GD) and (Algo-RP) algorithms.

4.2.1 Token-priority Graph of the Dataset

A token-priority graph (TPG) is a directed graph with at most K nodes corresponding to the elements in the vocabulary. We associate the dataset $\texttt{DSET} = \{(\boldsymbol{X}_i, y_i)\}_{i=1}^{n}$ with multiple TPGs $\{\mathcal{G}^{(k)}\}_{k=1}^{K}$, with each TPG focusing on a subset of the dataset comprising of those

input sequences that agree on the last token $\bar{\boldsymbol{x}}$. Concretely, we construct $\mathcal{G}^{(k)}$'s as follows:

1. Split DSET into K subsets $\{\texttt{DSET}^{(k)}\}_{k=1}^{K}$ with $\texttt{DSET}^{(k)}$ containing all input sequences that end with the same last token $\bar{\boldsymbol{x}} = \boldsymbol{e}_k$.

2. For each $(\boldsymbol{X}, y) \in \texttt{DSET}^{(k)}$ and for all (x, y) pairs in $(\boldsymbol{X}, y)$ where x is the corresponding token ID of $\boldsymbol{x} \in \boldsymbol{X}$, add a directed edge $(y \to x)$ to $\mathcal{G}^{(k)}$.

An illustration is provided in Fig. 4.3, where we construct two TPGs ($\mathcal{G}^{(1)}$ and $\mathcal{G}^{(2)}$) based on the last tokens (depicted in yellow), and the directed edges are presented as arrows starting from labels (orange) to input tokens (blue/yellow) within each sequence. Note that nodes of each $\mathcal{G}^{(k)}$ constitute a subset of the indices $[K]$. The edges in $\mathcal{G}^{(k)}$ capture the priorities across the tokens in an extended data sequence, conditioned on the last token of the input being $\bar{\boldsymbol{x}} = \boldsymbol{e}_k$. We will see that if there is a cycle, i.e., $y \to x$ and $x \to y$ are both directionally reachable in the graph, then the self-attention learnt via next-token prediction task can assign comparable priorities to the tokens x and y. In contrast, if y always *dominates* x, i.e., $y \to x$ is reachable but $x \not\to y$, then, when x and y are both present in an input sequence, self-attention will be learnt to suppress x and select y through an SVM mechanism along the line of [185].

Strongly-connected components in TPGs. To formalize the aforementioned SVM mechanism, we need the notion of *strongly-connected components* (SCCs). A directed graph is strongly connected if every node in the graph is reachable from every other node. SCCs of a directed graph form a partition into subgraphs that are themselves strongly connected. Given the TPGs $\{\mathcal{G}^{(k)}\}_{k=1}^{K}$ associated with the dataset DSET, we can split the directed graph $\mathcal{G}^{(k)}$ into its SCCs, denoted by $\{\mathcal{C}_i^{(k)}\}_{i=1}^{N_k}$. Note that the number of SCCs in $\mathcal{G}^{(k)}$, as denoted by N_k, is at most the number of nodes in $\mathcal{G}^{(k)}$, which is upper bounded by the vocabulary size K. Furthermore, by definition, different SCCs within a graph consist of distinct nodes, i.e., $\mathcal{C}_i^{(k)} \bigcap \mathcal{C}_j^{(k)} = \emptyset$, for $i \neq j$. Now, returning to Fig. 4.3, each of the dashed grey rectangle represents an SCC. $\mathcal{G}^{(1)}$(left) contains four SCCs and therefore, all tokens within the graph have strict priority orders. In contrast, $\mathcal{G}^{(2)}$(right) consists of two SCCs, with one containing three nodes. Following the arrows, we can see that all the tokens/nodes within this specific SCC are directional reachable.

Before formally connecting TPGs and their SCCs to the SVM mechanism that enables next-token prediction, we introduce some necessary graph-related notation. Given a directed graph $\mathcal{G}$, for $i, j \in [K]$ such that $i \neq j$:

- $i \in \mathcal{G}$ denotes that the node i belongs to $\mathcal{G}$.

- $(i \Rightarrow j) \in \mathcal{G}$ denotes that the directed edge $(i \to j)$ is present in $\mathcal{G}$ but $j \to i$ is not reachable.
- $(i \asymp j) \in \mathcal{G}$ means that both two nodes (i, j) are in the same SCC of $\mathcal{G}$.

From the construction, for any two distinct nodes i, j in the same TPG, they either satisfy $(i \Rightarrow j)/(j \Rightarrow i)$ or $(i \asymp j)$.

4.2.2 SVM Bias of Self-attention Learning

The main contribution of this paper is to establish the SVM equivalence that captures the optimization geometry of the next-token prediction problem. We will show that the self-attention model learnt via either (Algo-GD) or (Algo-RP) converges to the solution of an SVM defined by the TPGs of the underlying dataset DSET. In particular, given $(\mathcal{G}_k^{(k)})_{k=1}^K$, we introduce the following SVM formulation:

$$\boldsymbol{W}^{\mathrm{mm}} = \arg\min_{\boldsymbol{W}} \|\boldsymbol{W}\|_F \qquad \text{(Graph-SVM)}$$
$$\text{s.t. } (\boldsymbol{e}_i - \boldsymbol{e}_j)^\top \boldsymbol{W} \boldsymbol{e}_k \begin{cases} = 0 & \forall (i \asymp j) \in \mathcal{G}^{(k)} \\ \geq 1 & \forall (i \Rightarrow j) \in \mathcal{G}^{(k)} \end{cases} \quad \text{for all} \quad k \in [K].$$

Fix last token $\boldsymbol{e}_k$, and consider any token IDs $i, j \in [K]$, $i \neq j$. When $(i \Rightarrow j)$, token ID i has a higher *priority* than j and hence, the SVM problem (Graph-SVM) aims to find a $\boldsymbol{W}$ such that $\boldsymbol{W}\boldsymbol{e}_k$ achieves strictly higher correlation to token embedding $\boldsymbol{e}_i$ than $\boldsymbol{e}_j$, that is, $\boldsymbol{e}_i^\top \boldsymbol{W} \boldsymbol{e}_k \geq \boldsymbol{e}_j^\top \boldsymbol{W} \boldsymbol{e}_k + 1$, and then softmax operation will assign higher probability to the token i. While if $(i \asymp j)$, there is not strict priority order between i and j, and hence we set the correlation difference equal to zero to prevent the SVM solution $\boldsymbol{W}$ from distinguishing them. The existence of the solution $\boldsymbol{W}^{\mathrm{mm}}$ ensures the separability of tokens i's from the j's for all pairs $(i \Rightarrow j) \in \mathcal{G}^{(k)}$. Additionally, if for all $k \in [K]$, the number of SCCs[3] $N_k \leq 1$, then $\boldsymbol{W}^{\mathrm{mm}} = 0$.

Lemma 4.1 *Suppose that the embedding matrix $\boldsymbol{E}$ is full row rank. Then,* (Graph-SVM) *is feasible.*

Next focusing on the nodes i, j, with $(i \asymp j)$, we introduce the following subspace definition.

[3]Note that $N_k = 0$ implies that within DSET, there is not training sample whose input sequence has $\boldsymbol{e}_k$ as its last token; or equivalently, $\mathtt{DSET}^{(k)} = \emptyset$.

Definition 4.1 (Cyclic subspace) *Define cyclic subspace $\mathcal{S}_{fin}$ as the span of all matrices $(\boldsymbol{e}_i - \boldsymbol{e}_j)\boldsymbol{e}_k^\top$ for all $(i \asymp j) \in \mathcal{G}^{(k)}$ and $k \in [K]$.*

Note that since $\boldsymbol{W}^{\mathrm{mm}}$ satisfies all the "$= 0$" constraints in (Graph-SVM), if (Graph-SVM) is feasible and $\boldsymbol{W}^{\mathrm{mm}} \neq 0$, $\boldsymbol{W}^{\mathrm{mm}} \perp \mathcal{S}_{\mathrm{fin}}$.

4.2.3 Technical Assumptions

In what follows, we work with a few assumptions that will make the optimization landscape of the underlying learning problem more benign, and we introduce these assumptions along with their justifications.

Assumption 4.1 *For $\forall y, k \in [K]$, $k \neq y$, $\boldsymbol{c}_y^\top \boldsymbol{e}_y = 1$ and $\boldsymbol{c}_y^\top \boldsymbol{e}_k = 0$.*

This assumption essentially enforces that the rows of the prediction head $\boldsymbol{C}$ are aligned with the corresponding vocabulary embeddings in $\boldsymbol{E}$. This is a variation of the *weight tying* strategy which is commonly employed in language models [160, 192]. It should be noted that Assumption 4.1 implies $K \leq d$, which further establishes the feasibility of (Graph-SVM) as demonstrated by Lemma 4.1. Given objective (ERM) and a decreasing loss function ℓ, our ideal goal is for the attention (4.1) to output $\boldsymbol{e}_y$ which minimizes the training risk. Recall that single-layer self-attention outputs a convex combination of the input tokens (cf. (4.1)). If tokens are linearly independent, the only way model can output the embedding $\boldsymbol{e}_y$ corresponding to the target label y would be if $\boldsymbol{e}_y$ was among the input sequence. This motivates the following realizability assumption.

Assumption 4.2 *For any $(\boldsymbol{X}, y) \in$ `DSET`, the token $\boldsymbol{e}_y$ is contained in input sequence $\boldsymbol{X}$.*

In the scenario where $(\boldsymbol{X}, y)$ is not realizable, self-attention would select $\boldsymbol{e}_{\hat{y}} \neq \boldsymbol{e}_y$ and the SVM formula would be established via separating $\hat{y}$ from the other tokens in the sequence instead of the true label y. Additionally, when Assumption 4.1 holds, the model can only make a random prediction over the output labels (since any output of the self-attention model would result in the same training risk); consequently, such non-realizable examples will not play roles in optimizing $\boldsymbol{W}$, i.e. $\nabla_{\boldsymbol{W}} \ell(\boldsymbol{c}_{y_i}^\top \boldsymbol{X}_i^\top \mathbb{S}(\boldsymbol{X}_i \boldsymbol{W} \bar{\boldsymbol{x}}_i)) = 0$.

4.3 Global Convergence of Gradient Descent

In this section, we assume the log-loss function, i.e., $\ell(u) = -\log(u)$, and establish the gradient descent convergence of attention weight $\boldsymbol{W}$ via the convexity of $\mathcal{L}(\boldsymbol{W})$. Note that

although loss function ℓ is convex and the classification head is linear, due to the non-convexity of softmax, the convexity of $\mathcal{L}(\boldsymbol{W})$ is not immediately clear. Towards this, we introduce the following lemma:

Lemma 4.2 *Suppose Assumptions 4.1 and 4.2 hold and consider the log-loss $\ell(u) = -\log(u)$, then $\mathcal{L}(\boldsymbol{W})$ is convex. Furthermore, $\mathcal{L}(\boldsymbol{W})$ is strictly convex on $\mathcal{S}_{fin}$.*

Let $(\boldsymbol{X}, y) \in \texttt{DSET}$ be any sample and set $\gamma_t = \boldsymbol{c}_y^\top \boldsymbol{x}_t$. Assumption 4.1 guarantees that $\gamma_t = 1$ when $x_t = y$, otherwise $\gamma_t = 0$. Consider the attention output $\boldsymbol{c}_y^\top \boldsymbol{X}^\top \mathbb{S}(\boldsymbol{X}\boldsymbol{W}\bar{\boldsymbol{x}})$ (cf. (ERM)) and let softmax probabilities be $s_t = \mathbb{S}(\boldsymbol{X}\boldsymbol{W}\bar{\boldsymbol{x}})_t$, where $\sum_t s_t = 1$. Then, the loss of this single sample $\boldsymbol{X}$ is $\ell(\bar{\gamma})$ where $\bar{\gamma} = \sum_t \gamma_t s_t = \sum_{x_t = y} s_t$. Given log-loss, note that when $\bar{\gamma} \to 0^+$, $-\log(\bar{\gamma})$ results in the infinite loss, which suggests that, once attention weight $\boldsymbol{W}$ diverges to saturate the softmax probability, the finite training risk is achievable only when $s_t \not\to 0$ for all t satisfying $x_t = y$. Given different label y for different input and recalling the SCC definition in Section 4.2.1, attention selects all $\boldsymbol{x}_t$'s within the same SCC as y. The following result characterizes the global directional convergence of the GD iterates to the solution of (Graph-SVM).

Theorem 4.2 *Consider a dataset $\texttt{DSET}$ and suppose Assumptions 4.1 and 4.2 hold. Set loss function as $\ell(u) = -\log(u)$. Let $\boldsymbol{W}^{mm} \in \mathcal{S}_{fin}^{\perp}$ be the solution of* (Graph-SVM)*. Starting from any $\boldsymbol{W}(0)$ with constant step size η, the algorithm Algo-GD satisfies $\lim_{\tau\to\infty} \|\boldsymbol{W}(\tau)\|_F = \infty$ and*

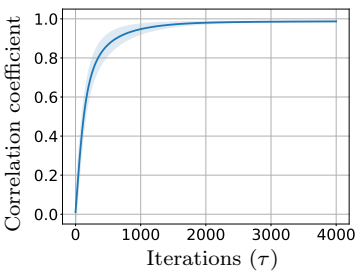
$$\lim_{\tau\to\infty} \boldsymbol{\Pi}_{\mathcal{S}_{fin}}(\boldsymbol{W}(\tau)) = \boldsymbol{W}^{fin}. \tag{4.2}$$

Here $\boldsymbol{W}^{fin}$ is the unique finite minima of the loss $\tilde{\mathcal{L}}(\boldsymbol{W}) := \lim_{R\to\infty} \mathcal{L}(\boldsymbol{W} + R \cdot \boldsymbol{W}^{mm})$ over $\mathcal{S}_{fin}$. Additionally, if $\boldsymbol{W}^{mm} \neq 0$,

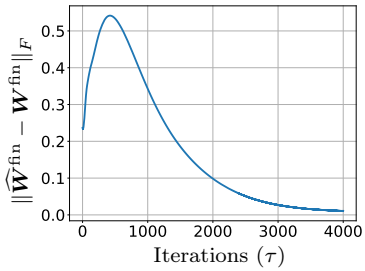
$$\lim_{\tau\to\infty} \frac{\boldsymbol{W}(\tau)}{\|\boldsymbol{W}(\tau)\|_F} = \frac{\boldsymbol{W}^{mm}}{\|\boldsymbol{W}^{mm}\|_F}.$$

Otherwise, $\boldsymbol{\Pi}_{\mathcal{S}_{fin}^{\perp}}(\boldsymbol{W}(\tau))$ remains unchanged throughout the optimization.

This theorem demonstrates the directional convergence of attention weight $\boldsymbol{W}$, and the limits imply the decomposition $\boldsymbol{W}(\tau) \approx C(\tau) \cdot \boldsymbol{W}^{\text{mm}} + \boldsymbol{W}^{\text{fin}}$ for an appropriate $C(\tau) > 0$ with $\lim_{\tau\to\infty} C(\tau) = \infty$. Note that, for directional convergence to happen, we need $\boldsymbol{W}^{\text{mm}} \neq 0$ which happens if and only if (Graph-SVM) has " ≥ 1" constraints. That is, the token graph contains a strict priority order. This is consistent with our Theorem 4.1 where $\boldsymbol{W}^{\text{mm}}$ corresponds to the hard retrieval component ($\boldsymbol{W}_{\text{hard}}$) that selects the high-priority tokens



(a) Evolution of $\frac{\boldsymbol{W}(\tau)}{\|\boldsymbol{W}(\tau)\|_F} \to \frac{\boldsymbol{W}^{\mathrm{mm}}}{\|\boldsymbol{W}^{\mathrm{mm}}\|_F}$



(b) Evolution of $\boldsymbol{\Pi}_{\mathcal{S}_{\mathrm{fin}}}(\boldsymbol{W}(\tau)) \to \boldsymbol{W}^{\mathrm{fin}}$

Figure 4.4: GD convergence of attention weight $\boldsymbol{W}$ when training with general dataset. **(a)** shows the directional convergence of $\boldsymbol{W}(\tau)$; while **(b)** presents the convergence of $\boldsymbol{\Pi}_{\mathcal{S}_{\mathrm{fin}}}(\boldsymbol{W}(\tau))$.

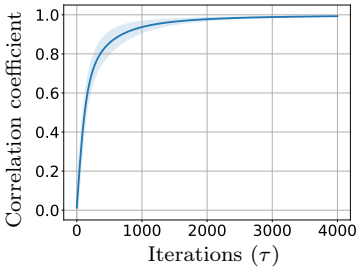


Figure 4.5: GD convergence of $\boldsymbol{W}$ when training with acyclic dataset (Def. 4.2). Correlation coefficient between $\boldsymbol{W}(\tau)$ and $\boldsymbol{W}^{\mathrm{mm}}$ are presented.

and $\boldsymbol{W}^{\mathrm{fin}}$ is the soft composition component ($\boldsymbol{W}_{\mathrm{soft}}$) that determines the softmax probability assignments among these selected tokens. Importantly, this is a global convergence result thanks to the convexity of the optimization problem, which is enabled by the log likelihood optimization combined with Assumption 4.1. In Section 4.5, we will demonstrate that global convergence of gradient descent does not hold in general.

To illustrate Theorem 4.2, we conduct experiments with results presented in Figure 4.4. We create embedding tables with $K = 6$, $d = 8$ and randomly generate dataset with $n = 6$, $T = 4$. Here we choose step size $\eta = 0.01$ and perform the normalized gradient descent method to accelerate the increase in the norm of attention weight, so that softmax can easily saturate. Specifically, we update attention weight $\boldsymbol{W}$ via $\boldsymbol{W}(\tau+1) = \boldsymbol{W}(\tau) - \eta \frac{\nabla \mathcal{L}(\boldsymbol{W}(\tau))}{\|\nabla \mathcal{L}(\boldsymbol{W}(\tau))\|_F}$. At each iteration τ, correlation coefficient is computed by $\langle \boldsymbol{W}(\tau), \boldsymbol{W}^{\mathrm{mm}} \rangle / (\|\boldsymbol{W}(\tau)\|_F \|\boldsymbol{W}^{\mathrm{mm}}\|_F)$, and results averaged over 100 random instances are displayed in Fig. 4.4a, which end in correlation ≈ 0.987 after training with 4000 iterations. In addition to the directional convergence of $\boldsymbol{W}(\tau)$, we also verify the convergence of finite component by tracking the matrix distance $\|\widehat{\boldsymbol{W}}^{\mathrm{fin}} - \boldsymbol{W}^{\mathrm{fin}}\|_F$ where $\widehat{\boldsymbol{W}}^{\mathrm{fin}} = \boldsymbol{\Pi}_{\mathcal{S}_{\mathrm{fin}}}(\boldsymbol{W}(\tau))$, and results are displayed in Fig. 4.4b, which obtains < 0.01 final distance. Both results validate our Theorem 4.2.

4.4 Implicit Bias of Self-attention

In Section 4.3, we have discussed that gradient descent with log loss guarantees the global convergence. To proceed, in this section, we discuss the implicit bias of attention via analysis of regularization path (RP) as employed in Algo-RP and identify the implicit bias of self-

attention on more general next-token prediction problems.

Theorem 4.3 *Consider any dataset* `DSET` *and suppose Assumptions 4.1 and 4.2 hold. Additionally, assume loss* $\ell : \mathbb{R} \to \mathbb{R}$ *is strictly decreasing and* $|\ell'|$ *is bounded. Let* $\boldsymbol{W}^{mm} \in \mathcal{S}_{fin}^{\perp}$ *be the solution of* (Graph-SVM) *and suppose* $\boldsymbol{W}^{mm} \neq 0$*. Then the solution of regularization path Algo-RP obeys*

$$\lim_{R\to\infty} \frac{\bar{\boldsymbol{W}}_R}{R} = \frac{\boldsymbol{W}^{mm}}{\|\boldsymbol{W}^{mm}\|_F} \quad \textit{and} \quad \lim_{R\to\infty} \boldsymbol{\Pi}_{\mathcal{S}_{fin}}(\bar{\boldsymbol{W}}_R) \in \mathcal{W}^{fin}.$$

Here $\mathcal{W}^{fin} = \arg\min_{\boldsymbol{W}\in\mathcal{S}_{fin}} \lim_{R\to\infty} \mathcal{L}(\boldsymbol{W} + R \cdot \boldsymbol{W}^{mm})$ *and we assume that* $\mathcal{W}^{fin}$ *is a bounded set.*

Here, we allow more general loss function ℓ which is different from the log-loss employed in Section 4.3, and the ERM problem (cf. (ERM)) is not guaranteed to be convex.

4.4.1 Acyclic Dataset

Below, in contrast, we introduce the concept of acyclic dataset which implies that the next-token prediction task always encounters a strict priority order among tokens within each TPG. This corresponds to the setting where all SCCs $((\mathcal{C}_i^{(k)})_{i=1}^{N_k})_{k=1}^{K}$ are all singletons; or equivalently, $\overline{\texttt{DSET}} = \emptyset$.

Definition 4.2 (Acyclic dataset) *We call* `DSET` *acyclic if all of its TPGs are directed acyclic graphs.*

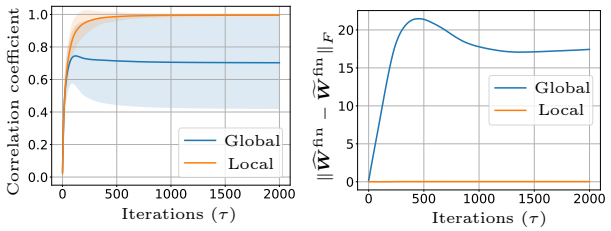
For an acyclic dataset, there are not $i,\, j \in [K]$ satisfying $(i \asymp j) \in \mathcal{G}^{(k)}$, for all $k \in [K]$. Thus, SVM formulation (Graph-SVM) reduces to the following simpler form:

$$\begin{aligned} &\boldsymbol{W}^{\mathrm{mm}} = \arg\min_{\boldsymbol{W}} \|\boldsymbol{W}\|_F \qquad \text{(Acyc-SVM)} \\ &\text{s.t.} \quad (\boldsymbol{e}_i - \boldsymbol{e}_j)^\top \boldsymbol{W} \boldsymbol{e}_k \geq 1 \quad \forall (i \Rightarrow j) \in \mathcal{G}^{(k)},\ k \in [K]. \end{aligned}$$

We next make the following assumption on the linear head. Notably, Assumption 4.1 is a special case of Assumption 4.3.

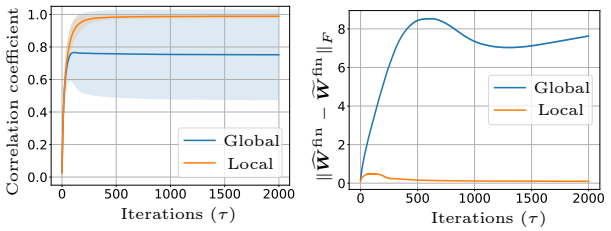
Assumption 4.3 *For* $\forall y \in [K]$, $\arg\max_{k\in[K]} \boldsymbol{c}_y^\top \boldsymbol{e}_k = y$.

Lemma 4.3 *Consider acyclic dataset* `DSET` *per Def. 4.2 and suppose Assumptions 4.2 and 4.3 hold. Additionally, assume loss function* ℓ *is strictly decreasing and* $|\ell'|$ *is bounded. Then for any finite* $\boldsymbol{W} \in \mathbb{R}^{d\times d}$*, training risk obeys* $\mathcal{L}(\boldsymbol{W}) > \mathcal{L}_\star := \frac{1}{n}\sum_{i=1}^n \ell(\boldsymbol{c}_{y_i}^\top \boldsymbol{e}_{y_i})$*. Additionally,*



(a) $\frac{\boldsymbol{W}(\tau)}{\|\boldsymbol{W}(\tau)\|_F} \to \frac{\widetilde{\boldsymbol{W}}^{\text{svm}}}{\|\widetilde{\boldsymbol{W}}^{\text{svm}}\|_F}$ (b) $\boldsymbol{\Pi}_{\widetilde{\mathcal{S}}_{\text{fin}}}(\boldsymbol{W}(\tau)) \to \widetilde{\boldsymbol{W}}^{\text{fin}}$

Figure 4.6: Squared loss & general classifier



(a) $\frac{\boldsymbol{W}(\tau)}{\|\boldsymbol{W}(\tau)\|_F} \to \frac{\widetilde{\boldsymbol{W}}^{\text{svm}}}{\|\widetilde{\boldsymbol{W}}^{\text{svm}}\|_F}$ (b) $\boldsymbol{\Pi}_{\widetilde{\mathcal{S}}_{\text{fin}}}(\boldsymbol{W}(\tau)) \to \widetilde{\boldsymbol{W}}^{\text{fin}}$

Figure 4.7: Cross-entropy & general classifier

if (Acyc-SVM) *is feasible, then for any* $\boldsymbol{W}$ *that satisfies the constraints in* (Acyc-SVM), $\lim_{R\to\infty} \mathcal{L}(R \cdot \boldsymbol{W}) = \mathcal{L}_\star$.

Assumption 4.3 and Lemma 4.3 ensure that the best way for attention to make a correct prediction on class k is to output the vector $\boldsymbol{e}_k$, i.e., $f_{\boldsymbol{W}}(\boldsymbol{X}) = \boldsymbol{e}_k$. The next theorem states the directional bias of self-attention on the acyclic dataset towards the solution of (Acyc-SVM).

Theorem 4.4 *Suppose* `DSET` *is acyclic per Definition 4.2 and Assumptions 4.2 and 4.3 hold. Additionally, assume loss* $\ell : \mathbb{R} \to \mathbb{R}$ *is strictly decreasing and* $|\ell'|$ *is bounded. Suppose* (Acyc-SVM) *is feasible with* $\boldsymbol{W}^{mm}$ *denoting its solution. Then, Algorithm Algo-RP satisfies*

$$\lim_{R\to\infty} \frac{\bar{\boldsymbol{W}}(R)}{R} = \frac{\boldsymbol{W}^{mm}}{\|\boldsymbol{W}^{mm}\|_F}.$$

Recall that a dataset being acyclic implies that there is strict priority order among all tokens in each TPG. Theorem 4.4 establishes the implicit bias of self-attention model for next-token prediction problem in the presence of such strict priority order. It demonstrates that once the SVM problem (Acyc-SVM) is feasible, the regularized path of optimizing (ERM) converges directionally toward it solution $\boldsymbol{W}^{\text{mm}}$.

Following the same implementation setting as in Section 4.3, in Fig. 4.5, we again conduct 100 trials but with randomly generated acyclic dataset `DSET` under the setting of $K = d = 8, n = 4$ and $T = 6$. The results averaged over 100 random instances are presented in Fig. 4.5 with correlation coefficient exceeds 0.99 after training with 4000 iterations. Note that in these experiments, log-loss is employed as loss function and Assumption 4.1 is satisfied which guarantee the convexity of $\mathcal{L}(\boldsymbol{W})$ following Lemma 4.2 and hence, the connection between Algo-GD and Algo-RP is built by [87].

4.5 Further Investigation on Local Convergence

So far, we have proved the global GD convergence of attention weight when one employs the log-loss (Section 4.3) and studied the implicit bias of self-attention over next-token prediction problem using RP analysis (Section 4.4). In this section, we investigate further on the convergence performance of GD and ask:

When does GD exhibit local convergence rather than global? Can we characterize its implicit bias?

Convergence performance of learning 1-layer attention has been analyzed in the previous work [185, 186], and they have observed the local convergence phenomenon, and also provided the theoretical explanation and empirical evidence. Inspired by their work, we define the *pseudo* TPGs for obtaining *locally-optimal* SVM equivalence $\widetilde{\boldsymbol{W}}^{\text{svm}}$ and cyclic component $\widetilde{\boldsymbol{W}}^{\text{fin}}$ as follows:

1. Given any dataset $\texttt{DSET}$, consider GD solution $\boldsymbol{W}_{\text{gd}}$. For each training example $(\boldsymbol{X}, y) \in \texttt{DSET}$, let $\boldsymbol{s} = \mathbb{S}(\boldsymbol{X}\boldsymbol{W}_{\text{gd}}\bar{\boldsymbol{x}})$.

2. Construct TPGs based on $\boldsymbol{s}$ by adding directed edge $(x_{t_1} \to x_{t_2})$ to $\mathcal{G}^{(k)}$ if $\boldsymbol{s}_{t_1} > 0$, where k, x_t are the token IDs of last token and $\boldsymbol{x}_t$, respectively.

Different from the TPGs defined in Section 4.2.1 which is uniquely determined by the dataset and the ground truth labels, pseudo-TPGs build edges based on the tokens selected by GD solution $\boldsymbol{W}_{\text{gd}}$. To further investigate under which scenarios local convergence phenomenon exists, we consider the following cases and provide experimental evidence.

General loss function ℓ. In Section 4.3, we analyze the convergence performance of gradient descent when employing log-loss. As we have discussed, such loss guarantees the convexity of the problem and therefore, GD of attention weight (directional) converges to its global minima. Here, we investigate the performance of more general loss function, i.e., squared loss, and find empirical evidence of local convergence (Figures 4.6 and 4.7).

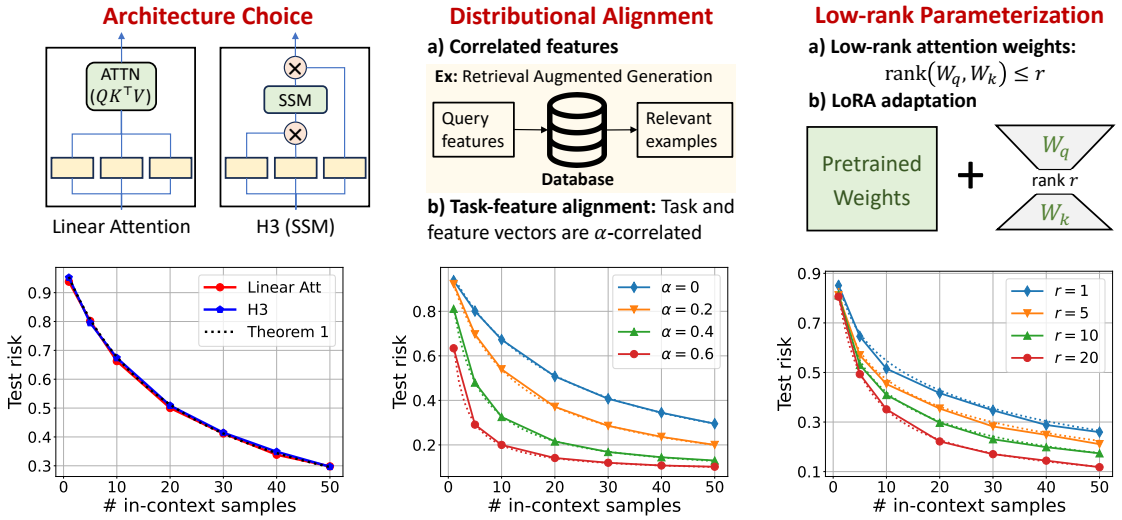
Extended linear head. In this work, GD experiments are conducted under Assumptions 4.1 and 4.2, which implies the convex training loss $\mathcal{L}(\boldsymbol{W})$ and global convergence performance. Now consider a more general linear head (i.e., Assumption 4.3). As have also been observed and discussed in [185, 186], Algo-GD can converge to a locally-optimal solution.

Figures 4.6 and 4.7 display our local convergence results where Fig. 4.6 employs squared loss, i.e. $\ell(u) = (1-u)^2$ and Fig. 4.7 utilizes cross-entropy loss. Both apply general head

following Assumption 4.3. Similar to Fig. 4.4, we present (directional) convergence performance of $\boldsymbol{W}$ towards $\boldsymbol{W}^{\mathrm{mm}}$ and $\boldsymbol{W}^{\mathrm{fin}}$. Results indicate that instead of converging to the global solution (blue curves), attention weights trained via GD align more closely with the locally-optimal SVM solution defined via the pseudo TPGs constructed by $\boldsymbol{W}_{\mathrm{gd}}$ (orange curves). In Fig. 4.6b, the norm difference to $\widetilde{\boldsymbol{W}}^{\mathrm{fin}}$ remains zero, indicating that all SCCs in the pseudo TPGs are singleton and GD optimizes attention weights towards selecting one token per sequence. While in Fig. 4.7, multiple tokens can be selected by $\boldsymbol{W}_{\mathrm{gd}}$. Note that in Fig. 4.7b, the norm of difference does not end with zero value on average. The potential explanations can be: Due to the non-convexity of training loss, training $\boldsymbol{W}$ with GD may not fully capture its RP solution $\widetilde{\boldsymbol{W}}^{\mathrm{fin}}$ over the cyclic subspace, and general classification head induces correlation among tokens, leading the attention mechanism to generate more intricate composed tokens. Nevertheless, our empirical results indicate that $\boldsymbol{W}$ more closely aligns with the local $\widetilde{\boldsymbol{W}}^{\mathrm{fin}}$ within its cyclic subspace. We defer a rigorous definition of local $\widetilde{\boldsymbol{W}}^{\mathrm{fin}}$ and guarantees related to gradient descent for future exploration. Experimental details are deferred to the appendix.

CHAPTER 5

Optimization Landscape of Linear Attention and State Space Models



(a) Linear attention=H3 (b) RAG with α correlation (c) LoRA

Figure 5.1: We investigate the optimization landscape of in-context learning from the lens of architecture choice, the role of distributional alignment, and low-rank parameterization. The empirical performance (solid curves) are aligned with our theoretical results (dotted curves) from Section 5.3. More experimental details and discussion are deferred to Section 5.4.

5.1 Introduction

In-context learning (ICL) ability presents an important research avenue to develop stronger theoretical and mechanistic understanding of large language models. There has been significant recent interest in demystifying ICL through the lens of function approximation [125], Bayesian inference [71, 137, 208], and learning and optimization theory [3, 52, 129, 214]. The latter is concerned with understanding the optimization landscape of ICL, which is also

crucial for understanding the generalization properties of the model. A notable result in this direction is the observation that linear attention models [3, 172, 196] implement *preconditioned gradient descent* (PGD) during ICL [3, 130]. While this line of works provide a fresh perspective to ICL, the existing studies do not address many questions arising from real-life applications nor provide guiding principles for various ICL setups motivated by practical considerations.

To this aim, we revisit the theoretical exploration of ICL with linear data model where we feed an in-context prompt containing n examples $(\boldsymbol{x}_i, y_i = \boldsymbol{x}_i^\top \boldsymbol{\beta} + \xi_i)_{i=1}^n \subset \mathbb{R}^d \times \mathbb{R}$ and a test instance or query $\boldsymbol{x}_{n+1} \in \mathbb{R}^d$ to the model, with d being the feature dimension, $\boldsymbol{\beta} \in \mathbb{R}^d$ being the task weight vector, and $(\xi_i)_{i=1}^n$ denoting the noise in individual labels. Given the in-context prompt, the model is tasked to predict $\hat{y}_{n+1}$ – an estimate for $y_{n+1} = \boldsymbol{x}_{n+1}^\top \boldsymbol{\beta} + \xi_{n+1}$. We aim to provide answers to the following questions by exploring the loss landscape of ICL:

(Q1) Is the ability to implement gradient-based ICL unique to (linear) attention? Can alternative sequence models implement richer algorithms beyond PGD?

(Q2) In language modeling, ICL often works well with few-shot samples whereas standard linear estimation typically requires $\mathcal{O}(d)$ samples. How can we reconcile this discrepancy between classical learning and ICL?

(Q3) To our knowledge, existing works assume linear-attention is fully parameterized, i.e., key and query projections $\boldsymbol{W}_k, \boldsymbol{W}_q \in \mathbb{R}^{d \times d}$. What happens when they are low-rank? What happens when there is distribution shift between training and test in-context prompts and we use LoRA [81] for adaptation?

In this work, we conduct a careful investigation of these questions. Specifically, we focus on ICL with 1-layer models and make the following contributions:

(A1) We jointly investigate the landscape of linear attention and H3 [57], a widely popular state-space model (SSM). We prove that under correlated design, both models implement 1-step PGD (c.f. Proposition 5.1) and the alignments in Fig. 5.1a verify that where the dotted curve represents the theoretical PGD result derived from Theorem 5.1. Our analysis reveals that the gating mechanism in H3 imitates attention. We also empirically show that H3 has the advantage of implementing sample-weighting which allows it to outperform linear attention in temporally-heterogeneous problem settings in Appendix D.4.

(A2) Proposition 5.1 allows for task and features to be correlated to each other as long as odd moments are zero. Through this, we can assess the impact of distributional alignment on the sample complexity of ICL. Specifically, we characterize the performance

of *Retrieval Augmented Generation* (RAG) (c.f. Theorem 5.2 and Fig. 5.1b) and *Task-Feature Alignment* (c.f. Theorem 5.3), where the in-context examples are α-correlated with either the query or the task vector. For both settings, we prove that alignment amplifies the *effective sample size* of ICL by a factor of $\alpha^2 d+1$, highlighting that aligned data are crucial for the success of ICL in few-shot settings.

(A3) We show that, under low-rank parameterization, optimal attention-weights still implements PGD according to the truncated eigenspectrum of the fused task-feature covariance (see Section 5.3.2). We similarly derive risk upper bounds for LoRA adaptation (c.f. Eq. (5.14) and Fig. 5.1c), and show that, these bounds accurately predict the empirical performance.

5.2 Preliminaries and Problem Setup

Notation. Let bold lowercase and uppercase letters (e.g., $\boldsymbol{x}$ and $\boldsymbol{X}$) represent vectors and matrices, respectively. The symbol $\odot$ is defined as the element-wise (Hadamard) product, and $*$ denotes the convolution operator. $\mathbf{1}_d$ and $\mathbf{0}_d$ denote the d-dimensional all-ones and all-zeros vectors, respectively; and $\boldsymbol{I}_d$ denotes the identity matrix of dimension $d \times d$. Additionally, let $\mathtt{tr}(\boldsymbol{W})$ denote the trace of the square matrix $\boldsymbol{W}$.

As mentioned earlier, we study the optimization landscapes of 1-layer linear attention [92, 172] and H3 [57] models when training with prompts containing in-context data following a linear model. We construct the input in-context prompt similar to [3, 129, 214] as follows.

Linear data distribution. Let $(\boldsymbol{x}, y) \in \mathbb{R}^d \times \mathbb{R}$ be a (feature, label) pair generated by a d-dimensional linear model parameterized by $\boldsymbol{\beta} \in \mathbb{R}^d$, i.e., $y = \boldsymbol{x}^\top \boldsymbol{\beta} + \xi$, where $\boldsymbol{x}$ and $\boldsymbol{\beta}$ are feature and task vectors, and ξ is the label noise. Given demonstrations $(\boldsymbol{x}_i, y_i)_{i=1}^{n+1}$ sampled from a single $\boldsymbol{\beta}$, define the input in-context prompt

$$\boldsymbol{Z} = [\boldsymbol{z}_1 \ \ldots \ \boldsymbol{z}_n \ \boldsymbol{z}_{n+1}]^\top = \begin{bmatrix} \boldsymbol{x}_1 & \ldots & \boldsymbol{x}_n & \boldsymbol{x}_{n+1} \\ y_1 & \ldots & y_n & 0 \end{bmatrix}^\top \in \mathbb{R}^{(n+1)\times(d+1)}. \tag{5.1}$$

Here, we set $\boldsymbol{z}_i = \begin{bmatrix} \boldsymbol{x}_i \\ y_i \end{bmatrix}$ for $i \le n$ and the last/query token $\boldsymbol{z}_{n+1} = \begin{bmatrix} \boldsymbol{x}_{n+1} \\ 0 \end{bmatrix}$. Then, given $\boldsymbol{Z}$, the goal of the model is to predict the correct label y_{n+1} corresponding to $\boldsymbol{x}_{n+1}$. For cleaner notation, when it is clear from context, we drop the subscript $n+1$ and set $\boldsymbol{x} = \boldsymbol{x}_{n+1},\ \boldsymbol{z} = \boldsymbol{z}_{n+1}$. Different from the previous work [3, 129, 214, 130] where $(\boldsymbol{x}_i)_{i=1}^{n+1}$ and $\boldsymbol{\beta}$ are assumed

to be independent, our analysis focuses on a more general linear setting that captures the dependency between $(\boldsymbol{x}_i)_{i=1}^{n+1}$ and $\boldsymbol{\beta}$.

Model architectures. To start with, we first review the architectures of both Transformer and state-space model (SSM). Similar to the previous work [196, 3, 129, 214] and to simplify the model structure, we focus on single-layer models and omit the nonlinearity, e.g., softmax operation and MLP activation, from the Transformer. Given the input prompt $\boldsymbol{Z} \in \mathbb{R}^{(n+1)\times(d+1)}$ in (5.1), which can be treated as a sequence of $(d+1)$-dimensional tokens, the single-layer linear attention att and H3-like single-layer SSM ssm are denoted by

$$\mathrm{att}(\boldsymbol{Z}) = (\boldsymbol{Z}\boldsymbol{W}_q\boldsymbol{W}_k^\top\boldsymbol{Z}^\top)\boldsymbol{Z}\boldsymbol{W}_v \tag{5.2a}$$

$$\mathrm{ssm}(\boldsymbol{Z}) = ((\boldsymbol{Z}\boldsymbol{W}_q) \odot ((\boldsymbol{Z}\boldsymbol{W}_k \odot \boldsymbol{Z}\boldsymbol{W}_v) * \boldsymbol{f})) \tag{5.2b}$$

where $\boldsymbol{W}_k,\ \boldsymbol{W}_q,\ \boldsymbol{W}_v \in \mathbb{R}^{(d+1)\times(d+1)}$ denote the key, query and value weight matrices, respectively. In (5.2b), the parameter $\boldsymbol{f} \in \mathbb{R}^{n+1}$ is a 1-D convolutional filter that mixes tokens. The Hadamard product $\odot$ is the gating mechanism [41] between key and query channels, which is crucial for attention-like feature creation. Thus, (5.2b) is more generally a gated-convolution layer. For $\boldsymbol{f}$ only, we use indexing $\boldsymbol{f} = [f_0\ \ldots\ f_n]^\top \in \mathbb{R}^{n+1}$ and given any vector $\boldsymbol{a}$, denote convolution output $(\boldsymbol{a} * \boldsymbol{f})_i = \sum_{j=1}^{i} f_{i-j}a_j$. Note that our notation slightly differs from the original H3 model [57] in two ways:

1. SSMs provide efficient parameterization of $\boldsymbol{f}$ which would otherwise grow with sequence length. In essence, H3 utilizes a linear state-space model $\boldsymbol{s}_i = \boldsymbol{A}\boldsymbol{s}_{i-1} + \boldsymbol{B}u_i$ and $y_i = \boldsymbol{C}\boldsymbol{s}_i$ with parameters $(\boldsymbol{A} \in \mathbb{R}^{d\times d}, \boldsymbol{B} \in \mathbb{R}^{d\times 1}, \boldsymbol{C} \in \mathbb{R}^{1\times d})$ from which the filter $\boldsymbol{f}$ is obtained via the impulse response $f_i = \boldsymbol{C}\boldsymbol{A}^i\boldsymbol{B}$ for $i \geq 0$. Here d is the state dimension and, in practice, $\boldsymbol{A}$ is chosen to be diagonal. Observe that, setting $d = 1$ and $\boldsymbol{A} = \rho, \boldsymbol{C} = \boldsymbol{B} = 1$, SSM reduces to the exponential smoothing $f_i = \rho^i$ for $i \geq 0$. Thus, H3 also captures the all-ones filter as a special instance. As we show in Proposition 5.1, this simple filter is optimal under independent data model and exactly imitates linear attention. Note that, utilizing a filter $\boldsymbol{f}$ as in (5.2b) is strictly more expressive than the SSM as it captures all possible impulse responses.

2. H3 also applies a shift SSM to the key embeddings to enable the retrieval of the local context around associative recall hits. We opted not to incorporate this shift operator in our model. This is because unless the features of the neighboring tokens are correlated (which is not the case for the typical independent data model), the entry-wise products between values and shifted keys will have zero mean and be redundant for the final prediction.

We note that we conduct all empirical evaluations with the original H3 model, which displays exact agreement with our theory formalized for (5.6b), further validating our modeling choice.

5.2.1 In-context Linear Estimation

We will next study the algorithms that can be implemented by the single-layer attention and state-space models. Through this, we will show that training att and ssm with linear ICL data is equivalent to the prediction obtained from one step of optimally-*preconditioned gradient descent* (PGD) and *sample-weighted preconditioned gradient descent* (WPGD), respectively. We will further show that under mild assumption, the optimal sample weighting for ssm (e.g., $\boldsymbol{f}$) is an all-ones vector and therefore, establishing the equivalence among PGD, att, and ssm.

Background: 1-step gradient descent. Consider minimizing squared loss and solving linear regression using one step of PGD and WPGD. Given n samples $(\boldsymbol{x}_i, y_i)_{i=1}^n$, define

$$\boldsymbol{X} = [\boldsymbol{x}_1 \ \cdots \ \boldsymbol{x}_n]^\top \in \mathbb{R}^{n\times d} \quad \text{and} \quad \boldsymbol{y} = [y_1 \ \cdots \ y_n]^\top \in \mathbb{R}^n.$$

Starting from $\boldsymbol{\beta}_0 = \boldsymbol{0}_d$ and letting $\eta = 1/2$ be the step size, a single-step GD preconditioned with weights $\boldsymbol{W}$ returns prediction

$$\hat{y} = \boldsymbol{x}^\top \boldsymbol{W} \boldsymbol{X}^\top \boldsymbol{y} := g_{\text{pgd}}(\boldsymbol{Z}), \tag{5.3}$$

and a single-step *sample-weighted* GD given weights $\boldsymbol{\omega} \in \mathbb{R}^n$ and $\boldsymbol{W} \in \mathbb{R}^{d\times d}$ returns prediction

$$\hat{y} = \boldsymbol{x}^\top \boldsymbol{W} \boldsymbol{X}^\top (\boldsymbol{\omega} \odot \boldsymbol{y}) := g_{\text{wpgd}}(\boldsymbol{Z}), \tag{5.4}$$

where $\boldsymbol{Z}$ is defined in (5.1) consisting of $\boldsymbol{X}, \boldsymbol{y}$ and $\boldsymbol{x}$. Our goal is to find the optimal $\boldsymbol{W}$, as well as $\boldsymbol{\omega}$ in (5.4) that minimize the population risks defined as follows.

$$\min_{\boldsymbol{W}} \mathcal{L}_{\text{pgd}}(\mathcal{W}) \quad \text{where} \quad \mathcal{L}_{\text{pgd}}(\mathcal{W}) = \mathbb{E}\left[(y - g_{\text{pgd}}(\boldsymbol{Z}))^2\right], \tag{5.5a}$$

$$\min_{\boldsymbol{W},\boldsymbol{\omega}} \mathcal{L}_{\text{wpgd}}(\mathcal{W}) \quad \text{where} \quad \mathcal{L}_{\text{wpgd}}(\mathcal{W}) = \mathbb{E}\left[(y - g_{\text{wpgd}}(\boldsymbol{Z}))^2\right]. \tag{5.5b}$$

Here, the expectation is over the randomness in $(\boldsymbol{x}_i, \xi_i)_{i=1}^{n+1}$ and $\boldsymbol{\beta}$, and we use $\mathcal{W}$ to represent the set of corresponding trainable parameters. The search spaces for $\boldsymbol{\omega}$ and $\boldsymbol{W}$ are $\mathbb{R}^n$ and $\mathbb{R}^{d\times d}$, respectively.

As per (5.2), given input prompt $\boldsymbol{Z} \in \mathbb{R}^{(n+1)\times(d+1)}$, either of the underlying models

outputs a $(n+1)$-length sequence. Note that the label for the query $\boldsymbol{x} = \boldsymbol{x}_{n+1}$ is excluded from the prompt $\boldsymbol{Z}$. Similar to [3, 129], we consider a training objective with a causal mask to ensure inputs cannot attend to their own labels and training can be parallelized. Let $\boldsymbol{Z}_0 = [\boldsymbol{z}_1 \ \dots \ \boldsymbol{z}_n \ 0]^\top$ be the features post-causal masking at time/index $n+1$. Given weights $\boldsymbol{W}_k, \boldsymbol{W}_q, \boldsymbol{W}_v$ and the filter $\boldsymbol{f}$ for ssm, predictions at the query token $\boldsymbol{z} = \begin{bmatrix} \boldsymbol{x} \\ 0 \end{bmatrix}$ take the following forms following sequence-to-sequence mappings in (5.2):

$$
\begin{aligned}
g_{\text{att}}(\boldsymbol{Z}) &= (\boldsymbol{z}^\top \boldsymbol{W}_q \boldsymbol{W}_k^\top \boldsymbol{Z}_0^\top) \boldsymbol{Z}_0 \boldsymbol{W}_v \boldsymbol{v}, \\
g_{\text{ssm}}(\boldsymbol{Z}) &= \left((\boldsymbol{z}^\top \boldsymbol{W}_q)^\top \odot ((\boldsymbol{Z}_0 \boldsymbol{W}_k \odot \boldsymbol{Z}_0 \boldsymbol{W}_v) * \boldsymbol{f})_{n+1} \right) \boldsymbol{v},
\end{aligned}
$$

where $\boldsymbol{v} \in \mathbb{R}^{d+1}$ is the linear prediction head and $((\boldsymbol{Z}_0 \boldsymbol{W}_k \odot \boldsymbol{Z}_0 \boldsymbol{W}_v) * \boldsymbol{f})_{n+1}$ returns the last row of the convolution output. Note that SSM can implement the mask by setting $f_0 = 0$. Now consider the meta learning setting and select loss function to be the squared loss, same as in (5.5). Thus, the objectives for both models take the following forms.

$$
\min_{\boldsymbol{W}_k, \boldsymbol{W}_q, \boldsymbol{W}_v, \boldsymbol{v}} \mathcal{L}_{\text{att}}(\mathcal{W}) \quad \text{where} \quad \mathcal{L}_{\text{att}}(\mathcal{W}) = \mathbb{E}\left[(y - g_{\text{att}}(\boldsymbol{Z}))^2 \right], \tag{5.6a}
$$

$$
\min_{\boldsymbol{W}_k, \boldsymbol{W}_q, \boldsymbol{W}_v, \boldsymbol{v}, \boldsymbol{f}} \mathcal{L}_{\text{ssm}}(\mathcal{W}) \quad \text{where} \quad \mathcal{L}_{\text{ssm}}(\mathcal{W}) = \mathbb{E}\left[(y - g_{\text{ssm}}(\boldsymbol{Z}))^2 \right]. \tag{5.6b}
$$

Here, similarly, the expectation subsumes the randomness of $(\boldsymbol{x}_i, \xi_i)_{i=1}^{n+1}$ and $\boldsymbol{\beta}$ and $\mathcal{W}$ represents the set of trainable parameters. The search space for matrices $\boldsymbol{W}_k$, $\boldsymbol{W}_q$, $\boldsymbol{W}_v$ is $\mathbb{R}^{(d+1)\times(d+1)}$, for head $\boldsymbol{v}$ is $\mathbb{R}^{d+1}$, and for $\boldsymbol{f}$ is $\mathbb{R}^{n+1}$.

Note that for all the optimization methods (c.f. (5.5), (5.6)), to simplify the analysis, we train the models without capturing additional bias terms. Therefore, in the following, we introduce the centralized data assumptions such that the models are trained to make unbiased predictions.

To begin with, a cross moment of random variables is defined as the expectation of a monomial of these variables, with the order of the cross moment being the same as order of the monomial. For example, $\mathbb{E}[\boldsymbol{x}^\top \boldsymbol{W} \boldsymbol{\beta}]$ is a sum of cross-moments of order 2. Then, it motivates the following data assumptions.

Assumption 5.1 *All cross moments of the entries of $(\boldsymbol{x}_i)_{i=1}^{n+1}$ and $\boldsymbol{\beta}$ with odd orders are zero.*

Assumption 5.2 *The label noise $(\xi_i)_{i=1}^{n+1}$ are independent of $(\boldsymbol{x}_i)_{i=1}^{n+1}$ and $\boldsymbol{\beta}$, and their cross moments with odd orders are zero.*

Note that compared to [3, 129, 214], Assumption 5.1 is more general which also subsumes the

dependent distribution settings. In this work, we consider the following three linear models (omitting noise) satisfying Assumption 5.1. Let $\boldsymbol{\Sigma}_{\boldsymbol{\beta}}, \boldsymbol{\Sigma}_{\boldsymbol{x}} \in \mathbb{R}^{d\times d}$ represent the task and feature covariance matrices for independent data, and let $0 \leq \alpha \leq 1$ be the correlation level when considering data dependency. More specific discussions are deferred to Section 5.3.

- Independent task and data: $\boldsymbol{\beta} \sim \mathcal{N}(0, \boldsymbol{\Sigma}_{\boldsymbol{\beta}}),\ \boldsymbol{x}_i \sim \mathcal{N}(0, \boldsymbol{\Sigma}_{\boldsymbol{x}}),$ for all $1 \leq i \leq n+1$.
- Retrieval augmented generation: $\boldsymbol{\beta}, \boldsymbol{x} \sim \mathcal{N}(0, \boldsymbol{I}_d),\ \boldsymbol{x}_i \,\big|\, \boldsymbol{x} \sim \mathcal{N}(\alpha\boldsymbol{x}, (1-\alpha^2)\boldsymbol{I}_d),$ for all $1 \leq i \leq n$.
- Task-feature alignment: $\boldsymbol{\beta} \sim \mathcal{N}(0, \boldsymbol{I}_d),\ \boldsymbol{x}_i \,\big|\, \boldsymbol{\beta} \sim \mathcal{N}(\alpha\boldsymbol{\beta}, \boldsymbol{I}_d),$ for all $1 \leq i \leq n+1$.

Next, we introduce the following result which establishes the equivalence among optimizing 1-layer linear attention (c.f. (5.6a)), 1-layer H3 (c.f. (5.6b)), and 1-step gradient descent (c.f. (5.5)).

Proposition 5.1 *Suppose Assumptions 5.1 and 5.2 hold. Consider the objectives as defined in* (5.5) *and* (5.6)*, and let* $\mathcal{L}^\star_{pgd}$, $\mathcal{L}^\star_{wpgd}$, $\mathcal{L}^\star_{att}$, *and* $\mathcal{L}^\star_{ssm}$ *be their optimal risks, respectively. Then,*

$$\mathcal{L}^\star_{pgd} = \mathcal{L}^\star_{att} \quad \textit{and} \quad \mathcal{L}^\star_{wpgd} = \mathcal{L}^\star_{ssm}.$$

Additionally, if the examples $(\boldsymbol{x}_i, y_i)_{i=1}^n$ *follow the same distribution and are conditionally independent given* $\boldsymbol{x}, \boldsymbol{\beta}$*, then SSM/H3 can achieve the optimal loss using the all-ones filter and* $\mathcal{L}^\star_{pgd} = \mathcal{L}^\star_{ssm}$.

We defer the proof to Appendix D.1.1. Proposition 5.1 establishes that analyzing the optimization landscape of ICL for both single-layer linear attention and the H3 model can be effectively reduced to examining the behavior of a one-step PGD algorithm. Notably, under the independent, RAG and task-feature alignment data settings discussed above, examples $(\boldsymbol{x}_i, y_i)_{i=1}^n$ are independently sampled given $\boldsymbol{x}$ and $\boldsymbol{\beta}$, and we therefore conclude that $\mathcal{L}^\star_{\mathrm{pgd}} = \mathcal{L}^\star_{\mathrm{att}} = \mathcal{L}^\star_{\mathrm{ssm}}$. Leveraging this result, the subsequent section of the paper concentrate on addressing (5.5a), taking into account various linear data distributions.

While Proposition 5.1 demonstrates the equivalence of optimal losses, we also study the uniqueness and equivalence of optimal prediction functions. To this end, we analyze the strong convexity of $\mathcal{L}_{\mathrm{pgd}}(\mathcal{W})$ and derive the subsequent lemmas.

Lemma 5.1 *Suppose Assumption 5.2 holds and let* $\boldsymbol{\xi} = [\xi_1\ \xi_2\ \cdots\ \xi_n]^\top$. *Then the loss* $\mathcal{L}_{pgd}(\mathcal{W})$ *in* (5.5a) *is strongly-convex if and only if* $\mathbb{E}[(\boldsymbol{x}^\top \boldsymbol{W}\boldsymbol{X}^\top \boldsymbol{X}\boldsymbol{\beta})^2] + \mathbb{E}[(\boldsymbol{x}^\top \boldsymbol{W}\boldsymbol{X}^\top \boldsymbol{\xi})^2]$ *is strongly-convex. Additionally, let* $g^\star_{pgd}$, $g^\star_{att}$ *be the optimal prediction functions of* (5.5a) *and* (5.6a)*. Then under the conditions of Assumptions 5.1 and 5.2, and the strong convexity,* $g^\star_{pgd} = g^\star_{att}$.

Lemma 5.2 *Suppose that the label noise $(\xi_i)_{i=1}^n$ are i.i.d., zero-mean, variance σ^2 and independent of everything else, and that there is a decomposition $\boldsymbol{x} = \boldsymbol{x}_1 + \boldsymbol{x}_2$, $\boldsymbol{X} = \boldsymbol{X}_1 + \boldsymbol{X}_2$, and $\boldsymbol{\beta} = \boldsymbol{\beta}_1 + \boldsymbol{\beta}_2$ such that either of the following holds*

- *$\sigma > 0$, and $(\boldsymbol{x}_1, \boldsymbol{X}_1)$ have full rank covariance and are independent of each other and $(\boldsymbol{x}_2, \boldsymbol{X}_2)$.*
- *$(\boldsymbol{x}_1, \boldsymbol{\beta}_1, \boldsymbol{X}_1)$ have full rank covariance and are independent of each other and $(\boldsymbol{x}_2, \boldsymbol{\beta}_2, \boldsymbol{X}_2)$.*

Then, the loss $\mathcal{L}_{pgd}(\mathcal{W})$ in (5.5a) is strongly-convex.

As mentioned above, in this work, we study three specific linear models: with general independent, RAG-related, and task-feature alignment data. Note that for all the three cases, according to Proposition 5.1, we have $\mathcal{L}^\star_{\text{pgd}} = \mathcal{L}^\star_{\text{att}} = \mathcal{L}^\star_{\text{ssm}}$. Additionally, the second claim in Lemma 5.2 holds, and $\mathcal{L}_{\text{pgd}}(\mathcal{W})$ is strongly convex. Therefore, following Lemma 5.1, we have $g^\star_{\text{pgd}} = g^\star_{\text{att}}$. Thanks to the equivalence among PGD, att, and ssm, in the next section, we focus on the solution of objective (5.5a) under different scenarios, which will reflect the optimization landscapes of att and ssm models.

5.3 Main Results

In light of Proposition 5.1, optimizing a single layer linear-attention or H3 model is equivalent to solving the objective (5.5a). Therefore, in this section, we examine the properties of the one-step PGD in (5.5a). To this end, we consider multiple problem settings, including distinct data distributions and low-rank training. The latter refers to the scenario where the key and query matrices have rank restrictions, e.g., $\boldsymbol{W}_k, \boldsymbol{W}_q \in \mathbb{R}^{(d+1)\times r}$, as well as LoRA-tuning when adapting the model under distribution shift.

5.3.1 Analysis of Linear Data Models

We first consider the standard independent data setting. We will then examine correlated designs.

Independent data model. Let $\boldsymbol{\Sigma}_{\boldsymbol{x}}$ and $\boldsymbol{\Sigma}_{\boldsymbol{\beta}}$ be the covariance matrices of the input feature and task vectors, respectively, and $\sigma \geq 0$ be the noise level. We assume

$$\boldsymbol{\beta} \sim \mathcal{N}(0, \boldsymbol{\Sigma}_{\boldsymbol{\beta}}), \quad \boldsymbol{x}_i \sim \mathcal{N}(0, \boldsymbol{\Sigma}_{\boldsymbol{x}}), \quad \xi_i \sim \mathcal{N}(0, \sigma^2), \quad 1 \leq i \leq n+1 \tag{5.7}$$

and the label is obtained via $y_i = \boldsymbol{x}_i^\top \boldsymbol{\beta} + \xi_i$. Our following result characterizes the optimal solution of (5.5a). Note that the data generated from (5.7) satisfies the conditions in

Proposition 5.1. Therefore, the same results can be applied to both linear-attention and H3 models.

Theorem 5.1 *Consider independent linear data as defined in* (5.7)*, and suppose the covariance matrices* $\boldsymbol{\Sigma}_{\boldsymbol{x}}, \boldsymbol{\Sigma}_{\boldsymbol{\beta}}$ *are full rank. Recap the objective from* (5.5a) *and let* $\boldsymbol{W}_\star := \arg\min_{\boldsymbol{W}} \mathcal{L}_{pgd}(\boldsymbol{W})$*, and* $\mathcal{L}_\star = \mathcal{L}_{pgd}(\boldsymbol{W}_\star)$*. Additionally, let* $\boldsymbol{\Sigma} = \boldsymbol{\Sigma}_{\boldsymbol{x}}^{1/2}\boldsymbol{\Sigma}_{\boldsymbol{\beta}}\boldsymbol{\Sigma}_{\boldsymbol{x}}^{1/2}$ *and* $M = \boldsymbol{tr}(\boldsymbol{\Sigma}) + \sigma^2$*. Then* $\boldsymbol{W}^{mm}$ *and* $\mathcal{L}_\star$ *satisfy*

$$\boldsymbol{W}_\star = \boldsymbol{\Sigma}_{\boldsymbol{x}}^{-1/2}\bar{\boldsymbol{W}}_\star\boldsymbol{\Sigma}_{\boldsymbol{x}}^{-1/2} \qquad \textit{and} \qquad \mathcal{L}_\star = M - n\,\boldsymbol{tr}\left(\boldsymbol{\Sigma}\bar{\boldsymbol{W}}_\star\right), \tag{5.8}$$

where we define $\bar{\boldsymbol{W}}_\star = \left((n+1)\boldsymbol{I}_d + M\boldsymbol{\Sigma}^{-1}\right)^{-1}$.

Corollary 5.1 *Consider noiseless i.i.d. linear data where* $\boldsymbol{\Sigma}_{\boldsymbol{x}} = \boldsymbol{\Sigma}_{\boldsymbol{\beta}} = \boldsymbol{I}_d$ *and* $\sigma = 0$*. Then, the objective in* (5.5a) *returns*

$$\boldsymbol{W}_\star = \frac{1}{n+d+1}\boldsymbol{I}_d \qquad \textit{and} \qquad \mathcal{L}_\star = d - \frac{nd}{n+d+1}.$$

See Appendix D.2.2 for proofs. Note that Theorem 5.1 is consistent with prior work [3, Theorem 1] when specialized to isotropic task covariance, i.e., $\boldsymbol{\Sigma}_{\boldsymbol{\beta}} = \boldsymbol{I}_d$. However, their result is limited as the features and task are assumed to be independent. This prompts us to ask: *What is the optimization landscape with correlated in-context samples?* Toward this, we consider the following RAG-inspired and task-feature alignment models, where Assumptions 5.1 and 5.2 continue to hold and Proposition 5.1 applies.

Retrieval augmented generation. To provide a statistical model of the practical RAG approaches, given the query vector $\boldsymbol{x}_{n+1} = \boldsymbol{x}$, we propose to draw ICL demonstrations that are similar to $\boldsymbol{x}$ with the same shared task vector $\boldsymbol{\beta}$. Modeling feature similarity through the cosine angle, RAG should sample the ICL examples $\boldsymbol{x}_i$, $i \le n$, from the original feature distribution conditioned on the event $\cos(\boldsymbol{x}_i, \boldsymbol{x}) \ge \alpha$ where α is the similarity threshold. As an approximate proxy, under the Gaussian distribution model, we assume that $\boldsymbol{\beta} \sim \mathcal{N}(0, \boldsymbol{I}_d)$, $\boldsymbol{x} \sim \mathcal{N}(0, \boldsymbol{I}_d)$ and that RAG samples α-correlated demonstrations $(\boldsymbol{x}_i, y_i)_{i=1}^n$ as follows:

$$\boldsymbol{x}_i \;\big|\; \boldsymbol{x} \sim \mathcal{N}(\alpha\boldsymbol{x}, (1-\alpha^2)\boldsymbol{I}_d), \quad \xi_i \sim \mathcal{N}(0, \sigma^2) \quad \text{and} \quad y_i = \boldsymbol{x}_i^\top\boldsymbol{\beta} + \xi_i, \quad 1 \le i \le n. \tag{5.9}$$

Note that the above normalization ensures that the marginal feature distribution remains $\mathcal{N}(0, \boldsymbol{I}_d)$. The full analysis of RAG is provides in Appendix D.2.3. Specifically, when we carry out the analysis by assuming $\alpha = \mathcal{O}(1/\sqrt{d})$ and $d/n = \mathcal{O}(1)$ where $\mathcal{O}(\cdot)$ denotes proportionality, our derivation leads to the following result:

Theorem 5.2 *Consider linear model as defined in* (5.9). *Recap the objective from* (5.5a) *and let* $\boldsymbol{W}_\star := \arg\min_{\boldsymbol{W}} \mathcal{L}_{pgd}(\boldsymbol{W})$, *and* $\mathcal{L}_\star = \mathcal{L}_{pgd}(\boldsymbol{W}_\star)$. *Additionally, let* $\kappa = \alpha^2 d + 1$ *and suppose* $\alpha = \mathcal{O}(1/\sqrt{d})$, $d/n = \mathcal{O}(1)$ *and* d *is sufficiently large. Then* $\boldsymbol{W}^{mm}$ *and* $\mathcal{L}_\star$ *have approximate forms*

$$\boldsymbol{W}_\star \approx \frac{1}{\kappa n + d + \sigma^2} \boldsymbol{I}_d \qquad \text{and} \qquad \mathcal{L}_\star \approx d + \sigma^2 - \frac{\kappa n d}{\kappa n + d + \sigma^2}. \tag{5.10}$$

Here, (5.10) is reminiscent of Corollary 5.1 and has a surprisingly clean message. Observe that, $\alpha^2 d + 1$ is the dominant multiplier ahead of n in both equations. Thus, we deduce that, RAG model follows the same error bound as the independent data model, however, its sample size is amplified by a factor of $\alpha^2 d + 1$. $\alpha = 0$ reduces to the result of Corollary 5.1 whereas we need to set $\alpha = \mathcal{O}(1/\sqrt{d})$ for constant amplification. When $\alpha = 1$, RAG achieves the approximate risk $\mathcal{L}_\star \approx 2 + \sigma^2$, where the constant bias is due to the higher order moments (e.g., the 4'th and 6'th moments) of the standard Gaussian distribution. As d increases, the normalized loss $\mathcal{L}_\star / d \to 0$. The full analysis of its optimal solution $\boldsymbol{W}_\star$ and loss $\mathcal{L}_\star$ are deferred Theorem D.1 in Appendix D.2.3.

Task-feature alignment. We also consider another dependent data setting where task and feature vectors are assumed to be correlated. This dataset model has the following motivation: In general, an LLM can generate any token within the vocabulary. However, once we specify the task (e.g. domain of the prompt), the LLM output becomes more deterministic and there are much fewer token candidates. For instance, if the task is "Country", "France" is a viable output compared to "Helium" and vice versa when the task is "Chemistry". Formally speaking, this can be formalized as the input $\boldsymbol{x}$ having a diverse distribution whereas it becomes more predictable conditioned on $\boldsymbol{\beta}$. Therefore, it can be captured through a linear model by making the conditional covariance of $\boldsymbol{x} \mid \boldsymbol{\beta}$ to be approximately low-rank. This formalism can be viewed as a *spectral alignment* between input and task, which is also well-established in deep learning both empirically and theoretically [109, 6, 24, 26]. Here, we consider such a setting where the shared task vector is sampled as standard Gaussian distribution $\boldsymbol{\beta} \sim \mathcal{N}(0, \boldsymbol{I}_d)$ and letting $\kappa = \alpha^2 d + 1$, we sample the α-correlated ICL demonstrations $(\boldsymbol{x}_i, y_i)_{i=1}^{n+1}$ as follows:

$$\boldsymbol{x}_i \mid \boldsymbol{\beta} \sim \mathcal{N}(\alpha\boldsymbol{\beta}, \boldsymbol{I}_d), \quad \xi_i \sim \mathcal{N}(0, \sigma^2) \quad \text{and} \quad y_i = \kappa^{-1/2} \boldsymbol{x}_i^\top \boldsymbol{\beta} + \xi_i, \quad 1 \le i \le n+1. \tag{5.11}$$

Above, $\kappa^{-1/2}$ is a normalization factor to ensure that label variance remains invariant to α. To keep the exposition cleaner, we defer the full analysis of its optimal solution $\boldsymbol{W}_\star$ and loss $\mathcal{L}_\star$ to Theorem D.2 in Appendix D.2.4. Similar to the RAG setting, by assuming $\alpha = \mathcal{O}(1/\sqrt{d})$ and $d/n = \mathcal{O}(1)$, we obtain the following results for the optimal parameter

and risk.

Theorem 5.3 *Consider linear model as defined in* (5.11)*. Recap the objective from* (5.5a) *and let* $\boldsymbol{W}_\star := \arg\min_{\boldsymbol{W}} \mathcal{L}_{pgd}(\boldsymbol{W})$*, and* $\mathcal{L}_\star = \mathcal{L}_{pgd}(\boldsymbol{W}_\star)$*. Additionally, given* $\kappa = \alpha^2 d + 1$ *and suppose* $\alpha = \mathcal{O}(1/\sqrt{d})$*,* $d/n = \mathcal{O}(1)$ *and* d *is sufficiently large. Then* $\boldsymbol{W}^{mm}$ *and* $\mathcal{L}_\star$ *have approximate forms*

$$\boldsymbol{W}_\star \approx \frac{1}{\kappa n + (d+\sigma^2)/\kappa} \boldsymbol{I}_d \qquad \text{and} \qquad \mathcal{L}_\star \approx d + \sigma^2 - \frac{\kappa n d}{\kappa n + (d+\sigma^2)/\kappa}. \tag{5.12}$$

Similar to (5.10), (5.12) contains $\kappa = \alpha^2 + 1$ multiplier ahead of n, which reduces the in-context sample complexity and setting $\alpha = 0$ reduces to the results of Corollary 5.1.

5.3.2 Low-rank Parameterization and LoRA

In this section, we investigate training low-rank models, which assume $\boldsymbol{W}_k, \boldsymbol{W}_q \in \mathbb{R}^{(d+1)\times r}$ where r is the rank restriction. Equivalently, we consider objective (5.5a) under condition $\text{rank}(\boldsymbol{W}) = r$.

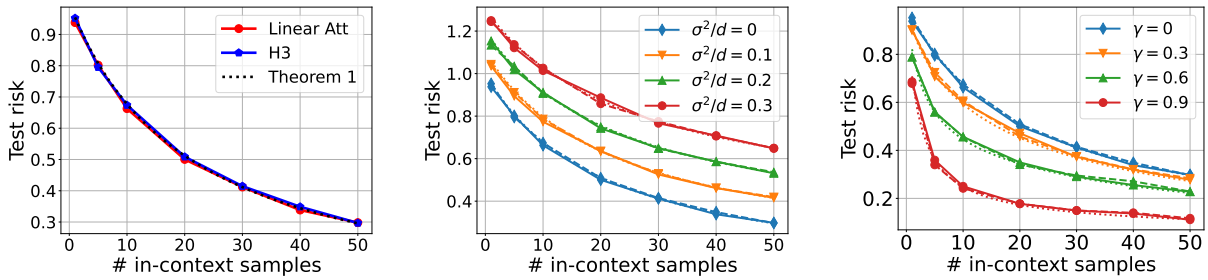
Lemma 5.3 *Consider independent linear data as defined in* (5.7)*. Recap the objective from* (5.5a) *and enforce rank*$(\boldsymbol{W}) \le r$ *and* $\boldsymbol{W}^\top = \boldsymbol{W}$*. Let* $\boldsymbol{\Sigma} = \boldsymbol{\Sigma}_{\boldsymbol{x}}^{1/2} \boldsymbol{\Sigma}_{\boldsymbol{\beta}} \boldsymbol{\Sigma}_{\boldsymbol{x}}^{1/2}$ *and* $M = \mathtt{tr}(\boldsymbol{\Sigma}) + \sigma^2$*. Denoting* λ_i *to be the* i*'th largest eigenvalue of* $\boldsymbol{\Sigma}$*, we have that*

$$\min_{rank(\boldsymbol{W})\le r, \boldsymbol{W}=\boldsymbol{W}^\top} \mathcal{L}(\boldsymbol{W}) = M - \sum_{i=1}^{r} \frac{n\lambda_i^2}{(n+1)\lambda_i + M}. \tag{5.13}$$

Note that $\mathtt{tr}(\boldsymbol{\Sigma}) = \sum_{i=1}^d \lambda_i$. Removing the rank constraint and considering noiseless data setting, this reduces to the following optimal risk $\mathcal{L}_\star = \sum_{i=1}^d \frac{\lambda_i + M}{n+1+M/\lambda_i}$. See Appendix D.3.1 for more details.

Impact of LoRA: Based on the above lemma, we consider the impact of LoRA for adapting the pretrained model to a new task distribution under jointly-diagonalizable old and new eigenvalues of $\boldsymbol{\Sigma}$, $\boldsymbol{\Sigma}^{new}$, $(\lambda_i)_{i=1}^d$, $(\lambda_i^{new})_{i=1}^d$. Consider adapting LoRA matrix to the combined key and value weights in attention, which reflects minimizing the population loss $\tilde{\mathcal{L}}(\boldsymbol{W}_{\text{lora}}) := \mathcal{L}(\boldsymbol{W} + \boldsymbol{W}_{\text{lora}})$ in (5.5a) with fixed $\boldsymbol{W}$. Suppose $\mathtt{tr}(\boldsymbol{\Sigma}) = \mathtt{tr}(\boldsymbol{\Sigma}^{new}) = M$, $\sigma = 0$ and $\boldsymbol{W}$ is jointly diagonalizable with $\boldsymbol{\Sigma}$, $\boldsymbol{\Sigma}^{new}$, then LoRA's risk is upper-bounded by

$$\min_{\text{rank}(\boldsymbol{W}_{\text{lora}})\le r} \tilde{\mathcal{L}}(\boldsymbol{W}_{lora}) \le \min_{|\mathcal{I}|\le r, \mathcal{I}\subset[d]} \left(\sum_{i\notin\mathcal{I}} \frac{\lambda_i + M}{n+1+M/\lambda_i} + \sum_{i\in\mathcal{I}} \frac{\lambda_i^{new} + M}{n+1+M/\lambda_i^{new}} \right). \tag{5.14}$$



(a) $\boldsymbol{\Sigma}_{\boldsymbol{x}} = \boldsymbol{\Sigma}_{\boldsymbol{\beta}} = \boldsymbol{I}_d$ and $\sigma = 0$ (b) Noisy label (c) Non-isotropic task

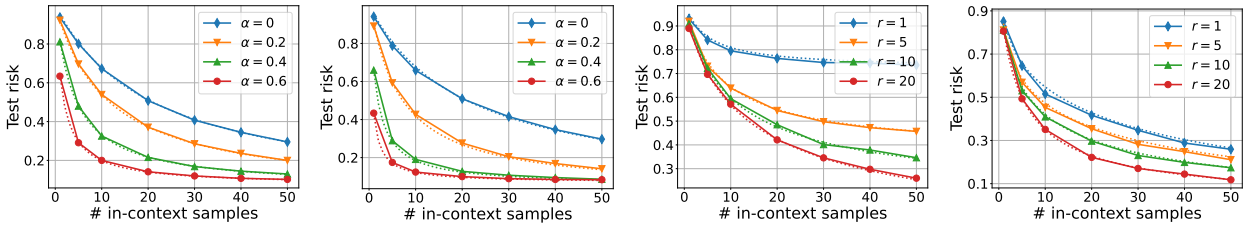
Figure 5.2: Empirical evidence validates Theorem 5.1 and Proposition 5.1. We train 1-layer linear attention and H3 models with prompts containing independent demonstrations following a linear model, and dotted curves are the theory curves following Eq. (5.8). **(a):** We consider noiseless i.i.d. setting where $\boldsymbol{\Sigma}_{\boldsymbol{x}} = \boldsymbol{\Sigma}_{\boldsymbol{\beta}} = \boldsymbol{I}_d$ and $\sigma = 0$, with results presented in red (attention) and blue (H3) solid curves. **(b):** We conduct noisy label experiments by choosing $\sigma \neq 0$. **(c):** Consider non-isotropic task by setting $\boldsymbol{\Sigma}_{\boldsymbol{\beta}} = \gamma \mathbf{1}\mathbf{1}^\top + (1-\gamma)\boldsymbol{I}_d$. Solid and dashed curves in (b) and (c) represent attention and H3 results, respectively. The alignments in (a), (b) and (c) show the equivalence between attention and H3, validating Theorem 5.1 and Proposition 5.1. More experimental details are discussed in Section 5.4.

Note that, the right hand side is provided assuming the optimal LoRA-updated model $\boldsymbol{W}_{\text{lora}}$ is also jointly diagonalizable with covariances $\boldsymbol{\Sigma}$, $\boldsymbol{\Sigma}^{new}$, and $\boldsymbol{W}$.

5.4 Experiments

We now conduct synthetic experiments to support our theoretical findings and further explore the behavior of different models of interest under different conditions. The experiments are designed to investigate various scenarios, including independent data, retrieval-augmented generation (RAG), task-feature alignment, low-rank parameterization, and LoRA adaption.

Experimental setting. We train 1-layer attention and H3 models for solving the linear regression ICL. As described in Section 5.2, we consider meta-learning setting where task parameter $\boldsymbol{\beta}$ is randomly generated for each training sequence. In all experiments, we set the dimension $d = 20$. Depending on the in-context length (n), different models are trained to make in-context predictions. We train each model for 10000 iterations with batch size 128 and Adam optimizer with learning rate 10^{-3}. Since our study focuses on the optimization landscape, and experiments are implemented via gradient descent, we repeat 20 model trainings from different initialization and results are presented as the minimal test risk among those 20 trails. In all the plots, theoretical predictions are obtained via the corresponding formulae presented in Section 5.3 and the test risks are normalized by the dimension d.



(a) RAG (b) Task-feature align. (c) Low-rank attention (d) LoRA adaptation

Figure 5.3: Distributional alignment and low-rank parameterization experiments. **(a)** and **(b)** show the ICL results using data generated via (5.9) and (5.11), respectively, by changing α from 0 to 0.6. In **(c)**, we train low-rank linear attention models by setting $\boldsymbol{W}_k, \boldsymbol{W}_q \in \mathbb{R}^{(d+1)\times r}$ and in **(d)**, we apply the low-rank LoRA adaptor, $\boldsymbol{W}_{\text{lora}} := \boldsymbol{W}_{\text{up}}\boldsymbol{W}_{\text{down}}^\top$ where $\boldsymbol{W}_{\text{up}}, \boldsymbol{W}_{\text{down}} \in \mathbb{R}^{(d+1)\times r}$, to pretrained linear attention models and adjust the LoRA parameters under different task distribution. Solid and dotted curves correspond to the linear attention and theoretical results (c.f. Section 5.3), respectively, and the alignments validate our theorems in Section 5.3. More experimental details are discussed in Section 5.4.

• **Equivalence among $\mathcal{L}^\star_{\text{pgd}}$, $\mathcal{L}^\star_{\text{att}}$ and $\mathcal{L}^\star_{\text{ssm}}$ (Figure 5.2).** To verify Proposition 5.1 as well as Theorem 5.1, we run random linear regression instances where in-context samples are generated obeying (5.7). Fig. 5.2a is identical to Fig. 5.1a where we set $\boldsymbol{\Sigma}_{\boldsymbol{x}} = \boldsymbol{\Sigma}_{\boldsymbol{\beta}} = \boldsymbol{I}_d$ and $\sigma = 0$. In Fig. 5.2b, set $\boldsymbol{\Sigma}_{\boldsymbol{x}} = \boldsymbol{\Sigma}_{\boldsymbol{\beta}} = \boldsymbol{I}$ and vary noise level σ^2 from 0 to $0.3 \times d$. In Fig. 5.2c, we consider noiseless labels, $\sigma = 0$, isotropic feature distribution $\boldsymbol{\Sigma}_{\boldsymbol{x}} = \boldsymbol{I}_d$ and set task covariance to be $\boldsymbol{\Sigma}_{\boldsymbol{\beta}} = \gamma \mathbf{1}\mathbf{1}^\top + (1-\gamma)\boldsymbol{I}_d$ by choosing γ in $\{0, 0.3, 0.6, 0.9\}$. Note that in Fig. 5.2c, we train a sufficient number of models (greater than 20) to ensure the optimal model is obtained. In all the figures, solid and dashed curves correspond to the ICL results from training 1-layer att and ssm models, respectively, and dotted curves are obtained from (5.8) in Theorem 5.1. The alignment of solid, dashed and dotted curves validates our Proposition 5.1 and Theorem 5.1.

• **Distributional alignment experiments (Figs. 5.3a&5.3b).** In Figs. 5.3a and 5.3b, we generate RAG and task-feature alignment data following (5.9) and (5.11), respectively, by setting $\sigma = 0$ and varying α from 0 to 0.6. Attention training results are displayed in solid curves, and we generate theory curve (dotted) via the $\mathcal{L}_\star$ formula as described in (D.23) in Appendix D.2.3 and (D.29) in Appendix D.2.4. The empirical alignments corroborate Theorems D.1 and D.2, further confirming that Proposition 5.1 is applicable to a broader range of real-world distributional alignment data.

• **Low-rank (Fig. 5.3c) and LoRA (Fig. 5.3d) experiments.** We also run simulations to verify our theoretical findings in Section 5.3.2. Consider the independent data

setting as described in (5.7). In Fig. 5.3c, we set $\boldsymbol{\Sigma}_{\boldsymbol{x}} = \boldsymbol{I}_d$, $\sigma = 0$ and task covariance to be diagonal with diagonal entries $c[1\ 2^{-1}\ \cdots\ d^{-1}]^\top$ for some normalization constant $c = d/\sum_{i=1}^{d} i^{-1}$, and parameterize the attention model using matrices $\boldsymbol{W}_k, \boldsymbol{W}_q \in \mathbb{R}^{(d+1)\times r}$ and vary r across the set $\{1, 5, 10, 20\}$. Results show that empirical (solid) and theoretical (dotted, c.f. (5.13)) curves overlap. In Fig. 5.3d, we implement two phases of training. *Phase 1:* Setting $\boldsymbol{\Sigma}_{\boldsymbol{x}} = \boldsymbol{\Sigma}_{\boldsymbol{\beta}} = \boldsymbol{I}_d$ and $\sigma = 0$, we pretrain the model with full rank parameters and obtain weights $\hat{\boldsymbol{W}}_k, \hat{\boldsymbol{W}}_q, \hat{\boldsymbol{W}}_v \in \mathbb{R}^{(d+1)\times(d+1)}$. *Phase 2:* We generate new examples with task covariance $\boldsymbol{\Sigma}_{\boldsymbol{\beta}}$ being a diagonal matrix with diagonal entries $c'[2^{-1}\ 2^{-2}\ \cdots\ 2^{-d}]^\top$ for some normalization constant $c' = d/\sum_{i=1}^{d} 2^{-i}$. Given the rank restriction r, we train additional LoRA parameters $\boldsymbol{W}_{\text{up}}, \boldsymbol{W}_{\text{down}} \in \mathbb{R}^{(d+1)\times r}$ where $\boldsymbol{W}_{\text{lora}} := \boldsymbol{W}_{\text{up}}\boldsymbol{W}_{\text{down}}^\top$ and (5.2a) becomes $\text{att}(\boldsymbol{Z}) = (\boldsymbol{Z}(\hat{\boldsymbol{W}}_q\hat{\boldsymbol{W}}_k^\top + \boldsymbol{W}_{\text{up}}\boldsymbol{W}_{\text{down}}^\top)\boldsymbol{Z}^\top)\boldsymbol{Z}\hat{\boldsymbol{W}}_v$. Fig. 5.3d presents the results after two phases of training where dotted curves are drawn from the right hand side of (5.14) directly. Here, note that since $\boldsymbol{\Sigma}, \boldsymbol{\Sigma}^{new}$ are diagonal, the right hand side of (5.14) returns the exact optimal risk of LoRA and the alignments verify it.

CHAPTER 6

Optimization Landscape of Gated Linear Attention

6.1 Introduction

The Transformer [193] has become the de facto standard for language modeling tasks. The key component of the Transformer is the self-attention mechanism, which computes softmax-based similarities between all token pairs. Despite its success, the self-attention mechanism has quadratic complexity with respect to sequence length, making it computationally expensive for long sequences. To address this issue, a growing body of work has proposed near-linear time approaches to sequence modeling. The initial approaches included linear attention and state-space models, both achieving $O(1)$ inference complexity per generated token, thanks to their recurrent form. While these initial architectures typically do not match softmax attention in performance, recent recurrent models such as Mamba [68], mLSTM [16], GLA Transformer [210], and RWKV-6 [155] achieve highly competitive results with the softmax Transformer. Notably, as highlighted in [210], these architectures can be viewed as variants of *gated linear attention* (GLA), which incorporates a gating mechanism within the recurrence of linear attention.

Given a sequence of tokens $(\boldsymbol{z}_i)_{i=1}^{n+1} \in \mathbb{R}^{d+1}$ and the associated query, key, and value embeddings $(\boldsymbol{q}_i, \boldsymbol{k}_i, \boldsymbol{v}_i)_{i=1}^{n+1} \subset \mathbb{R}^{d+1}$, with $d+1$ being the embedding dimension, the GLA recurrence is given by

$$\begin{aligned} \boldsymbol{S}_i &= \boldsymbol{G}_i \odot \boldsymbol{S}_{i-1} + \boldsymbol{v}_i \boldsymbol{k}_i^\top, \quad \text{and} \\ \boldsymbol{o}_i &= \boldsymbol{S}_i \boldsymbol{q}_i, \quad i \in \{1, \ldots, n+1\}. \end{aligned} \tag{GLA}$$

Here, $\boldsymbol{S}_i \in \mathbb{R}^{(d+1)\times(d+1)}$ represents the *2D state variable* with $\boldsymbol{S}_0 = \boldsymbol{0}$; $\boldsymbol{o}_i \in \mathbb{R}^{d+1}$ represents the ith output token [1]; and the *gating variable* $\boldsymbol{G}_i := G(\boldsymbol{z}_i) \in \mathbb{R}^{(d+1)\times(d+1)}$ is applied to the

[1] The sequence length and embedding dimension are set to $n+1$ and $d+1$ per the prompt definition in

state $\boldsymbol{S}_{i-1}$ through the Hadamard product $\odot$ and G represents the gating function. When $\boldsymbol{G}_i$ is a matrix of all ones, (GLA) reduces to causal linear attention [92].

The central objective of this work is to enhance the mathematical understanding of the GLA mechanism. In-context learning (ICL), one of the most remarkable features of modern sequence models, provides a powerful framework to achieve this aim. ICL refers to the ability of a sequence model to implicitly infer functional relationships from the demonstrations provided in its context window [22, 132]. It is inherently related to the model's ability to emulate learning algorithms. Notably, ICL has been a major topic of empirical and theoretical interest in recent years. More specifically, a series of works have examined the approximation and optimization characteristics of linear attention, and have provably connected linear attention to the preconditioned gradient descent algorithm [196, 3, 214]. Given that the GLA recurrence in (GLA) has a richer design space, this leads us to ask:

What learning algorithm does GLA emulate in ICL?
How does its optimization landscape behave in multi-task prompting?

Contributions: The GLA recurrence in (GLA) enables the sequence model to weight past information in a data-dependent manner through the gating mechanism $(\boldsymbol{G}_i)_{i=1}^n$. Building on this observation, we demonstrate that a GLA model can implement a *data-dependent Weighted Preconditioned Gradient Descent (WPGD)* algorithm. Specifically, a one-step version of this algorithm with scalar gating, where all entries of $\boldsymbol{G}_i$ are identical, is described by the prediction:

$$\hat{y} = \boldsymbol{x}^\top \boldsymbol{P} \boldsymbol{X}^\top (\boldsymbol{y} \odot \boldsymbol{\omega}). \tag{6.1}$$

Here, $\boldsymbol{X} \in \mathbb{R}^{n \times d}$ is the input feature matrix; $\boldsymbol{y} \in \mathbb{R}^n$ is the associated label vector; $\boldsymbol{x} \in \mathbb{R}^d$ represents the test/query input to predict; $\boldsymbol{P} \in \mathbb{R}^{d \times d}$ is the preconditioning matrix; and $\boldsymbol{\omega} \in \mathbb{R}^n$ weights the individual samples. When $\boldsymbol{\omega}$ is fixed, we drop "data-dependent" and simply refer to this algorithm as the WPGD algorithm. However, for GLA, the weighting vector $\boldsymbol{\omega}$ depends on the data through recursive multiplication of the gating variables. Building on this formalism, we make the following specific contributions:

- ◇ **ICL capabilities of GLA (§6.3):** Through constructive arguments, we demonstrate that a multilayer GLA model can implement data-dependent WPGD iterations, with weights induced by the gating function. This construction sheds light on the role of causal masking and the expressivity distinctions between scalar- and vector-valued gating functions.

(6.3).

◇ **Landscape of one-step WPGD (§6.4):** The GLA⇔WPGD connection motivates us to ask: *How does WPGD weigh demonstrations in terms of their relevance to the query?* To address this, we study the fundamental problem of learning an optimal WPGD algorithm: Given a tuple $(\boldsymbol{X}, \boldsymbol{y}, \boldsymbol{x}, y) \sim \mathcal{D}$, with $y \in \mathbb{R}$ being the label associated with the query, we investigate the population risk minimization:

$$\begin{aligned} &\mathcal{L}^{\star}_{\mathrm{wpgd}} := \min_{\boldsymbol{P}\in\mathbb{R}^{d\times d}, \boldsymbol{\omega}\in\mathbb{R}^{n}} \ \mathcal{L}_{\mathrm{wpgd}}(\boldsymbol{P}, \boldsymbol{\omega}), \\ &\text{where} \ \ \mathcal{L}_{\mathrm{wpgd}}(\boldsymbol{P}, \boldsymbol{\omega}) = \mathbb{E}_{\mathcal{D}}\left[\left(y - \boldsymbol{x}^{\top}\boldsymbol{P}\boldsymbol{X}^{\top}(\boldsymbol{y}\odot\boldsymbol{\omega})\right)^2\right]. \end{aligned} \tag{6.2}$$

As our primary mathematical contribution, we characterize the loss landscape under a general multitask data setting, where the tasks associated with the demonstrations $(\boldsymbol{X}, \boldsymbol{y})$ have varying degrees of correlation to the target task $(\boldsymbol{x}, y)$. We carefully analyze this loss landscape and show that, under mild conditions, there is a unique global minimum $(\boldsymbol{P}, \boldsymbol{\omega})$ up to scaling invariance, and the associated WPGD algorithm is also unique.

◇ **Loss landscape of one-layer GLA (§6.5):** The landscape is highly intricate due to the recursively multiplied gating variables. We show that learning the optimal GLA layer can be connected to solving (6.2) with a constraint $\boldsymbol{\omega} \in \mathcal{W}$, where the restriction $\mathcal{W}$ is induced by the choice of gating function and input space. Solidifying this connection, we introduce a multitask prompt model under which we characterize the loss landscape of GLA and the influence of task correlations. Our analysis and experiments reveal insightful distinctions between linear attention, GLA with scalar gating, and GLA with vector-valued gating.

6.2 Preliminaries and Problem Setup

Notations. $\mathbb{R}^d$ is the d-dimensional real space, with $\mathbb{R}^d_+$ and $\mathbb{R}^d_{++}$ as its positive and strictly positive orthants. The set $[n]$ denotes $\{1, \ldots, n\}$. Bold letters, e.g., $\boldsymbol{a}$ and $\boldsymbol{A}$, represent vectors and matrices. The identity matrix of size n is denoted by $\boldsymbol{I}_n$. The symbols $\mathbf{1}$ and $\mathbf{0}$ denote the all-one and all-zero vectors or matrices of proper size, such as $\mathbf{1}_d \in \mathbb{R}^d$ and $\mathbf{1}_{d\times d} \in \mathbb{R}^{d\times d}$. The subscript is omitted when the dimension is clear from the context. The Gaussian distribution with mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$ is written as $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$. The Hadamard product (element-wise multiplication) is denoted by $\odot$, and Hadamard division (element-wise division) is denoted by $\oslash$. Given any $\boldsymbol{a}_{i+1}, \ldots, \boldsymbol{a}_j \in \mathbb{R}^d$, we define $\boldsymbol{a}_{i:j} = \boldsymbol{a}_{i+1} \odot \cdots \odot \boldsymbol{a}_j$ for $i < j$, and $\boldsymbol{a}_{i:i} = \mathbf{1}_d$.

The objective of this work is to develop a theoretical understanding of GLA through ICL. The optimization landscape of standard linear attention has been a topic of significant interest in the ICL literature [3, 116]. Following these works, we consider the input prompt

$$\boldsymbol{Z} = [\boldsymbol{z}_1 \quad \dots \quad \boldsymbol{z}_n \quad \boldsymbol{z}_{n+1}]^\top = \begin{bmatrix} \boldsymbol{x}_1 & \dots & \boldsymbol{x}_n & \boldsymbol{x}_{n+1} \\ y_1 & \dots & y_n & 0 \end{bmatrix}^\top \in \mathbb{R}^{(n+1)\times(d+1)}, \tag{6.3}$$

where tokens encode the input-label pairs $(\boldsymbol{x}_i, y_i)_{i=1}^n \subset \mathbb{R}^d \times \mathbb{R}$.

We aim to enable ICL by training a sequence model $F : \mathbb{R}^{(n+1)\times(d+1)} \to \mathbb{R}$ that predicts the label $y := y_{n+1}$ associated with the query $\boldsymbol{x} := \boldsymbol{x}_{n+1}$. This model will utilize the demonstrations $(\boldsymbol{x}_i, y_i)_{i=1}^n$ to infer the mapping between $\boldsymbol{x}$ and y. Assuming that the data is distributed as $(y, \boldsymbol{Z}) \sim \mathcal{D}$, the ICL objective is defined as

$$\mathcal{L}(F) = \mathbb{E}_{\mathcal{D}} \left[(y - F(\boldsymbol{Z}))^2 \right]. \tag{6.4}$$

Linear attention and shared-task distribution. Central to our paper is the choice of the function class F. When F is a linear attention model, the prediction denoted by $\hat{y}_F$ takes the form

$$\hat{y}_F = F(\boldsymbol{Z}) = \boldsymbol{z}_{n+1}^\top \boldsymbol{W}_q \boldsymbol{W}_k^\top \boldsymbol{Z}^\top \boldsymbol{Z} \boldsymbol{W}_v h,$$

where $\boldsymbol{W}_k, \boldsymbol{W}_q, \boldsymbol{W}_v \in \mathbb{R}^{(d+1)\times(d+1)}$ are attention parameters, and $h \in \mathbb{R}^{d+1}$ is the linear prediction head.

To motivate our subsequent discussion on multitask distributions, we begin by reviewing a simplified *shared-task distribution*. Specifically, we let $\boldsymbol{\beta} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\Sigma}_{\boldsymbol{\beta}})$, $\boldsymbol{x}_i$ be i.i.d. with $\boldsymbol{x}_i \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\Sigma}_{\boldsymbol{x}})$, and $y_i \sim \mathcal{N}(\boldsymbol{\beta}^\top \boldsymbol{x}_i, \sigma^2)$, where $\sigma \geq 0$ denotes the noise level. Although this is not our primary modeling assumption, under this shared-task distribution, it has been shown [196, 3, 214] that the optimal one-layer linear attention prediction coincides with one-step preconditioned gradient descent (PGD). In particular, given the data distribution above, we have:

$$\hat{y}_F = \boldsymbol{x}^\top \hat{\boldsymbol{\beta}}, \quad \text{where} \quad \hat{\boldsymbol{\beta}} = \boldsymbol{P}^\star \boldsymbol{X}^\top \boldsymbol{y}, \tag{6.5}$$

and

$$\boldsymbol{P}^\star := \arg\min_{\boldsymbol{P} \in \mathbb{R}^{d\times d}} \mathbb{E}\left[\left(y - \boldsymbol{x}^\top \boldsymbol{P} \boldsymbol{X}^\top \boldsymbol{y} \right)^2 \right] \quad \text{with} \quad \boldsymbol{X} := \begin{bmatrix} \boldsymbol{x}_1 & \cdots & \boldsymbol{x}_n \end{bmatrix}^\top \quad \text{and} \quad \boldsymbol{y} := \begin{bmatrix} y_1 & \cdots & y_n \end{bmatrix}^\top. \tag{6.6}$$

Gated linear attention and multitask prompting. Given the input prompt $\boldsymbol{Z}$ as in (6.3), let $(\boldsymbol{q}_i, \boldsymbol{k}_i, \boldsymbol{v}_i) = (\boldsymbol{W}_q^\top \boldsymbol{z}_i, \boldsymbol{W}_k^\top \boldsymbol{z}_i, \boldsymbol{W}_v^\top \boldsymbol{z}_i)$ be the corresponding query, key, and value embeddings. The output of *causal* linear attention at time i can be computed using (GLA) with $\boldsymbol{G}_i = \boldsymbol{1}$ for all $i \in [n+1]$. This recurrent form implies that linear attention has $O(d^2)$ cost, that is independent of sequence length n, to generate per-token. GLA follows the same structure as linear attention but with a gating mechanism (i.e., $\boldsymbol{G}_i \neq \boldsymbol{1}$ for some $i \in [n+1]$), which equips the model with the option to pass or supress the history. As discussed in [210], the different choices of the gating correspond to different popular recurrent architectures such as Mamba [68], Mamba2 [40], RWKV [155], etc.

We will show that GLA can weigh the context window through gating, thus, its capabilities are linked to the WPGD algorithm described in (6.8). This will in turn facilitate GLA to effectively learn *multitask prompt distributions* described by $y_i \sim \mathcal{N}(\boldsymbol{\beta}_i^\top \boldsymbol{x}_i, \sigma^2)$ with $\boldsymbol{\beta}_i$'s not necessarily identical.

6.3 What Gradient Methods can GLA Emulate?

In this section, we investigate the ICL capabilities of GLA and show that under suitable instantiations of model weights, GLA can implement *data-dependent* WPGD.

6.3.1 GLA as a data-dependent WPGD predictor

Data-Dependent WPGD. Given $\boldsymbol{X}$ and $\boldsymbol{y}$ as defined in (6.6), consider the weighted least squares objective $\mathcal{L}(\boldsymbol{\beta}) = \sum_{i=1}^n \omega_i \cdot (y_i - \boldsymbol{\beta}^\top \boldsymbol{x}_i)^2$ with weights $\boldsymbol{\omega} = [\omega_1, \omega_2, \ \cdots, \ \omega_n]^\top \in \mathbb{R}^n$. To optimize this, we use gradient descent (GD) starting from zero initialization, $\boldsymbol{\beta}_0 = \boldsymbol{0}$ with a step size of $\eta = 1/2$. One step of standard GD is given by

$$\boldsymbol{\beta}_1 = \boldsymbol{\beta}_0 - \eta \nabla \mathcal{L}(\boldsymbol{\beta}_0) = \sum_{i=1}^n \omega_i \cdot \boldsymbol{x}_i y_i = \boldsymbol{X}^\top (\boldsymbol{\omega} \odot \boldsymbol{y}).$$

Given a test/query feature $\boldsymbol{x}$, the corresponding prediction is $\hat{y} = \boldsymbol{x}^\top \hat{\boldsymbol{\beta}}$ where $\hat{\boldsymbol{\beta}} = \boldsymbol{\beta}_1$. Additionally, if we were using PGD with a preconditioning matrix $\boldsymbol{P} \in \mathbb{R}^{d \times d}$, $\boldsymbol{\beta}_0 = \boldsymbol{0}$, and $\eta = 1/2$, then a single iteration results in

$$\hat{y} = \boldsymbol{x}^\top \hat{\boldsymbol{\beta}}, \quad \text{where} \quad \hat{\boldsymbol{\beta}} = \boldsymbol{\beta}_0 - \eta \boldsymbol{P} \nabla \mathcal{L}(\boldsymbol{\beta}_0) = \boldsymbol{P} \boldsymbol{X}^\top (\boldsymbol{\omega} \odot \boldsymbol{y}). \tag{6.7}$$

Above is the basic *sample-wise* WPGD predictor which weights individual datapoints, in contrast to the PGD predictor as presented in (6.5). It turns out, *vector-valued gating* can

facilitate a more general estimator which weights individual coordinates. To this aim, we introduce an extension as follows: Let $\boldsymbol{P}_1, \boldsymbol{P}_2 \in \mathbb{R}^{d\times d}$ denote the preconditioning matrices, and let $\boldsymbol{\Omega} \in \mathbb{R}^{n\times d}$ denote the *vector-valued weighting* matrix. Note that $\boldsymbol{\Omega}$ is a weight matrix that enables coordinate-wise weighting. Throughout the paper, we use $\boldsymbol{\omega}$ and $\boldsymbol{\Omega}$ to denote sample-wise and coordinate-wise weighting parameters, corresponding to scalar and vector gating strategies, respectively. Then given $\boldsymbol{\Omega}$, we can similarly define

$$\boldsymbol{\beta}_1^{\mathrm{gd}}(\boldsymbol{P}_1, \boldsymbol{P}_2, \boldsymbol{\Omega}) := \boldsymbol{P}_2(\boldsymbol{X}\boldsymbol{P}_1 \odot \boldsymbol{\Omega})^\top \boldsymbol{y} \tag{6.8a}$$

as one-step of (generalized) WPGD. Its corresponding prediction on a test query $\boldsymbol{x}$ is:

$$\hat{y} = \boldsymbol{x}^\top \hat{\boldsymbol{\beta}}, \quad \text{where} \quad \hat{\boldsymbol{\beta}} = \boldsymbol{\beta}_1^{\mathrm{gd}}(\boldsymbol{P}_1, \boldsymbol{P}_2, \boldsymbol{\Omega}). \tag{6.8b}$$

We note that by removing the weighting matrix $\boldsymbol{\Omega}$, (6.8) reduces to standard PGD (cf. (6.5)) and setting $\boldsymbol{\Omega} = \boldsymbol{\omega}\mathbf{1}_d^\top$ reduces to sample-wise WPGD (cf. (6.7)). [116] has demonstrated that H3-like models implement one-step sample-wise WPGD, and they focus on the shared-task distribution where $\boldsymbol{\beta}_i \equiv \boldsymbol{\beta}$ for all $i \in [n]$. In contrast, our work considers a more general data setting where tasks within an in-context prompt are not necessarily identical.

We introduce the following model constructions under which we establish the equivalence between GLA (cf. (GLA)) and WPGD (cf. (6.8)), where the weighting matrix is induced by the input data and the gating function. Inspired by prior works [196, 3], we consider the following restricted attention matrices:

$$\boldsymbol{W}_k = \begin{bmatrix} \boldsymbol{P}_k & \mathbf{0} \\ \mathbf{0} & 0 \end{bmatrix}, \quad \boldsymbol{W}_q = \begin{bmatrix} \boldsymbol{P}_q & \mathbf{0} \\ \mathbf{0} & 0 \end{bmatrix}, \quad \text{and} \quad \boldsymbol{W}_v = \begin{bmatrix} \mathbf{0}_{d\times d} & \mathbf{0} \\ \mathbf{0} & 1 \end{bmatrix}, \tag{6.9}$$

where $\boldsymbol{P}_k, \boldsymbol{P}_q \in \mathbb{R}^{d\times d}$. Note that the $(d+1, d+1)$-th entry of $\boldsymbol{W}_v$ is set to one for simplicity. More generally, this entry can take any nonzero value, e.g., $v \in \mathbb{R}$. Parameterizing $\boldsymbol{W}_q$ with $\boldsymbol{P}_q/v$ produces the same output as in (6.9).

Theorem 6.1 *Recall* (GLA) *with input sequence* $\boldsymbol{Z} = [\boldsymbol{z}_1 \ \ldots \ \boldsymbol{z}_n \ \boldsymbol{z}_{n+1}]^\top$ *defined in* (6.3)*, the gating* $G(\boldsymbol{z}_i) = \boldsymbol{G}_i \in \mathbb{R}^{(d+1)\times(d+1)}$*, and outputs* $(\boldsymbol{o}_i)_{i=1}^{n+1}$*. Consider model construction in* (6.9) *and take the last coordinate of the last token output denoted by* $(\boldsymbol{o}_{n+1})_{d+1}$ *as a prediction. Then, we have*

$$f_{gla}(\boldsymbol{Z}) := (\boldsymbol{o}_{n+1})_{d+1} = \boldsymbol{x}^\top \hat{\boldsymbol{\beta}}, \quad \text{where} \quad \hat{\boldsymbol{\beta}} = \boldsymbol{\beta}_1^{gd}(\boldsymbol{P}_k, \boldsymbol{P}_q, \boldsymbol{\Omega}).$$

Here, $\boldsymbol{\beta}_1^{gd}(\cdot)$ *is a one-step WPGD feature predictor defined in* (6.8a)*;* $\boldsymbol{P}_k$ *and* $\boldsymbol{P}_q$ *correspond*

to attention weights following (6.9)*; and* $\boldsymbol{\Omega} = [\boldsymbol{g}_{1:n+1}\ \boldsymbol{g}_{2:n+1}\ \cdots\ \boldsymbol{g}_{n:n+1}]^\top \in \mathbb{R}^{n\times d}$*, where* $\boldsymbol{g}_{i:n+1}, i \in [n]$ *is given by*

$$\boldsymbol{g}_{i:n+1} := \boldsymbol{g}_{i+1} \odot \boldsymbol{g}_{i+2} \cdots \boldsymbol{g}_{n+1} \in \mathbb{R}^d, \quad \text{and} \quad \boldsymbol{G}_i = \begin{bmatrix} * & * \\ \boldsymbol{g}_i^\top & * \end{bmatrix}. \tag{6.10}$$

Here, and throughout, we use $*$ to fill the entries of the matrices that do not affect the final output, and based on the model construction given in (6.9), these entries can be assigned any value.

Observe that, crucially, since $\boldsymbol{g}_i$ is associated with $\boldsymbol{z}_i$, $\boldsymbol{z}_i$ influences the weighting of $\boldsymbol{z}_j$ for all $j < i$. We defer the proof of Theorem 6.1 to the Appendix E.2.1. It is noticeable that only d of the total $(d+1)^2$ entries in each gating matrix $\boldsymbol{G}_i$ are useful due to the model construction presented in (6.9). However, if we relax the weight restriction, e.g., $\boldsymbol{W}_v = [\boldsymbol{0}_{(d+1)\times d}\ \ \boldsymbol{u}]^\top$, where $\boldsymbol{u} \in \mathbb{R}^{d+1}$, then the weighting matrix $\boldsymbol{\Omega}$ in Theorem 6.1 is associated with all rows of the $\boldsymbol{G}_i$ matrices. We defer to Eqn. (E.2) and the discussion in Appendix E.2.1.

6.3.2 Capabilities of multi-layer GLA

[3] demonstrated that, with appropriate construction, an L-layer linear attention model performs L-step preconditioned GD on the dataset $(\boldsymbol{x}_i, y_i)_{i=1}^n$ provided within the prompt. In this work, we study multi-layer GLA and analyze the associated algorithm class it can emulate. It is worth mentioning that [3] does not consider *causal masking* which is integral to multilayer GLA due to its recurrent nature described in (GLA). Our analysis will capture the impact of gating and causal mask through n separate GD trajectories that are coupled.

Multi-layer GLA. For any input prompt $\boldsymbol{Z} \in \mathbb{R}^{(n+1)\times(d+1)}$, let $\mathrm{gla}(\boldsymbol{Z}) := [\boldsymbol{o}_1\ \ \boldsymbol{o}_2\ \ \cdots\ \ \boldsymbol{o}_{n+1}]^\top \in \mathbb{R}^{(n+1)\times(d+1)}$ denote its GLA output, as defined in (GLA). We consider an L-layer GLA model as follows: For each layer $\ell \in [L]$, let $\boldsymbol{Z}_\ell$ denote its input, where we set $\boldsymbol{Z}_1 = \boldsymbol{Z}$, and let $\boldsymbol{W}_{k,\ell}, \boldsymbol{W}_{q,\ell}, \boldsymbol{W}_{v,\ell}$ denote the key, query, and value matrices for layer ℓ, respectively. After applying a residual connection, the input of the ℓ-th layer is given by

$$\boldsymbol{Z}_\ell := \boldsymbol{Z}_{\ell-1} + \mathrm{gla}_{\ell-1}(\boldsymbol{Z}_{\ell-1}). \tag{6.11}$$

Here, $\mathrm{gla}_\ell(\cdot)$ denotes the output of the ℓ-th GLA layer, associated with the attention matrices $(\boldsymbol{W}_{q,\ell}, \boldsymbol{W}_{k,\ell}, \boldsymbol{W}_{v,\ell})$ for all $\ell \in [L]$. In the following, we focus on the $(d+1)$-th entry of each token's output at each layer (after the residual connection), denoted by $o_{i,\ell} := [\boldsymbol{Z}_\ell + \mathrm{gla}_\ell(\boldsymbol{Z}_\ell)]_{i,d+1}$ for all $i \in [n+1]$ and $\ell \in [L]$.

Theorem 6.2 *Consider an L-layer GLA defined in (6.11), where $\boldsymbol{W}_{k,\ell}$, $\boldsymbol{W}_{q,\ell}$ in the ℓ'th layer parameterized by $\boldsymbol{P}_{k,\ell}, -\boldsymbol{P}_{q,\ell} \in \mathbb{R}^{d\times d}$, for all $\ell \in [L]$ following (6.9). Let the gating be a function of the features, e.g., $\boldsymbol{G}_i = G(\boldsymbol{x}_i)$, and define $\boldsymbol{\Omega}$ as in Theorem 6.1. Additionally, denote the masking as $\boldsymbol{M}_i = \begin{bmatrix} \boldsymbol{I}_i & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix} \in \mathbb{R}^{n\times n}$, and let $\hat{\boldsymbol{\beta}}_0 = \bigotimes i0 = \boldsymbol{0}$ for all $i \in [n]$. Recall that $o_{i,\ell}$ for all $i \in [n+1]$ and $\ell \in [L]$ stands for the last entry of the i'th token output at the ℓ'th layer. We have*

- *For all $i \le n$, $o_{i,\ell} = y_i - \boldsymbol{x}_i^\top \boldsymbol{\beta}_{i,\ell}$, where $\boldsymbol{\beta}_{i,\ell} = \boldsymbol{\beta}_{i,\ell-1} - \boldsymbol{P}_{q,\ell}\left(\nabla_{i,\ell} \oslash \boldsymbol{g}_{i:n+1}\right)$;*
- *$o_{n+1,\ell} = -\boldsymbol{x}^\top \hat{\boldsymbol{\beta}}_\ell$, where $\hat{\boldsymbol{\beta}}_\ell = (1-\alpha_\ell)\hat{\boldsymbol{\beta}}_{\ell-1} - \boldsymbol{P}_{q,\ell}\left(\nabla_{n,\ell} \oslash \boldsymbol{g}_{n+1}\right)$, and $\alpha_\ell = \boldsymbol{x}^\top \boldsymbol{P}_{q,\ell}\boldsymbol{P}_{k,\ell}^\top \boldsymbol{x}$.*

Here, letting $\boldsymbol{B}_\ell = [\bigotimes 1\ell \ \cdots \ \bigotimes n\ell]^\top$, $\boldsymbol{X}_\ell = \boldsymbol{X}\boldsymbol{P}_{k,\ell} \odot \boldsymbol{\Omega}$, and $\boldsymbol{y}_\ell = \mathrm{diag}(\boldsymbol{X}\boldsymbol{B}_\ell^\top)$, we define

$$\nabla_{i,\ell} = \boldsymbol{X}_\ell^\top \boldsymbol{M}_i \left(\boldsymbol{y}_\ell - \boldsymbol{y}\right).$$

We defer the proof of Theorem 6.2 to the Appendix E.2.2. Theorem 6.2 states that L-layer GLA (6.11) implements L steps of WPGD with gradient in a recurrent form and additional weight decay α_ℓ. To recap, given data $(\boldsymbol{X}, \boldsymbol{y})$ and prediction $\hat{\boldsymbol{\beta}}$, the gradient with respect to the squared loss takes the form $\boldsymbol{X}^\top(\boldsymbol{X}\hat{\boldsymbol{\beta}} - \boldsymbol{y})$, up to some constant c. In comparison, $\boldsymbol{P}_{q,\ell}\left(\nabla_{i,\ell} \oslash \boldsymbol{g}_{i:n+1}\right)$ similarly acts as a gradient but incorporates layer-wise feature preconditioners $(\boldsymbol{P}_{q,\ell}, \boldsymbol{P}_{k,\ell})$, data weighting $(\boldsymbol{\Omega})$, and causality $(\boldsymbol{g}_{i:n+1}, \boldsymbol{M}_i)$. Here, $\boldsymbol{M}_i$ represents causal masking, ensuring that at time i, only inputs from $j \le i$ are used for prediction. Notably, the recurrent structure of GLA allows the gating mechanism to apply context-dependent weighting strategies.

To simplify the theorem statement, we assume that the gating function depends only on the input feature, e.g., $\boldsymbol{G}_i = G(\boldsymbol{x}_i)$, ensuring that the corresponding data-dependent weighting is uniform across all layers. This assumption is included solely for clarity in the theorem statement, and if the gating function varies across layers or depends on the entire $(d+1)$-dimensional token rather than just the feature $\boldsymbol{x}_i$, each layer has its own gating $\boldsymbol{\Omega}_\ell$ and $\boldsymbol{X}_\ell = \boldsymbol{X}\boldsymbol{P}_{k,\ell} \odot \boldsymbol{\Omega}_\ell$. Note that our inclusion of the additional term α_ℓ captures the influence of the last token's output on the next layer's prediction. Based on the above L-layer GLA result, we have the following corollary for multi-layer linear attention network with causal mask in each layer.

Corollary 6.1 *Consider an L-layer linear attention model with causal mask and residual connection in each layer. Let ℓ'th layer follows the weight construction in (6.9) with $\boldsymbol{W}_{k,\ell}, \boldsymbol{W}_{q,\ell}$ parameterized by $\boldsymbol{P}_{k,\ell}, -\boldsymbol{P}_{q,\ell}$ as in Theorem 6.2 and define $\boldsymbol{P}_\ell := \boldsymbol{P}_{q,\ell}\boldsymbol{P}_{k,\ell}^\top$, for*

$\ell \in [L]$. *Let* $\hat{\boldsymbol{\beta}}_0 = \mathbf{0}$. *Then, the* $(d+1)$*'th entry of the last token output of each layer satisfies:*

$$o_{n+1,\ell} = -\boldsymbol{x}^\top \hat{\boldsymbol{\beta}}_\ell, \quad \textit{where} \quad \hat{\boldsymbol{\beta}}_\ell = (1-\alpha_\ell)\hat{\boldsymbol{\beta}}_{\ell-1} - \boldsymbol{P}_\ell \boldsymbol{X}^\top (\boldsymbol{y}_\ell - \boldsymbol{y}).$$

Here, we define $\alpha_\ell = \boldsymbol{x}^\top \boldsymbol{P}_\ell \boldsymbol{x}$ *and* $\boldsymbol{y}_\ell$ *follows the same definition as in Theorem 6.2.*

Note that Corollary 6.1 generalizes [3, Lemma 1] by extending it to causal attention. When considering full attention (i.e., without applying the causal mask), we have $\boldsymbol{y}_\ell = \boldsymbol{X}\hat{\boldsymbol{\beta}}_\ell$, and the expression $\boldsymbol{X}^\top(\boldsymbol{y}_\ell - \boldsymbol{y}) = \boldsymbol{X}^\top(\boldsymbol{X}\hat{\boldsymbol{\beta}}_\ell - \boldsymbol{y})$ corresponds to the standard gradient of the squared loss. Furthermore, if we disregard the impact of the last token (e.g., by incorporating the matrix M as in Eq. (3) of [3]), then $\alpha_\ell = 0$. These results are also consistent with [46], which demonstrate that causal masking limits convergence by introducing sequence biases, akin to online gradient descent with non-decaying step sizes.

Our theoretical results in Theorem 6.2 focus on multi-layer GLA without Multi-Layer Perceptron (MLP) layers to isolate and analyze the effects of the gating mechanism. However, MLP layers, a key component of standard Transformers, facilitate further nonlinear feature transformations and interactions, potentially enhancing GLA's expressive power. Future work could explore the theoretical foundations of integrating MLPs into GLA and analyze the optimization landscape of general gated attention models, aligning them more closely with conventional Transformer architectures [68, 40, 155].

6.3.3 GLA with scalar gating

Theorem 6.1 establishes a connection between one-layer GLA (cf. (GLA)) and one-step WPGD (cf. (6.8)), where the weighting in WPGD corresponds to the gating $G(\boldsymbol{z}_i) = \boldsymbol{G}_i$ in GLA, as detailed in Theorem 6.1. Now let us consider the widely used types of gating functions, such as $\boldsymbol{G}_i = \boldsymbol{\alpha}_i \mathbf{1}_{d+1}^\top$ [210, 93, 162, 155] or $\boldsymbol{G}_i = \gamma_i \mathbf{1}_{(d+1),(d+1)}$ [40, 16, 156, 182] where $\boldsymbol{\alpha}_i \in \mathbb{R}^{d+1}$ and $\gamma_i \in \mathbb{R}$. In both cases, the gating matrices in (6.10) take the form of $\begin{bmatrix} * & * \\ g_i \mathbf{1}_d^\top & * \end{bmatrix}$ for some $g_i \in \mathbb{R}$, thus simplifying the predictor to a sample-wise WPGD, as given by

$$f_{\mathrm{gla}}(\boldsymbol{Z}) = \hat{\boldsymbol{\beta}}^\top \boldsymbol{x}, \quad \text{with} \quad \hat{\boldsymbol{\beta}} = \boldsymbol{P}\boldsymbol{X}^\top(\boldsymbol{\omega} \odot \boldsymbol{y}), \tag{6.12}$$

where $\boldsymbol{P} = \boldsymbol{P}_q \boldsymbol{P}_k^\top$ and $\boldsymbol{\omega} = [g_{1:n+1} \ \cdots \ g_{n:n+1}]^\top \in \mathbb{R}^n$. In the remainder, we will mostly focus on the one-layer GLA with scalar gating as presented in (6.12).

6.4 Optimization Landscape of WPGD

In this section, we explore the problem of learning the optimal sample-wise WPGD algorithm described in (6.12), a key step leading to our analysis of GLA. The problem is as follows. Recap from (6.6) that we are given the tuple $(\boldsymbol{x}, y, \boldsymbol{X}, \boldsymbol{y}) \sim \mathcal{D}$, where $\boldsymbol{X} \in \mathbb{R}^{n\times d}$ is the input matrix, $\boldsymbol{y} \in \mathbb{R}^n$ is the label vector, $\boldsymbol{x} \in \mathbb{R}^d$ is the query, and $y \in \mathbb{R}$ is its associated label. The goal is to use $\boldsymbol{X}, \boldsymbol{y}$ to predict y given $\boldsymbol{x}$ via the one-step WPGD prediction $\hat{y} = \boldsymbol{x}^\top \hat{\boldsymbol{\beta}}$, with $\hat{\boldsymbol{\beta}}$ as in (6.12). The algorithm learning problem is given by (6.2) which minimizes the WPGD risk $\mathbb{E}_{\mathcal{D}}[\big(y - \boldsymbol{x}^\top \boldsymbol{P}\boldsymbol{X}(\boldsymbol{\omega} \odot \boldsymbol{y})\big)^2]$.

Prior research [129, 116, 3] has studied the problem of learning PGD when input-label pairs follow an i.i.d. distribution. It is worth noting that while [116] establishes a connection between H3-like models and (6.12) similar to ours, their work assumes that the optimal $\boldsymbol{\omega}$ consists of all ones and does not specifically explore the optimization landscape of $\boldsymbol{\omega}$ when in-context samples are not generated from the same task vector $\boldsymbol{\beta}$. Departing from this, we introduce a realistic model where each input-label pair is allowed to come from a distinct task.

Definition 6.1 (Correlated Task Model) *Suppose $\boldsymbol{\beta}_i \in \mathbb{R}^d \sim \mathcal{N}(0, \boldsymbol{I})$ are jointly Gaussian for all $i \in [n+1]$. Define*

$$\boldsymbol{\beta} := \boldsymbol{\beta}_{n+1}, \quad \boldsymbol{B} := [\boldsymbol{\beta}_1 \ \dots \ \boldsymbol{\beta}_n]^\top, \quad \boldsymbol{R} := \frac{1}{d}\mathbb{E}[\boldsymbol{B}\boldsymbol{B}^\top], \quad \text{and} \quad \boldsymbol{r} := \frac{1}{d}\mathbb{E}[\boldsymbol{B}\boldsymbol{\beta}]. \tag{6.13}$$

Note that in (6.13), we have $\boldsymbol{R} \in \mathbb{R}^{n\times n}$ and $\boldsymbol{r} \in \mathbb{R}^n$, with normalization ensuring that the entries of $\boldsymbol{R}$ and $\boldsymbol{r}$ lie in the range $[-1, 1]$, corresponding to correlation coefficients. Further, due to the joint Gaussian nature of the tasks, for any $i, j \in [n+1]$, the residual $\boldsymbol{\beta}_i - r_{ij}\boldsymbol{\beta}_j$ is independent of $\boldsymbol{\beta}_j$.

Definition 6.2 (Multitask Distribution) *Let $(\boldsymbol{\beta}_i)_{i=1}^{n+1}$ be drawn according to the correlated task model of Definition 6.1, $(\boldsymbol{x}_i)_{i=1}^{n+1} \in \mathbb{R}^d$ be i.i.d. with $\boldsymbol{x}_i \sim \mathcal{N}(0, \boldsymbol{\Sigma})$, and $y_i \sim \mathcal{N}(\boldsymbol{x}_i^\top \boldsymbol{\beta}_i, \sigma^2)$ for all $i \in [n+1]$.*

Definition 6.3 *Let the eigen decompositions of $\boldsymbol{\Sigma}$ and $\boldsymbol{R}$ be denoted by $\boldsymbol{\Sigma} = \boldsymbol{U}\mathrm{diag}(\boldsymbol{s})\boldsymbol{U}^\top$ and $\boldsymbol{R} = \boldsymbol{E}\mathrm{diag}(\boldsymbol{\lambda})\boldsymbol{E}^\top$, where $\boldsymbol{s} = [s_1 \ \cdots \ s_d]^\top \in \mathbb{R}^d_{++}$ and $\boldsymbol{\lambda} = [\lambda_1 \ \cdots \ \lambda_n]^\top \in \mathbb{R}^n_+$. Let $s_{\min}$ and $s_{\max}$ denote the smallest and largest eigenvalues of $\boldsymbol{\Sigma}$, respectively. Further, let $\lambda_{\min}$ and $\lambda_{\max}$ denote the nonzero smallest and largest eigenvalues of $\boldsymbol{R}$. Define the effective spectral gap of $\boldsymbol{\Sigma}$ and $\boldsymbol{R}$, respectively, as*

$$\Delta_{\boldsymbol{\Sigma}} := s_{\max} - s_{\min}, \quad \text{and} \quad \Delta_{\boldsymbol{R}} := \lambda_{\max} - \lambda_{\min}. \tag{6.14}$$

Assumption 6.1 *The correlation vector* $\boldsymbol{r}$ *from* (6.13) *lies in the range of* $\boldsymbol{E}$*, i.e.,* $\boldsymbol{r} = \boldsymbol{E}\boldsymbol{a}$ *for some nonzero* $\boldsymbol{a} = [a_1 \ \cdots \ a_n]^\top \in \mathbb{R}^n$*.*

Assumption 6.1 essentially ensures that $\boldsymbol{r}$ (representing the correlations between in-context tasks) can be expressed as a linear transformation of a vector $\boldsymbol{a}$ with at least one nonzero value. Note that since $\boldsymbol{r}$ lies in the range of the eigenvector matrix $\boldsymbol{E}$, it also lies in the range of $\boldsymbol{R}$ defined in (6.13). This guarantees that the correlation structure is non-degenerate, meaning that all elements of $\boldsymbol{r}$ are influenced by meaningful correlations. Assumption 6.1 avoids trivial cases where there are no correlations between tasks. By requiring at least one nonzero element in $\boldsymbol{a}$, the assumption ensures that the tasks are interrelated.

The following theorem characterizes the stationary points $(\boldsymbol{P}, \boldsymbol{\omega})$ of the WPGD objective in (6.2).

Theorem 6.3 *Consider independent linear data as described in Definition 6.2. Suppose Assumption 6.1 on the correlation vector* $\boldsymbol{r}$ *holds. Let the functions* $h_1 : \mathbb{R}_+ \to \mathbb{R}_+$ *and* $h_2 : [1, \infty) \to \mathbb{R}_+$ *be defined as*

$$h_1(\bar{\gamma}) := \left(\sum_{i=1}^{n} \frac{\lambda_i a_i^2}{(1 + \lambda_i \bar{\gamma})^2} \right) \left(\sum_{i=1}^{n} \frac{a_i^2}{(1 + \lambda_i \bar{\gamma})^2} \right)^{-1}, \tag{6.15a}$$

$$h_2(\gamma) := \left(1 + M \left(\sum_{i=1}^{d} \frac{s_i^2}{(M + s_i \gamma)^2} \right) \left(\sum_{i=1}^{d} \frac{s_i^3}{(M + s_i \gamma)^2} \right)^{-1} \right)^{-1}, \tag{6.15b}$$

where $\{s_i\}_{i=1}^d$ *and* $\{\lambda_i\}_{i=1}^n$ *are the eigenvalues of* $\boldsymbol{\Sigma}$ *and* $\boldsymbol{R}$*, respectively;* $\{a_i\}_{i=1}^n$ *are given in Assumption 6.1; and* $M = \sigma^2 + \sum_{i=1}^d s_i$*.*

The risk function $\mathcal{L}_{wpgd}(\boldsymbol{P}, \boldsymbol{\omega})$ *in* (6.2) *has a stationary point* $(\boldsymbol{P}^\star, \boldsymbol{\omega}^\star)$*, up to rescaling, defined as*

$$\boldsymbol{P}^\star = \boldsymbol{\Sigma}^{-\frac{1}{2}} \left(\frac{\gamma^\star}{M} \cdot \boldsymbol{\Sigma} + \boldsymbol{I} \right)^{-1} \boldsymbol{\Sigma}^{-\frac{1}{2}}, \quad \text{and} \quad \boldsymbol{\omega}^\star = \left(h_2(\gamma^\star) \cdot \boldsymbol{R} + \boldsymbol{I} \right)^{-1} \boldsymbol{r}, \tag{6.16}$$

where $\gamma^\star$ *is a fixed point of composite function* $h_1(h_2(\gamma)) + 1$*.*

Theorem 6.3 characterizes the stationary points $(\boldsymbol{P}^\star, \boldsymbol{\omega}^\star)$, which exist up to re-scaling. This result presents the first landscape analysis of GLA for the joint learning of $(\boldsymbol{P}, \boldsymbol{\omega})$, while also exploring the stationary points $(\boldsymbol{P}^\star, \boldsymbol{\omega}^\star)$. In the following, we provide mild conditions on effective spectral gaps of $\boldsymbol{R}$ and $\boldsymbol{\Sigma}$ under which a unique (global) minimum $(\boldsymbol{P}^\star, \boldsymbol{\omega}^\star)$ exists.

Theorem 6.4 (Uniqueness of the WPGD Predictor) *Consider independent linear data as given in Definition 6.2. Suppose Assumption 6.1 on the correlation vector* $\boldsymbol{r}$ *holds,*

and

$$\Delta_{\boldsymbol{\Sigma}} \cdot \Delta_{\boldsymbol{R}} < M + s_{\min}, \tag{6.17}$$

where $\Delta_{\boldsymbol{\Sigma}}$ *and* $\Delta_{\boldsymbol{R}}$ *denote the effective spectral gaps of* $\boldsymbol{\Sigma}$ *and* $\boldsymbol{R}$*, respectively, as given in* (6.14)*;* $s_{\min}$ *is the smallest eigenvalue of* $\boldsymbol{\Sigma}$*; and* $M = \sigma^2 + \sum_{i=1}^{d} s_i$*.*

T1. *The composite function* $h_1(h_2(\gamma)) + 1$ *is a contraction mapping and admits a unique fixed point* $\gamma = \gamma^\star$*.*

T2. *The function* $\mathcal{L}_{wpgd}(\boldsymbol{P}, \boldsymbol{\omega})$ *has a unique (global) minima* $(\boldsymbol{P}^\star, \boldsymbol{\omega}^\star)$*, up to re-scaling, given by* (6.16)*.*

Proof sketch. Let $\gamma := \frac{\boldsymbol{\omega}^\top \boldsymbol{R} \boldsymbol{\omega}}{\|\boldsymbol{\omega}\|^2} + 1$. Note that $\gamma \geq 1$ since $\boldsymbol{R}$ is positive semi-definite. From the first-order optimality condition, the solution to (6.2) takes the following form:

$$\begin{cases} \boldsymbol{P}(\gamma) = C(\boldsymbol{r}, \boldsymbol{\omega}, \boldsymbol{\Sigma}) \cdot \boldsymbol{\Sigma}^{-\frac{1}{2}} \left(\frac{\gamma}{M} \cdot \boldsymbol{\Sigma} + \boldsymbol{I} \right)^{-1} \boldsymbol{\Sigma}^{-\frac{1}{2}}, \\ \boldsymbol{\omega}(\gamma) = c(\boldsymbol{r}, \boldsymbol{\omega}, \boldsymbol{\Sigma}) \cdot \left(h_2(\gamma) \cdot \boldsymbol{R} + \boldsymbol{I} \right)^{-1} \boldsymbol{r}, \end{cases} \tag{6.18}$$

for some constants $C(\boldsymbol{r}, \boldsymbol{\omega}, \boldsymbol{\Sigma})$ and $c(\boldsymbol{r}, \boldsymbol{\omega}, \boldsymbol{\Sigma})$.

Substituting the expression for $\boldsymbol{\omega}(\gamma)$ into $\gamma = \frac{\boldsymbol{\omega}^\top \boldsymbol{R} \boldsymbol{\omega}}{\|\boldsymbol{\omega}\|^2} + 1$, and applying Assumption 6.1, we obtain the equation $\gamma = h_1(h_2(\gamma)) + 1$. We then show that whenever $\Delta_{\boldsymbol{\Sigma}} \cdot \Delta_{\boldsymbol{R}} < M + s_{\min}$, the mapping $h_1(h_2(\gamma)) + 1$ is a contraction (see Lemma E.2). By the Banach Fixed-Point Theorem, this guarantees the existence of a unique fixed point $\gamma = \gamma^\star$, where $\gamma^\star = h_1(h_2(\gamma^\star)) + 1$. Finally, substituting $\gamma^\star$ into (6.18) implies that $(\boldsymbol{P}^\star, \boldsymbol{\omega}^\star)$, as given in (6.16), is a unique (global) minima of (6.2), up to re-scaling. See Appendix E.3.2 for the complete proof of Theorem 6.4.

Theorem 6.4 establishes mild conditions under which a unique (global) minimum $(\boldsymbol{P}^\star, \boldsymbol{\omega}^\star)$ exists, up to scaling invariance, and guarantees the uniqueness of the associated WPGD algorithm. It provides the first global landscape analysis for GLA and generalizes prior work [116, 3] on the global landscape by extending the optimization properties of linear attention to the more complex *nonconvex* GLA with joint $(\boldsymbol{P}, \boldsymbol{\omega})$ optimization.

Remark 1. An interesting observation about the optimal gating parameter $\boldsymbol{\omega}^\star$ is its connection to the correlation matrix $\boldsymbol{R}$, which captures the task correlations in a multitask learning setting. Specifically, the optimal gating given in (6.16) highlights how $\boldsymbol{\omega}^\star$ depends directly on both the task correlation matrix $\boldsymbol{R}$ and the vector $\boldsymbol{r}$, which encodes the correlations between the tasks and the target task.

Remark 2. Condition (6.17) provides a *sufficient* condition for the uniqueness of a fixed point. This implies that whenever $\Delta_{\boldsymbol{\Sigma}} \cdot \Delta_{\boldsymbol{R}} < M + s_{\min}$, the mapping $h_1(h_2(\gamma)) + 1$ is a contraction, ensuring the existence of a unique fixed point. However, there may be cases where the mapping $h_1(h_2(\gamma)) + 1$ does not satisfy Condition (6.17), yet a unique fixed point (and a unique global minimum) still exists. This is because the Banach Fixed-Point Theorem does not provide a *necessary* condition.

Corollary 6.2 *Suppose* $\boldsymbol{\Sigma} = \boldsymbol{I}$. *Then,* $\Delta_{\boldsymbol{\Sigma}} = 0$, *satisfying Condition* (6.17), *and we have* $h_2(\gamma^\star) = \frac{1}{d+\sigma^2+1}$, *which yields*

$$\boldsymbol{P}^\star = \boldsymbol{I}, \quad \text{and} \quad \boldsymbol{\omega}^\star = \big(\boldsymbol{R} + (d + \sigma^2 + 1)\boldsymbol{I}\big)^{-1} \boldsymbol{r}. \tag{6.19}$$

Thus, the optimal risk $\mathcal{L}^\star_{wpgd}$ *defined in* (6.2) *is given by*

$$\mathcal{L}^\star_{wpgd} = d + \sigma^2 - d \cdot \boldsymbol{r}^\top \big(\boldsymbol{R} + (d + \sigma^2 + 1)\boldsymbol{I}\big)^{-1} \boldsymbol{r}. \tag{6.20}$$

6.5 Optimization Landscape of GLA

In Section 6.3, we demonstrated that GLA implements a data-dependent WPGD algorithm. Building on this, in Section 6.4, we analyze the optimization landscape for minimizing the one-step WPGD risk (cf. (6.2)) and show that a unique solution (up to rescaling) achieves the global minimum of the WPGD algorithm. However, in GLA, the search space for $\boldsymbol{\omega}$ is restricted and data-dependent, meaning that $\mathcal{L}^\star_{\text{wpgd}}$ in (6.2) represents the best possible risk a GLA model can achieve. In this section, we analyze the loss landscape for training a one-layer GLA model and explore the scenarios under which GLA can reach the optimal WPGD risk.

6.5.1 Multi-task prompt model

Following Definitions 6.1 and 6.2, we consider the multi-task prompt setting with K correlated tasks $(\boldsymbol{\beta}_k)_{k=1}^K$ and one query task $\boldsymbol{\beta}$ with distribution $\mathcal{N}(0, \boldsymbol{I})$. For each correlated task $k \in [K]$, a prompt of length n_k is drawn, consisting of IID input-label pairs $\{(\boldsymbol{x}_{ik}, y_{ik})\}_{i=1}^{n_k}$ where $y_{ik} \sim \mathcal{N}(\boldsymbol{\beta}_k^\top \boldsymbol{x}_{ik}, \sigma^2)$ to obtain sequence $\boldsymbol{Z}_k = \begin{bmatrix} \boldsymbol{x}_{1k} & \cdots & \boldsymbol{x}_{n_k,k} \\ y_{1k} & \cdots & y_{n_k,k} \end{bmatrix}^\top$. Let $n := \sum_{k=1}^K n_k$. The query is given by $\boldsymbol{z}_{n+1} = [\boldsymbol{x}_{n+1} \;\; 0]^\top$ with the corresponding label $y_{n+1} \sim \mathcal{N}(\boldsymbol{x}_{n+1}^\top \boldsymbol{\beta}, \sigma^2)$. All the features $\boldsymbol{x}_{ik}$ for $i \in [n_k], k \in [K]$ and query feature $\boldsymbol{x}_{n+1}$ are IID sampled from $\mathcal{N}(\boldsymbol{0}, \boldsymbol{\Sigma})$.

These sequences $(\boldsymbol{Z}_k)_{k=1}^K$, along with the query token $\boldsymbol{z}_{n+1}$, are concatenated to form a single prompt $\boldsymbol{Z}$. Recall (GLA), and let $f_{\text{gla}}(\boldsymbol{Z})$ denote the GLA prediction as defined in Theorem 6.1. Additionally, consider the model construction presented in (6.9), where $\boldsymbol{P}_q, \boldsymbol{P}_k \in \mathbb{R}^{d\times d}$ are trainable parameters. The GLA optimization problem is described as follows:

$$\begin{aligned} \mathcal{L}_{\text{gla}}^\star := & \min_{\boldsymbol{P}_k, \boldsymbol{P}_q \in \mathbb{R}^{d\times d}, G\in\mathcal{G}} \ \mathcal{L}_{\text{gla}}(\boldsymbol{P}_k, \boldsymbol{P}_q, G), \\ \text{where} \quad & \mathcal{L}_{\text{gla}}(\boldsymbol{P}_k, \boldsymbol{P}_q, G) = \mathbb{E}\left[(y - f_{\text{gla}}(\boldsymbol{Z}))^2\right]. \end{aligned} \tag{6.21}$$

Here, $G(\cdot)$ represents the gating function and $\mathcal{G}$ denotes the function search space, which is determined by the chosen gating strategies (cf. Table 1 in [210]).

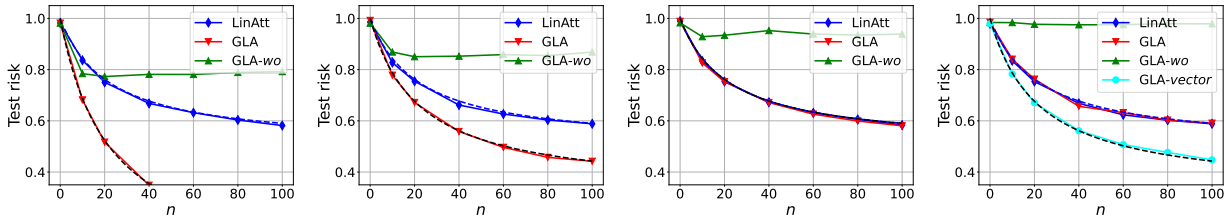
Note that 1) the task vectors $(\boldsymbol{\beta}_k)_{k=1}^K$ are not explicitly shown in the prompt, 2) in-context examples $(\boldsymbol{x}_{ik}, y_{ik})$ are randomly drawn, and 3) the gating function is applied to the tokens/input samples $(\boldsymbol{Z}_k)_{k=1}^K$. Given the above three evidences, the implicit weighting induced by the GLA model varies across different prompts, and it prevents the GLA from learning the optimal weighting. To address this, we introduce delimiters to mark the boundary of each task. Let $(\boldsymbol{d}_k)_{k=1}^K$ be the delimiters that determine stop of the tasks. Specifically, the final prompt is given by

$$\boldsymbol{Z} = \begin{bmatrix} \boldsymbol{Z}_1^\top & \boldsymbol{d}_1 & \cdots & \boldsymbol{Z}_K^\top & \boldsymbol{d}_K & \boldsymbol{z}_{n+1} \end{bmatrix}^\top. \tag{6.22}$$

Additionally, to decouple the influence of gating and data, we envision that each token is $\boldsymbol{z}_i = [\boldsymbol{x}_i^\top \ \ y_i \ \ \boldsymbol{c}_i^\top]^\top$ where $\boldsymbol{c}_i \neq \boldsymbol{0} \in \mathbb{R}^p$ is the contextual features with p being its dimension and $(\boldsymbol{x}_i, y_i)$ are the data features. Let $\bar{\boldsymbol{c}}_0, \cdots, \bar{\boldsymbol{c}}_K \in \mathbb{R}^p$ be $K+1$ contextual feature vectors.

- For task prompts $\boldsymbol{Z}_k$: Contextual features are set to a fixed vector $\bar{\boldsymbol{c}}_0 \neq \boldsymbol{0}$.
- For delimiters $\boldsymbol{d}_k$: Data features are set to zero (e.g., $\boldsymbol{x}_i = \boldsymbol{0}$ and $y_i = 0$) so that $\boldsymbol{d}_k = [\boldsymbol{0}_{d+1}^\top \ \bar{\boldsymbol{c}}_k^\top]^\top$.

Explicit delimiters have been utilized in addressing real-world problems [200, 7, 51] due to their ability to improve efficiency and enhance generalization, particularly in task-mixture or multi-document scenarios. To further motivate the introduction of $(\boldsymbol{d}_k)_{k=1}^K$, we present in Figure 6.1 the results of GLA training with and without delimiters, represented by the red and green curves, respectively. The black dashed curves indicate the optimal WPGD loss, $\mathcal{L}_{\text{wpgd}}^\star$, under different scenarios. Notably, training GLA without delimiters (the green solid curve) performs strictly worse. In contrast, training with delimiters can achieve optimal performance under certain conditions (see Figures 6.1a, 6.1b, and 6.1c). Theorem 6.5, presented in the next section, provides a theoretical explanation for these observations, including



(a) $(r_1, r_2) = (0, 1)$ (b) $(r_1, r_2) = (0.2, 0.8)$ (c) $(r_1, r_2) = (0.5, 0.5)$ (d) $(r_1, r_2) = (0.8, 0.2)$

Figure 6.1: We consider four different types of model training: `LinAtt` (blue solid): Standard linear attention training. `GLA` (red solid): GLA training using prompts with delimiters (see (6.22)) and scalar gating. `GLA-wo` (green solid): GLA training using prompts without delimiters and with scalar gating. `GLA-vector` (cyan solid): GLA training using prompts with delimiters and vector gating. The blue and black dashed curves represent the optimal linear attention and WPGD risks from (6.27) and (6.20), respectively, as the number of in-context examples n increases. Implementation details are provided in Appendix E.1.

the misalignment observed in Figure 6.1d. Further discussion and experimental details are provided in Section 6.5.2 and Appendix E.1.

6.5.2 Loss landscape of one-layer GLA

Given the input tokens with extended dimension, to ensure that GLA still implements WPGD as in Theorem 6.1, we propose the following model construction.

$$\tilde{\boldsymbol{W}}_k = \begin{bmatrix} \boldsymbol{W}_k & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix}, \quad \tilde{\boldsymbol{W}}_q = \begin{bmatrix} \boldsymbol{W}_q & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix}, \quad \text{and} \quad \tilde{\boldsymbol{W}}_v = \begin{bmatrix} \boldsymbol{W}_v & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix}. \tag{6.23}$$

Here, $\tilde{\boldsymbol{W}}_{k,q,v} \in \mathbb{R}^{(d+p+1)\times(d+p+1)}$ and $\boldsymbol{W}_{k,q,v} \in \mathbb{R}^{(d+1)\times(d+1)}$ are constructed via (6.9). The main idea is to set the last p rows and columns of attention matrices to zeros, ensuring that the delimiters do not affect the final prediction.

Assumption 6.2 *Contextual feature vectors $\bar{\boldsymbol{c}}_0, \cdots, \bar{\boldsymbol{c}}_K$ are linearly independent, and activation function $\phi(z) : \mathbb{R} \to [0, 1]$ is continuous, satisfying $\phi(-\infty) = 0$ and $\phi(+\infty) = 1$.*

Assumption 6.3 *The correlation between context tasks $(\boldsymbol{\beta}_k)_{k=1}^K$ and query task $\boldsymbol{\beta}$ satisfies $\mathbb{E}[\boldsymbol{\beta}_i^\top \boldsymbol{\beta}_j] = 0$ and $\mathbb{E}[\boldsymbol{\beta}_i^\top \boldsymbol{\beta}] \le \mathbb{E}[\boldsymbol{\beta}_j^\top \boldsymbol{\beta}]$ for all $1 \le i \le j \le K$.*

Given context examples $\{(\boldsymbol{X}_k, \boldsymbol{y}_k) := (\boldsymbol{x}_{ik}, y_{ik})_{i=1}^{n_k}\}_{k=1}^K$, define the concatenated data $(\boldsymbol{X}, \boldsymbol{y})$ as follows:

$$\boldsymbol{X} = \begin{bmatrix} \boldsymbol{X}_1^\top & \cdots & \boldsymbol{X}_K^\top \end{bmatrix}^\top \in \mathbb{R}^{n\times d}, \quad \text{and} \quad \boldsymbol{y} = \begin{bmatrix} \boldsymbol{y}_1^\top & \cdots & \boldsymbol{y}_K^\top \end{bmatrix}^\top \in \mathbb{R}^n. \tag{6.24}$$

Based on the assumptions above, we are able to establish the equivalence between optimizing one-layer GLA and optimizing one-step WPGD predictor under scalar gating.

Theorem 6.5 (Scalar Gating) *Suppose Assumption 6.2 holds. Recap the function* $\mathcal{L}_{wpgd}(\boldsymbol{P}, \boldsymbol{\omega})$ *from* (6.2) *with dataset* $(\boldsymbol{X}, \boldsymbol{y})$ *defined in* (6.24). *Consider* (GLA) *with input prompt* $\boldsymbol{Z}$ *defined in* (6.22), *the model constructions described in* (6.23), *and scalar gating* $G(\boldsymbol{z}) = \phi(\boldsymbol{w}_g^\top \boldsymbol{z})\mathbf{1}_{(d+p+1)\times(d+p+1)}$, *where* $\boldsymbol{w}_g$ *is a trainable parameter. Then, the optimal risk* $\mathcal{L}_{gla}^\star$ *defined in* (6.21) *obeys*

$$\mathcal{L}_{gla}^\star = \mathcal{L}_{wpgd}^{\star,\mathcal{W}}, \quad \textit{where} \quad \mathcal{L}_{wpgd}^{\star,\mathcal{W}} := \min_{\boldsymbol{P}\in\mathbb{R}^{d\times d}, \boldsymbol{\omega}\in\mathcal{W}} \mathcal{L}_{wpgd}(\boldsymbol{P}, \boldsymbol{\omega}). \tag{6.25}$$

Here, $\mathcal{W} := \left\{ \begin{bmatrix} \omega_1 \mathbf{1}_{n_1}^\top & \cdots & \omega_K \mathbf{1}_{n_K}^\top \end{bmatrix}^\top \in \mathbb{R}^n \;\middle|\; 0 \le \omega_i \le \omega_j \le 1,\ \forall 1 \le i \le j \le K \right\}$.

Additionally, suppose Assumption 6.3 holds and $n_i = n_j$, *for all* $i, j \in [K]$. *Let* $\mathcal{L}_{wpgd}^\star$ *be the optimal WPGD risk (cf.* (6.2)*). Then* $\mathcal{L}_{gla}^\star$ *satisfies*

$$\mathcal{L}_{gla}^\star = \mathcal{L}_{wpgd}^\star. \tag{6.26}$$

Assumption 6.2 ensures that any $\boldsymbol{\omega}$ in $\mathcal{W}$ can be achieved by an appropriate choice of gating parameters. Furthermore, Assumption 6.3 guarantees that the optimal choice of $\boldsymbol{\omega}$ under the WPGD objective lies within the search space $\mathcal{W}$. The proof is provided in Appendix E.4.1.

In Figure 6.1, we conduct model training to validate our findings. Consider the setting where $K = 2$ and let $(r_1, r_2) = \big(\mathbb{E}[\boldsymbol{\beta}_1^\top \boldsymbol{\beta}]/d, \mathbb{E}[\boldsymbol{\beta}_2^\top \boldsymbol{\beta}]/d\big)$. In Figures 6.1a, 6.1b, and 6.1c, Assumption 6.3 holds, and the GLA results (shown in solid red) align with the optimal WPGD risk (represented by the dashed black curves), validating (6.26). However, in Figure 6.1d, since $r_1 > r_2$, Assumption 6.3 does not hold, and as a result, the optimal GLA loss $\mathcal{L}_{\text{gla}}^\star$ obtained from (6.25) is lower than the optimal WPGD loss $\mathcal{L}_{\text{wpgd}}^\star$. Further experimental details are deferred to Appendix E.1.

Loss landscape of vector gating. Till now, much of our discussion has focused on the scalar gating setting. It is important to highlight that, even in the scalar-weighting context, analyzing the WPGD problem remains non-trivial due to the joint optimization over $(\boldsymbol{P}, \boldsymbol{\omega})$. However, as demonstrated in Theorem 6.5, scalar gating can only express weightings within the set $\mathcal{W}$. If Assumption 6.3 does not hold, $\mathcal{L}_{\text{gla}}^\star$ cannot achieve the optimal WPGD loss (see the misalignment between red solid curve, presenting $\mathcal{L}_{\text{gla}}^\star$, and black dashed curve, presenting $\mathcal{L}_{\text{wpgd}}^\star$ in Figure 6.1d). We argue that vector gating overcomes this limitation by applying distinct weighting mechanisms across different dimensions, facilitating stronger expressivity.

Theorem 6.6 (Vector Gating) *Suppose Assumption 6.2 holds. Consider* (GLA) *with input prompt* $\boldsymbol{Z}$ *from* (6.22)*, the model constructions from* (6.23) *but with* $\tilde{\boldsymbol{W}}_v = [\boldsymbol{0}_{(d+p+1)\times d} \quad \boldsymbol{u} \quad \boldsymbol{0}_{(d+p+1)\times p}]^\top$*, and a vector gating* $G(\boldsymbol{z}) = \phi(\boldsymbol{W}_g \boldsymbol{z})\boldsymbol{1}_{d+p+1}^\top$*. Recap Problem* (6.21) *with prediction defined as* $f_{gla}(\boldsymbol{Z}) := \boldsymbol{o}_{n+1}^\top \boldsymbol{h}$*, where* $\boldsymbol{h} \in \mathbb{R}^{d+p+1}$ *is a linear prediction head. Here,* $\boldsymbol{u}$*,* $\boldsymbol{W}_g$ *and* $\boldsymbol{h}$ *are trainable parameters. Let* $\mathcal{L}_{gla\text{-}v}^\star$ *denote its optimal risk, and* $\mathcal{L}_{wpgd}^\star$ *be defined as in* (6.2)*. Then, the optimal risk obeys* $\mathcal{L}_{gla\text{-}v}^\star = \mathcal{L}_{wpgd}^\star$*.*

In Theorem 6.5, the equivalence between $\mathcal{L}_{\text{gla}}^\star$ and $\mathcal{L}_{\text{wpgd}}^\star$ is established only when both Assumptions 6.2 and 6.3 are satisfied. In contrast, Theorem 6.6 demonstrates that applying vector gating requires only Assumption 6.2 to establish $\mathcal{L}_{\text{gla-}v}^\star = \mathcal{L}_{\text{wpgd}}^\star$. Specifically, under the bounded activation model of Assumption 6.2, scalar gating is unable to express non-monotonic weighting schemes. For instance, suppose there are two tasks: Even if Task 1 is more relevant to the query, Assumption 6.2 will assign a higher (or equal) weight to examples in Task 2 resulting in sub-optimal prediction. Theorem 6.6 shows that vector gating can avoid such bottlenecks by potentially encoding tasks in distinct subspaces. To verify these intuitions, in Figure 6.1d, we train a GLA model with vector gating and results are presented in cyan curve, which outperform the scalar gating results (red solid) and align with the optimal WPGD loss (black dashed).

Loss landscape of One-layer linear attention. Given the fact that linear attention is equivalent to (GLA) when implemented with all ones gating, that is, $\boldsymbol{G}_i \equiv \boldsymbol{1}$, we derive the following corollary. Consider training a standard single-layer linear attention, i.e., by setting $\boldsymbol{G}_i \equiv \boldsymbol{1}$ in (GLA), and let $f_{\text{att}}(\boldsymbol{Z})$ be its prediction. Let $\mathcal{L}_{\text{att}}^\star$ be the corresponding optimal risk following (6.21).

Corollary 6.3 *Consider a single-layer linear attention following model construction in* (6.9) *and let linear data as given in Definition 6.2. Let* $\boldsymbol{R}$ *and* $\boldsymbol{r}$ *be the corresponding correlation matrix and vector as defined in Definition 6.1. Suppose* $\boldsymbol{\Sigma} = \boldsymbol{I}$*. Then, the optimal risk obeys*

$$\mathcal{L}_{att}^\star := \min_{\boldsymbol{P} \in \mathbb{R}^{d\times d}} \mathcal{L}_{wpgd}(\boldsymbol{P}, \boldsymbol{\omega} = \boldsymbol{1}) = d + \sigma^2 - \frac{d(\boldsymbol{1}^\top \boldsymbol{r})^2}{n(d+\sigma^2+1) + \boldsymbol{1}^\top \boldsymbol{R} \boldsymbol{1}}. \tag{6.27}$$

Corollary 6.4 (Benefit of Gating) *Consider the same setting as discussed in Corollary 6.3, and suppose Assumption 6.2 holds. Then, we have that* $\mathcal{L}_{att}^\star \geq \mathcal{L}_{gla}^\star$*. Additionally, if Assumption 6.3 holds, we obtain*

$$\mathcal{L}_{att}^\star - \mathcal{L}_{gla}^\star = d \cdot \boldsymbol{r}^\top \left(\boldsymbol{R}_+^{-1} - \frac{\boldsymbol{1}\boldsymbol{1}^\top}{\boldsymbol{1}^\top \boldsymbol{R}_+ \boldsymbol{1}} \right) \boldsymbol{r} \geq 0, \quad \text{where} \quad \boldsymbol{R}_+ := \boldsymbol{R} + \left(d + \sigma^2 + 1\right) \boldsymbol{I}.$$

The proof of this corollary is directly from (6.20), (6.26) and (6.27). In the Figure 6.1, blue solid curves represent the linear attention results and blue dashed are the theory curves following (6.27). The two curves are aligned in all the subfigures, which validate our Corollary 6.3. More implementation details are deferred to Appendix E.1.

CHAPTER 7

Task-specific Prompt Optimization in Linear Attention

7.1 Introduction

In a basic ICL setting, we construct an input sequence $\boldsymbol{Z}$ that contains a query to label and related input-label demonstrations. We feed $\boldsymbol{Z}$ to a sequence model f to predict the label y of this query. Thus, the ICL optimization can be written as

$$\min_{\boldsymbol{W}} \mathbb{E}_{y,\boldsymbol{Z}}[\ell(y, f_{\boldsymbol{W}}(\boldsymbol{Z}))], \tag{7.1}$$

where $\boldsymbol{W}$ denotes the weights of f and $\ell(y, \cdot) : \mathbb{R} \to \mathbb{R}$ is the loss function. In practice, however, ICL examples are typically paired with task-specific prompts. These prompts can be manually crafted or automatically generated using methods like differentiable optimization or zero-order search. In fact, the model can often solve the task without any ICL example (zero-shot) by solely relying on the prompt. This motivates a deeper understanding of the synergies between prompt-tuning and in-context learning. Concretely, consider a multitask learning setting where we first sample a task

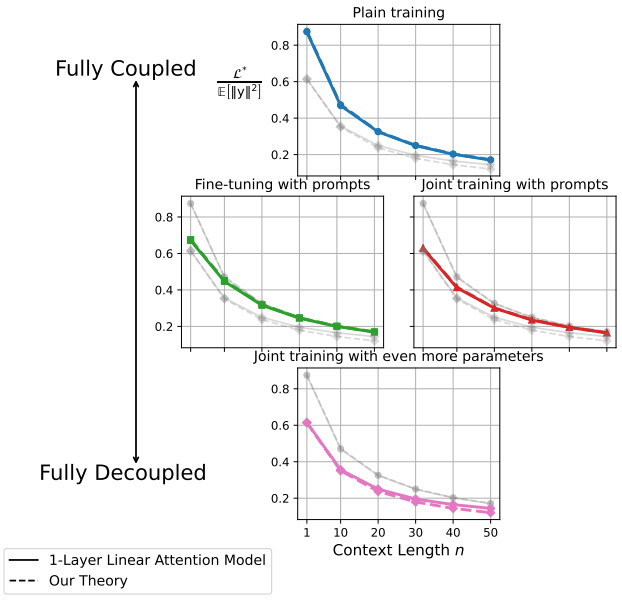


Figure 7.1: Overview of our work: We present a theoretical analysis of a 1-layer linear attention model for multi-task ICL, examining different configurations of task-specific parameters. By progressively introducing more task-specific parameters across various training settings, we achieve complete **covariance-mean decoupling**, leading to an optimal multi-task ICL loss.

t and then sample $(\boldsymbol{Z}, y)$ from the associated task distribution. In this case, a standard approach is crafting a dedicated prompt $\boldsymbol{p}_t$ to feed with the input sequence. The joint optimization of the weights $\boldsymbol{W}$ and prompts $\boldsymbol{P} = (\boldsymbol{p}_t)_{t\geq 1}$ takes the form

$$\min_{\boldsymbol{W},\boldsymbol{P}} \mathbb{E}[\ell(y, f_{\boldsymbol{W}}(\boldsymbol{Z}, \boldsymbol{p}_t))]. \tag{7.2}$$

Contrasting (7.1) with (7.2) motivate a few fundamental questions:

(Q1) How do ICL and prompt-tuning synergistically contribute to learning?

(Q2) In practice, we first pretrain a model via (7.1) and then tune a task-specific prompt. Does joint training have an advantage over this?

(Q3) Can utilizing additional task-specific parameters together with prompts, further boost performance?

To answer these questions, we conduct a comprehensive investigation of the optimization landscape for attention weights and task-specific prompts within a multitask dataset model, examining both joint optimization approaches and sequential strategies involving attention weight pretraining followed by prompt-tuning. We derive closed-form solutions for optimal parameters and loss landscapes, introducing the novel concept of "covariance-mean decoupling" to elucidate the impact of different training strategies on model performance. While previous theoretical research has primarily focused on attention weight optimization, we extend the analysis to include tunable prompts in a multi-task linear regression framework, addressing the practical scenario where models fine-tune task-specific modules on fixed backbones. Notably, our work advances beyond existing research by incorporating non-zero task mean and non-isotropic covariance considerations, revealing why fine-tuning may not consistently enhance performance. Our theoretical framework demonstrates how varying performance gains across training strategies can be attributed to covariance-mean decoupling, providing both theoretical foundations and practical insights for optimizing attention-based models through careful design of tunable components.

Our key contributions are:

- ◇ **Comprehensive analysis of training strategies (§7.3):** We analyze multi-task linear regression with a linear attention layer and tunable parameters, covering joint optimization and pretraining-finetuning methods. A unified parameterization allows for closed-form solutions of optimal parameters and loss landscapes, generalizing prior work.

- ◇ **Mean-covariance decoupling concept (§7.3.2):** We introduce covariance-mean decoupling through closed-form loss landscape analysis, demonstrating its correlation with model performance—the greater the decoupling, the better the model's predictions.
- ◇ **Path to optimal in-context learning (§7.4):** Our analysis guides the design of attention models, emphasizing the importance of training sequence and parameter selection. We propose a model design achieving full covariance-mean decoupling, offering a strategy for improving in-context learning.

These contributions provide a deeper understanding of prompt-tuning and weight optimization, offering insights for designing more effective models that minimize loss and maximize performance.

7.2 Preliminaries and Problem Setup

Our results are presented in a finite-dimensional setting.

7.2.1 In-context learning

We consider an in-context learning (ICL) problem with demonstrations $(\boldsymbol{x}_i, y_i)_{i=1}^{n+1}$, and the input sequence $\boldsymbol{Z}$ is defined by removing y_{n+1} as follows:

$$\boldsymbol{Z} = [\boldsymbol{z}_1 \ \ldots \ \boldsymbol{z}_n \ \boldsymbol{z}]^\top = \begin{bmatrix} \boldsymbol{x}_1 & \ldots & \boldsymbol{x}_n & \boldsymbol{x} \\ y_1 & \ldots & y_n & 0 \end{bmatrix}^\top = \begin{bmatrix} \boldsymbol{X}^\top & \boldsymbol{x} \\ \boldsymbol{y}^\top & 0 \end{bmatrix}^\top \in \mathbb{R}^{(n+1)\times(d+1)}. \tag{7.3}$$

Here, $\boldsymbol{z} = [\boldsymbol{x}^\top \ 0]^\top$ is the query token where $\boldsymbol{x} := \boldsymbol{x}_{n+1}$, and $\boldsymbol{X} = [\boldsymbol{x}_1 \ \cdots \ \boldsymbol{x}_n]^\top \in \mathbb{R}^{n\times d}$, $\boldsymbol{y} = [y_1 \ \cdots \ y_n]^\top \in \mathbb{R}^n$. Then, we aim for a sequence model to predict the associated label $y := y_{n+1}$ of the given input sequence $\boldsymbol{Z}$. In this work, we consider the following data generation of $(\boldsymbol{Z}, y)$. We will refer to $(\boldsymbol{X}, \boldsymbol{y})$, $\boldsymbol{x}$, and y as contexts, query feature and the label to predict, respectively.

Definition 7.1 (Single-task ICL) *Given a task mean $\boldsymbol{\mu} \in \mathbb{R}^d$, and covariances $\boldsymbol{\Sigma}_{\boldsymbol{x}}, \boldsymbol{\Sigma}_{\boldsymbol{\beta}} \succ 0 \in \mathbb{R}^{d\times d}$. The input sequence and its associated label, i.e., $(\boldsymbol{Z}, y)$ with $\boldsymbol{Z}$ denoted in (7.3), are generated as follows:*

- *A task parameter $\boldsymbol{\beta}$ is generated from a Gaussian prior $\boldsymbol{\beta} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma}_{\boldsymbol{\beta}})$.*
- *Conditioned on $\boldsymbol{\beta}$, for $i \in [n+1]$, $(\boldsymbol{x}_i, y_i)$ is generated by $\boldsymbol{x}_i \sim \mathcal{N}(0, \boldsymbol{\Sigma}_{\boldsymbol{x}})$ and $y_i \sim \mathcal{N}(\boldsymbol{x}_i^\top \boldsymbol{\beta}, \sigma^2)$.*

Here, $\sigma \geq 0$ is the noise level.

In this work, we study the task-mixture ICL problem where the task parameter $\boldsymbol{\beta}$ of each input sequence is sampled from K different distributions, $K \geq 1$.

Definition 7.2 (Multi-task ICL) *Consider a multi-task ICL problem with K different tasks. Each task generates $(\boldsymbol{Z}, y) \sim \mathcal{D}_k$ following Definition 7.1 using shared feature distribution $\boldsymbol{x}_i \sim \mathcal{N}(0, \boldsymbol{\Sigma}_{\boldsymbol{x}})$, $i \in [n+1]$ but distinct task distributions $\boldsymbol{\beta}_k \sim \mathcal{N}(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_{\boldsymbol{\beta}_k})$ with mean $\boldsymbol{\mu}_k$ and covariance $\boldsymbol{\Sigma}_{\boldsymbol{\beta}_k}$ for $k \in [K]$.*

Additionally, let $\{\pi_k\}_{k=1}^K$ be the probabilities of each task, satisfying $\sum_{k=1}^K \pi_k = 1$ and $\pi_k \geq 0$.

We consider a task-aware multi-task ICL setting. Specifically, when a task is selected according to π_k, its task index k is known.

Let $\bar{\mathcal{D}} := \sum_{k=1}^K \pi_k \mathcal{D}_k$ be the mixture of distributions and given sequence model $f : \mathbb{R}^{(n+1)\times(d+1)} \to \mathbb{R}$, we define the multi-task ICL objective as follows:

$$\mathcal{L}(f) = \mathbb{E}_{(\boldsymbol{Z},y)\sim\bar{\mathcal{D}}}[(y - f(\boldsymbol{Z}))^2]. \tag{7.4}$$

Notably, the multi-task ICL defined in Definition 7.2 differs from conventional multi-task learning [29, 219, 115] where finite examples optimize task-specific parameters. We instead parameterize task distributions with $\boldsymbol{\mu}_k$ and $\boldsymbol{\Sigma}_{\boldsymbol{\beta}_k}$, sampling unseen test tasks $\boldsymbol{\beta}_k$ and focusing on distribution-level generalization. We address the meta-learning objective in (7.4), treating each distribution as a meta-learning problem (c.f. Definition 7.1).

7.2.2 Single-layer linear attention

Considering the task-aware multi-task ICL problem as described in Section 7.2.1, we explore the benefits of using *task-specific prompts* to enhance the performance.

Definition 7.3 (Task-specific prompts) *Recap input sequence $\boldsymbol{Z}$ from (7.3). Given a task index k, $k \in [K]$, let $\boldsymbol{p}_k \in \mathbb{R}^{d+1}$ represent its corresponding trainable prompt token. Then the input sequence of task k is denoted by:*

$$\boldsymbol{Z}^{(k)} = [\boldsymbol{p}_k \ \boldsymbol{z}_1 \ \ldots \ \boldsymbol{z}_n \ \boldsymbol{z}]^\top \in \mathbb{R}^{(n+2)\times(d+1)}. \tag{7.5}$$

Our work focuses on the single-layer linear attention model in solving multi-task ICL problem with data distribution following Definition 7.2. Given an input sequence $\boldsymbol{Z}^{(k)}$ corresponding to task k as defined in (7.5), let $\boldsymbol{W}_q, \boldsymbol{W}_k, \boldsymbol{W}_v \in \mathbb{R}^{(d+1)\times(d+1)}$ denote the query,

key, and value parameters. Then the single-layer linear attention model outputs

$$\mathrm{att}(\boldsymbol{Z}^{(k)}) = (\boldsymbol{Z}^{(k)}\boldsymbol{W}_q\boldsymbol{W}_k^\top(\boldsymbol{Z}^{(k)})^\top)\boldsymbol{M}\boldsymbol{Z}^{(k)}\boldsymbol{W}_v$$

where $\mathrm{att}(\cdot) : \mathbb{R}^{(n+2)\times(d+1)} \to \mathbb{R}^{(n+2)\times(d+1)}$. Here, inspired by the previous work [3], we apply mask matrix $\boldsymbol{M} = \begin{bmatrix} \boldsymbol{I}_{n+1} & \boldsymbol{0}_{n+1} \\ \boldsymbol{0}_{n+1}^\top & 0 \end{bmatrix}$ to to separate the $(n+1)$-colomun context and the query $\boldsymbol{z}$ in $\boldsymbol{Z}^{(k)}$. Let $\boldsymbol{h} \in \mathbb{R}^{d+1}$ be the linear head that enables the single-layer linear attention model to map the input sequence to the prediction. Additionally, for simplification and without loss of generality, let $\boldsymbol{A} := \boldsymbol{W}_q\boldsymbol{W}_k^\top$ and $\boldsymbol{a} := \boldsymbol{W}_v\boldsymbol{h}$. Then the prediction returns

$$\hat{y} := f_{\mathrm{att}}(\boldsymbol{Z}^{(k)}) = (\boldsymbol{z}^\top\boldsymbol{A}(\boldsymbol{Z}^{(k)})^\top)\boldsymbol{M}\boldsymbol{Z}^{(k)}\boldsymbol{a}. \tag{7.6}$$

This is a more general form than the widely discussed [207] case, which uses the bottom-right entry of the attention layer output as prediction (and equivalent to a one-hot head $\boldsymbol{h} = \boldsymbol{e}_{d+1}$), i.e., $f_{\mathrm{att}}(\boldsymbol{Z}^{(k)}) = \mathrm{att}(\boldsymbol{Z}^{(k)})_{n+2,d+1}$.

7.2.3 Optimizing the attention model

Our goal is to understand how optimizing f_{att} in (7.6) results in in-context learning. To this aim, we introduce the following widely applied [207, 3] assumption on the model construction.

Assumption 7.1 (Preconditioning) *Given parameters $\boldsymbol{A} := \boldsymbol{W}_q\boldsymbol{W}_k^\top$ and $\boldsymbol{a} := \boldsymbol{W}_v\boldsymbol{h}$, they are constrained by*

$$\boldsymbol{A} = \begin{bmatrix} \boldsymbol{W}_{d\times d} & \boldsymbol{0}_{d\times 1} \\ *_{1\times d} & *_{1\times 1} \end{bmatrix}, \boldsymbol{a} = \begin{bmatrix} \boldsymbol{0}_{d\times 1} \\ 1_{1\times 1} \end{bmatrix}.$$

Here, we use $*$ to fill the entries that do not affect the final prediction, with subscripts indicating the dimensions. $\boldsymbol{W} \in \mathbb{R}^{d\times d}$ represents the parameter that governs $\boldsymbol{A}$.

Under Assumption 7.1, let the prompt token for task k be $\boldsymbol{p}_k = \begin{bmatrix} \bar{\boldsymbol{p}}_k \\ 1 \end{bmatrix}$, hence $\boldsymbol{Z}^{(k)} = \begin{bmatrix} \bar{\boldsymbol{p}}_k & \boldsymbol{X}^\top & \boldsymbol{x} \\ 1 & \boldsymbol{y}^\top & 0 \end{bmatrix}^\top$. The prediction of a single-layer linear attention model in (7.6) can then be written as:

$$\begin{aligned} f_{\mathrm{att}}(\boldsymbol{Z}^{(k)}) &= \boldsymbol{x}^\top\boldsymbol{W}\begin{bmatrix} \bar{\boldsymbol{p}}_k & \boldsymbol{X}^\top \end{bmatrix}\begin{bmatrix} 1 \\ \boldsymbol{y} \end{bmatrix} \\ &= \boldsymbol{x}^\top\boldsymbol{W}(\boldsymbol{X}^\top\boldsymbol{y} + \bar{\boldsymbol{p}}_k). \end{aligned} \tag{7.7}$$

It is worth noting that we set the last entry of $\boldsymbol{a}$ and each $\boldsymbol{p}_k$ for $k \in [K]$ to one for simplicity, as any nonzero value yields the same output of (7.7) due to rescaling invariance of: $\boldsymbol{a} \leftarrow \gamma_1 \boldsymbol{a}$, $\boldsymbol{A} \leftarrow \gamma_1^{-1} \boldsymbol{A}$, and $\bar{\boldsymbol{p}}_k \leftarrow \begin{bmatrix} \gamma_2^{-1} \bar{\boldsymbol{p}}_k \\ \gamma_2 \end{bmatrix}$ for any nonzero scalar γ_1, γ_2.

We define the set of tunable prompts as:

$$\boldsymbol{P} = [\bar{\boldsymbol{p}}_1 \ \ldots \ \bar{\boldsymbol{p}}_K]^\top \in \mathbb{R}^{K \times d}. \tag{7.8}$$

Notably, previous research [3] has rigorously proven that for single-layer linear attention applied to a single-task linear regression ICL problem with zero-mean features, i.e., $\mathbb{E}[\boldsymbol{x}_i] = \mathbb{E}[\boldsymbol{\beta}] = \boldsymbol{0}$, the optimal solution must conform to the structure specified in Assumption 7.1. This insight motivates our adoption of this assumption when extending the analysis to more complex multi-task ICL settings with non-zero task means. Building on this foundation, our work breaks new ground by exploring task-specific tuning for ICL across multiple tasks with varying means, offering a more general perspective on ICL. This represents a significant advancement in the field, as understanding the loss landscape in the simpler $\boldsymbol{W}$-preconditioned space is an essential step toward tackling the complexities of the full $\boldsymbol{W}_k, \boldsymbol{W}_q, \boldsymbol{W}_v$ parameter space.

Additionally, in Section 7.4, we show that it is possible to derive closed-form optimal solutions for $\boldsymbol{W}_k, \boldsymbol{W}_q, \boldsymbol{W}_v$ with both task-specific prompts and heads, even without relying on Assumption 7.1, further expanding the scope of our analysis.

Recall the attention predictor from (7.7) and loss function from (7.4). Consider the multi-task ICL problem defined in Definition 7.2 and let $\boldsymbol{W} \in \mathbb{R}^{d \times d}$ and $\boldsymbol{P} \in \mathbb{R}^{K \times d}$ be the tunable parameters corresponding to attention weights and task-specific prompt tokens. The multi-task ICL loss is given by

$$\begin{aligned} &\mathcal{L}(\boldsymbol{W}, \boldsymbol{P}) = \sum_{k=1}^{K} \pi_k \mathcal{L}_k(\boldsymbol{W}, \bar{\boldsymbol{p}}_k) \\ &\text{where } \mathcal{L}_k(\boldsymbol{W}, \bar{\boldsymbol{p}}_k) = \mathbb{E}_{(\boldsymbol{Z}, y) \sim \mathcal{D}_k} \left[(f_{\text{att}}(\boldsymbol{Z}^{(k)}) - y)^2 \right]. \end{aligned} \tag{7.9}$$

In this work, we address multi-task ICL problems by investigating and comparing three optimization settings: plain training, fine-tuning, and joint training.

Plain training: Plain training refers to a standard ICL problem that train an linear attention model without applying the task-specific prompts that are defined in Definition 7.3.

Therefore, following loss function (7.9), its objective can be defined via

$$\boldsymbol{W}_{\mathrm{pt}}^{\star} = \arg\min_{\boldsymbol{W}} \mathcal{L}(\boldsymbol{W}, \boldsymbol{P} = \boldsymbol{0}), \tag{7.10}$$

and $\mathcal{L}_{\mathrm{pt}}^{\star} = \mathcal{L}(\boldsymbol{W}_{\mathrm{pt}}^{\star}, \boldsymbol{P} = \boldsymbol{0})$ is the optimal loss.

Fine-tuning: Fine-tuning/Prompt-tuning involves training separate prompts for each task while keeping the attention weights fixed. The goal then is to fine-tune the prompt parameters $\boldsymbol{P}$ (c.f. (7.8)) for all the tasks $k \in [K]$. Suppose that parameter $\boldsymbol{W}$ is given, the the optimal $\boldsymbol{P}$ based on $\boldsymbol{W}$ is defined by:

$$\boldsymbol{P}^{\star}(\boldsymbol{W}) = \arg\min_{\boldsymbol{P}} \mathcal{L}(\boldsymbol{W}, \boldsymbol{P}).$$

In this work, we consider fine-tuning based on the plain pretrained model, that is, by setting $\boldsymbol{W} = \boldsymbol{W}_{\mathrm{pt}}^{\star}$ given in (7.10), and define the optimal solution by

$$\boldsymbol{P}_{\mathrm{ft}}^{\star} := \boldsymbol{P}^{\star}(\boldsymbol{W}_{\mathrm{pt}}^{\star}) = \arg\min_{\boldsymbol{P}} \mathcal{L}(\boldsymbol{W}_{\mathrm{pt}}^{\star}, \boldsymbol{P}). \tag{7.11}$$

The optimal loss is given via $\mathcal{L}_{\mathrm{ft}}^{\star} = \mathcal{L}(\boldsymbol{W}_{\mathrm{pt}}^{\star}, \boldsymbol{P}_{\mathrm{ft}}^{\star})$.

Joint training: In contrast, joint training involves jointly optimizing the attention weights $\boldsymbol{W}$ and prompt tokens $\boldsymbol{P}$. Hence, the optimization problem can be formulated as:

$$\boldsymbol{W}_{\mathrm{jt}}^{\star}, \boldsymbol{P}_{\mathrm{jt}}^{\star} = \arg\min_{\boldsymbol{W}, \boldsymbol{P}} \mathcal{L}(\boldsymbol{W}, \boldsymbol{P}). \tag{7.12}$$

The optimal loss is given via $\mathcal{L}_{\mathrm{jt}}^{\star} = \mathcal{L}(\boldsymbol{W}_{\mathrm{jt}}^{\star}, \boldsymbol{P}_{\mathrm{jt}}^{\star})$.

7.3 Main Results

In this section, we train and optimize the single-layer linear attention model in a multi-task linear regression ICL setting with dataset described in Definition 7.2, and characterize the loss landscape under different settings, i.e., $\mathcal{L}_{\mathrm{pt}}^{\star}$, $\mathcal{L}_{\mathrm{ft}}^{\star}$ and $\mathcal{L}_{\mathrm{jt}}^{\star}$ in Section 7.2.3.

7.3.1 Optimization landscape

Recap the multi-task ICL dataset from Definition 7.2 where $\boldsymbol{\Sigma}_{\boldsymbol{x}}$ is the shared covariance matrix of the input features and $\{(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_{\boldsymbol{\beta}_k})\}_{k=1}^{K}$ are the task mean vectors and covariance

matrices. In the main paper, we consider noiseless data setting where $\sigma = 0$. We defer the exact analysis considering noisy labels to Appendix.

Consider any data distribution in Definition 7.1 and let $\boldsymbol{\beta} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma}_{\boldsymbol{\beta}})$. $\boldsymbol{\beta}$ can be rewritten via $\boldsymbol{\beta} = \bar{\boldsymbol{\theta}} + \boldsymbol{\mu}$ with $\bar{\boldsymbol{\theta}} \sim \mathcal{N}(0, \boldsymbol{\Sigma}_{\boldsymbol{\beta}})$. Then under the noiseless setting ($\sigma = 0$), the associated label $y_i = \boldsymbol{x}_i^\top \boldsymbol{\beta}$ is generated via

$$y_i = \underbrace{\boldsymbol{x}_i^\top \bar{\boldsymbol{\theta}}}_{\text{debiased}} + \boldsymbol{x}_i^\top \boldsymbol{\mu}. \tag{7.13}$$

Here we describe $\boldsymbol{x}_i^\top \bar{\boldsymbol{\theta}}$ as debiased since $\mathbb{E}[\boldsymbol{x}_i^\top \bar{\boldsymbol{\theta}}] = 0$. In this work, we investigate how task-specific prompts can help to capture individual task means such that learning task means ($\{\boldsymbol{\mu}_k\}_{k=1}^K$) and covariances ($\{\boldsymbol{\Sigma}_{\boldsymbol{\beta}_k}\}_{k=1}^K$) can be decoupled via optimizing prompts $\boldsymbol{P}$ and the attention weight $\boldsymbol{W}$. We say the model *fully decouples* mean and covariance if the optimized attention weight $\boldsymbol{W}^\star$ is only determined by the debiased term as shown in (7.13), with prompts responsible for capturing the bias introduced by the non-zero means.

To start with, recap from Definition 7.2 where task k has probability π_k and its task vector follows distribution $\boldsymbol{\beta}_k \sim \mathcal{N}(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_{\boldsymbol{\beta}_k})$. Following (7.13), we define the debiased and biased mixed-task covariances (variant with $\boldsymbol{\Sigma}_{\boldsymbol{x}}$ prior) as follows:

$$\text{Debiased: } \bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} = \boldsymbol{\Sigma}_{\boldsymbol{x}} \sum_{k=1}^K \pi_k \, \mathbb{E}[(\boldsymbol{\beta}_k - \boldsymbol{\mu}_k)(\boldsymbol{\beta}_k - \boldsymbol{\mu}_k)^\top]; \tag{7.14a}$$

$$\text{Biased:} \quad \tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} = \boldsymbol{\Sigma}_{\boldsymbol{x}} \sum_{k=1}^K \pi_k \, \mathbb{E}[\boldsymbol{\beta}_k \boldsymbol{\beta}_k^\top]. \tag{7.14b}$$

Note that they satisfy $\bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} = \boldsymbol{\Sigma}_{\boldsymbol{x}} \sum_{k=1}^K \pi_k \boldsymbol{\Sigma}_{\boldsymbol{\beta}_k}$ and $\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} = \boldsymbol{\Sigma}_{\boldsymbol{x}} \sum_{k=1}^K \pi_k (\boldsymbol{\Sigma}_{\boldsymbol{\beta}_k} + \boldsymbol{\mu}_k \boldsymbol{\mu}_k^\top)$.

We first analyze the plain training setting where no additional task-specific parameters are introduced, and all K tasks are mixed together.

Theorem 7.1 (Plain training) *Consider training a single-layer linear attention model in solving multi-task ICL problem with dataset defined in Definition 7.2 and model construction as described in Assumption 7.1. Let the optimal solution $\boldsymbol{W}_{pt}^\star$ (c.f. (7.10)) and the minimal plain training loss $\mathcal{L}_{pt}^\star$ as defined in Section 7.2.3. Additionally, let $\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}$ be defined in (7.14) and $\bar{\boldsymbol{W}}_{pt}^\star = \boldsymbol{\Sigma}_{\boldsymbol{x}} \boldsymbol{W}_{pt}^\star$. Then the solution $\bar{\boldsymbol{W}}_{pt}^\star$ and optimal loss $\mathcal{L}_{pt}^\star$ satisfy*

$$\bar{\boldsymbol{W}}_{pt}^\star = \tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} \left((n+1) \tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} + \mathtt{tr}\left(\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} \right) \boldsymbol{I} \right)^{-1},$$

$$\mathcal{L}_{pt}^\star = \mathtt{tr}\left(\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} \right) - n \cdot \mathtt{tr}\left(\bar{\boldsymbol{W}}_{pt}^\star \tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} \right).$$

Note that the above solution and optimal loss are identical to those in previous work [116] when considering a single-task ICL problem with with task vector following distribution $\boldsymbol{\beta} \sim \mathcal{N}(0, \tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}})$.

Theorem 7.2 (Fine-tuning) *Suppose a pretrained model as described in Theorem 7.1 is given with $\boldsymbol{W}_{pt}^{\star}$ being its optimal solution. Consider fine-tuning this model with task-specific prompts as defined in Definition 7.3, and let the optimal prompt matrix $\boldsymbol{P}_{ft}^{\star}$ (c.f. (7.11)) and the minimal fine-tuning loss $\mathcal{L}_{ft}^{\star}$ be defined in Section 7.2.3. Additionally, let $\bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}, \tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}$ be defined in (7.14) and $\bar{\boldsymbol{W}}_{pt}^{\star} = \boldsymbol{\Sigma}_{\boldsymbol{x}} \boldsymbol{W}_{pt}^{\star}$, and define the mean matrix*

$$\boldsymbol{M}_{\mu} = [\boldsymbol{\mu}_1 \ \cdots \ \boldsymbol{\mu}_K]^{\top} \in \mathbb{R}^{K \times d}.$$

Then the solution $\boldsymbol{P}_{ft}^{\star}$ and optimal loss $\mathcal{L}_{ft}^{\star}$ satisfy

$$\boldsymbol{P}_{ft}^{\star} = \boldsymbol{M}_{\mu} \left((\bar{\boldsymbol{W}}_{pt}^{\star})^{-1} - n\boldsymbol{I} \right) \boldsymbol{\Sigma}_{\boldsymbol{x}},$$
$$\mathcal{L}_{ft}^{\star} = \mathcal{L}_{pt}^{\star} - \mathtt{tr}\left((\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} - \bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}) (n\bar{\boldsymbol{W}}_{pt}^{\star} - \boldsymbol{I})^{\top} (n\bar{\boldsymbol{W}}_{pt}^{\star} - \boldsymbol{I}) \right).$$

Results in Theorem 7.2 show that, fine-tuning achieves better loss than plain training, $\mathcal{L}_{\mathrm{ft}}^{\star} \leq \mathcal{L}_{\mathrm{pt}}^{\star}$, and our results provably quantize the loss difference.

Theorem 7.3 (Joint training) *Consider training a single-layer linear attention model in solving multi-task ICL problem with dataset defined in Definition 7.2 and model construction as described in Assumption 7.1. Let $\boldsymbol{W}_{jt}^{\star}, \boldsymbol{P}_{jt}^{\star}$ (c.f. (7.12)) be the optimal solutions and $\mathcal{L}_{jt}^{\star}$ is the optimal joint training loss defined in Section 7.2.3. Additionally, let $\bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}, \tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}, \boldsymbol{M}_{\mu}$ follow the same definitions as in Theorem 7.2 and define $\bar{\boldsymbol{W}}_{jt}^{\star} = \boldsymbol{\Sigma}_{\boldsymbol{x}} \boldsymbol{W}_{jt}^{\star}$. Then the solution $(\boldsymbol{W}_{jt}^{\star}, \boldsymbol{P}_{jt}^{\star})$ and optimal loss $\mathcal{L}_{jt}^{\star}$ satisfy*

$$\bar{\boldsymbol{W}}_{jt}^{\star} = \bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} \left((n+1)\bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} + \mathtt{tr}\left(\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} \right) \boldsymbol{I} + \mathcal{O}(1) \right)^{-1},$$
$$\boldsymbol{P}_{jt}^{\star} = \boldsymbol{M}_{\mu} \left((\bar{\boldsymbol{W}}_{jt}^{\star})^{-1} - n\boldsymbol{I} \right) \boldsymbol{\Sigma}_{\boldsymbol{x}},$$
$$\mathcal{L}_{jt}^{\star} = \mathtt{tr}\left(\bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} \right) - n \cdot \mathtt{tr}\left(\bar{\boldsymbol{W}}_{jt}^{\star} \bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} \right).$$

Here, $\mathcal{O}(1)$ is a $d \times d$-sized matrix with entries being bounded by some constant value, regardless of n and d.

In Theorem 7.3, for clarity, we use $\mathcal{O}(1)$ to represent equality up to a matrix with entries bounded by a constant. The explicit form of $\bar{\boldsymbol{W}}_{\mathrm{jt}}^{\star}$ is provided in the Appendix.

The results of Theorem 7.2 and Theorem 7.3 highlight an important commonality of the optimal prompt: the optimal prompts $\bar{\boldsymbol{p}}_k$ capture the mean of corresponding tasks based on the attention weight $\boldsymbol{W}_{\text{pt}}^\star$ or $\boldsymbol{W}_{\text{jt}}^\star$. For a large context length n, $(\bar{\boldsymbol{p}}_k)_{\text{ft}}^\star$ and $(\bar{\boldsymbol{p}}_k)_{\text{jt}}^\star$ will be approximately $-n\boldsymbol{\Sigma}_{\boldsymbol{x}}\boldsymbol{\mu}_k$. Additionally, in a finite-dimensional setting where $d < \infty$, as $n \to \infty$, the solutions $\bar{\boldsymbol{W}}_{\text{pt}}^\star$ and $\bar{\boldsymbol{W}}_{\text{jt}}^\star$ converge to $\boldsymbol{I}/n$ and all the optimal losses $\mathcal{L}_{\text{pt}}^\star$, $\mathcal{L}_{\text{ft}}^\star$ and $\mathcal{L}_{\text{jt}}^\star$ approach 0. Therefore, the benefits of prompt tuning are more apparent for finite n.

Corollary 7.1 *Let $\mathcal{L}_{pt}^*$, $\mathcal{L}_{ft}^*$, and $\mathcal{L}_{jt}^*$ denote the optimal losses for plain training, fine-tuning, and joint training, as described in Theorems 1, 2 and 3, respectively. These losses satisfy:*

$$\mathcal{L}_{jt}^* \leq \mathcal{L}_{ft}^* \leq \mathcal{L}_{pt}^*. \tag{7.15}$$

The equalities hold if and only if $\bar{\boldsymbol{\Sigma}}_\beta = \tilde{\boldsymbol{\Sigma}}_\beta$ (c.f. (14)), which occurs when all task means $\boldsymbol{\mu}_k = \boldsymbol{0}$ for $k \in [K]$. Furthermore, the loss gaps satisfy the following:

1. *The loss gaps scale quadratically with task mean:*

$$\mathcal{L}_{pt}^* - \mathcal{L}_{ft}^* \sim \mathcal{O}\left(\frac{1}{n^2}\right) \|\Delta\|_F, \;\; \mathcal{L}_{ft}^* - \mathcal{L}_{jt}^* \sim \mathcal{O}\left(\frac{1}{n}\right) \|\Delta\|_F,$$

   where $\Delta := \tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} - \bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} = \boldsymbol{\Sigma}_x \sum_{k=1}^K \pi_k \boldsymbol{\mu}_k \boldsymbol{\mu}_k^\top$.

2. *The ratio between gaps is: $\frac{\mathcal{L}_{pt}^* - \mathcal{L}_{ft}^*}{\mathcal{L}_{ft}^* - \mathcal{L}_{jt}^*} \sim \mathcal{O}\left(\frac{1}{n}\right)$, indicating that fine-tuning provides most of the benefit in few-shot regimes (small n), while joint training benefits more for larger n.*

7.3.2 Covariance-mean decoupling

For a single-layer linear attention model under Assumption 7.1, where $\boldsymbol{W} \in \mathbb{R}^{d\times d}$ captures all the statistics, the plain training loss $\mathcal{L}_{\text{pt}}^\star$ is determined by the second-order moment of the task parameters $\boldsymbol{\beta}_k$:

$$\mathbb{E}[\boldsymbol{\beta}_k \boldsymbol{\beta}_k^\top] = \boldsymbol{\Sigma}_{\boldsymbol{\beta}_k} + \boldsymbol{\mu}_k \boldsymbol{\mu}_k^\top,$$

which represents the biased covariance of task k. Consequently, $\mathcal{L}_{\text{pt}}^\star$ can be viewed as a function of this biased variable, defined as $\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}$ (see Theorem 7.1), which affects both the terms in $\boldsymbol{W}_{\text{pt}}^\star$ and those that appear directly in $\mathcal{L}_{\text{pt}}^\star$ but outside of $\boldsymbol{W}_{\text{pt}}^\star$.

When all tasks have zero mean, i.e., $\mathbb{E}[\boldsymbol{\beta}_k] = \boldsymbol{\mu}_k = \boldsymbol{0}$ for all $k \in [K]$, leading to $\bar{\boldsymbol{\Sigma}}_{\boldsymbol{\mu}} = \boldsymbol{0}_{d\times d}$, the optimal losses for pretraining, fine-tuning, and joint training become identical:

$$\boldsymbol{\mu}_k = \boldsymbol{0},\ k \in [K] \Rightarrow \mathcal{L}_{\text{pt}}^\star = \mathcal{L}_{\text{ft}}^\star = \mathcal{L}_{\text{jt}}^\star. \tag{7.16}$$

In this case, the fine-tuned prompts remain as zero vectors, learning nothing during fine-tuning (as shown in Theorem 7.2 and Theorem 7.3). This implies that any differences in loss arise from the ability of the trainable prompts to **decouple** (i.e., remove task-mean related bias terms) from $\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}$ to obtain $\bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}$. Specifically, when all task means are zero, this decoupling is nullified, resulting in no difference in losses across the training methods.

When the task means $\mathbb{E}[\boldsymbol{\beta}_k] = \boldsymbol{\mu}_k$ are non-zero, the decoupling effect varies between different training settings. In joint training, the simultaneous optimization of prompts and attention weights enables decoupling in both $\mathcal{L}^\star_{\text{jt}}$ and $\boldsymbol{W}^\star_{\text{jt}}$, while in fine-tuning, only the biased terms in $\boldsymbol{W}^\star_{\text{ft}}$ are decoupled. As a result, joint training achieves greater decoupling than fine-tuning. According to Corollary 7.1, when a loss is more directly influenced by the biased covariance $\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}$, it tends to be higher, whereas reducing the influence of $\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}$ through decoupling generally leads to a lower loss.

These findings suggest that introducing additional task-specific trainable parameters into the single-layer linear attention model, combined with joint optimization, can effectively reduce the bias in mixed-task covariance, thereby improving performance.

7.4 Fully-decoupled Loss

In Section 7.3, we focus on cases where model parameters are constructed according to Assumption 7.1 and analyze the loss landscapes for plain training, finetuning, and joint training. However, our results indicate that with a shared head $\boldsymbol{a} := \boldsymbol{W}_v\boldsymbol{h}$, none of these methods fully decouple the mean and covariance. To address this, we introduce an alternative approach that allows each task to have its own specific linear prediction head $\boldsymbol{h}_k \in \mathbb{R}^{d+1}$. It is worth noting that using separate heads for different tasks is a common practice in the general multi-task learning literature [29, 219, 115]. Our results demonstrate that optimizing task-specific prompts, heads, and attention weights leads to a fully decoupled loss (Theorem 7.4).

Definition 7.4 (Task-specific heads) *Given K tasks, let $\{\boldsymbol{h}_k\}_{k=1}^K \subset \mathbb{R}^{d+1}$ represent their corresponding trainable linear prediction heads. Recalling the input sequence and prediction from (7.5) and (7.6), the prediction for task k returns*

$$\tilde{f}_{att}(\boldsymbol{Z}^{(k)}) = (\boldsymbol{z}^\top \boldsymbol{W}_q \boldsymbol{W}_k^\top (\boldsymbol{Z}^{(k)})^\top) \boldsymbol{M} \boldsymbol{Z}^{(k)} \boldsymbol{W}_v \boldsymbol{h}_k. \tag{7.17}$$

Recap ICL problem from Definition 7.2, $\boldsymbol{Z}^{(k)}$ from Definition 7.3 and loss function from (7.4), the ICL objective considering task-specific prompts and heads is:

$$\tilde{\mathcal{L}}^{\star}_{\text{att}} = \min_{(\boldsymbol{p}_k, \boldsymbol{h}_k)_{k=1}^K, \boldsymbol{W}_{k,q,v}} \mathcal{L}(\tilde{f}_{\text{att}}) \tag{7.18}$$
$$\text{where } \mathcal{L}(\tilde{f}_{\text{att}}) = \sum_{k=1}^{K} \pi_k \, \mathbb{E}_{\boldsymbol{Z}, y \sim \mathcal{D}_k}[(\tilde{f}_{\text{att}}(\boldsymbol{Z}^{(k)}) - y)^2].$$

Here, the search space for $\boldsymbol{p}_k, \boldsymbol{h}_k$ is $\mathbb{R}^{d+1}$ and the search space for $\boldsymbol{W}_{k,q,v}$ is $\mathbb{R}^{(d+1)\times(d+1)}$.

Next, given task k with mean $\boldsymbol{\mu}_k$, we introduce debiased preconditioned gradient descent (PGD) predictor as follows:

$$\tilde{f}_{\text{pgd}}(\boldsymbol{Z}^{(k)}) = \boldsymbol{x}^\top \boldsymbol{W} \boldsymbol{X}^\top (\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\mu}_k) + \boldsymbol{x}^\top \boldsymbol{\mu}_k.$$

Note that for any task k, we have $\mathbb{E}[y_i - \boldsymbol{x}_i^\top \boldsymbol{\mu}_k] = 0$, and therefore, we expect $\tilde{f}_{\text{pgd}}(\boldsymbol{Z}^{(k)}) - \boldsymbol{x}^\top \boldsymbol{\mu}_k$ to predict unbiased label $y - \boldsymbol{x}^\top \boldsymbol{\mu}_k$. The corresponding PGD objective is defined as:

$$\tilde{\mathcal{L}}^{\star}_{\text{pgd}} = \min_{\boldsymbol{W} \in \mathbb{R}^{d\times d}} \mathcal{L}(\tilde{f}_{\text{pgd}}) \tag{7.19}$$
$$\text{where } \mathcal{L}(\tilde{f}_{\text{pgd}}) := \sum_{k=1}^{K} \pi_k \, \mathbb{E}_{\boldsymbol{Z}, y \sim \mathcal{D}_k}[(\tilde{f}_{\text{pgd}}(\boldsymbol{Z}^{(k)}) - y)^2].$$

The following proposition establishes the equivalence between optimizing single layer linear attention (c.f. (7.18)) and one step of PGD predictor (c.f. (7.19)).

Proposition 7.1 *Consider the multi-task ICL data as described in Definition 7.2 and let* $\tilde{\mathcal{L}}^{\star}_{att}$ *and* $\tilde{\mathcal{L}}^{\star}_{\text{pgd}}$ *be the optimal linear attention and debiased preconditioned gradient descent losses as presented in* (7.18) *and* (7.19)*, respectively. Then,* $\tilde{\mathcal{L}}^{\star}_{att} = \tilde{\mathcal{L}}^{\star}_{\text{pgd}}$.

Considering the single-task and zero-mean setting, Proposition 7.1 aligns with the findings of previous work [3, 116]. Our results further confirm the necessity of both task-specific prompts and heads in effectively decoupling the influence of non-zero means from the data.

Then, based on the Proposition 7.1, we are able to analyze the optimization landscape of (7.18) via studying (7.19).

Theorem 7.4 *Consider the multi-task ICL problem with dataset defined in Definition 7.2. Let* $\boldsymbol{W}^{\star}_{\text{pgd}} := \arg\min_{\boldsymbol{W}} \mathcal{L}(\tilde{f}_{\text{pgd}})$ *following* (7.19)*. Define* $\bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}$ *in* (7.14) *and let* $\bar{\boldsymbol{W}}^{\star}_{\text{pgd}} = \boldsymbol{\Sigma}_{\boldsymbol{x}} \boldsymbol{W}^{\star}_{\text{pgd}}$*. Then the solution* $\bar{\boldsymbol{W}}^{\star}_{\text{pgd}}$ *and optimal loss* $\tilde{\mathcal{L}}^{\star}_{\text{pgd}}$ *(c.f.* (7.19)*) satisfy*

$$\bar{\boldsymbol{W}}_{\text{pgd}}^{\star} = \bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} \left((n+1)\bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} + \boldsymbol{tr}\left(\bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}\right) \boldsymbol{I} \right)^{-1},$$

$$\tilde{\mathcal{L}}_{\text{pgd}}^{\star} = \boldsymbol{tr}\left(\bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}\right) - n \cdot \boldsymbol{tr}\left(\bar{\boldsymbol{W}}_{\text{pgd}}^{\star} \bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}\right).$$

See Appendix for a proof.

Comparing Theorem 7.4 with Theorem 7.1, it is evident that $\tilde{\mathcal{L}}_{\text{pgd}}^{\star}$ represents a fully decoupled loss, demonstrating the clear benefit of adding task-specific heads. While Theorem 7.1 provides an upper bound $\mathcal{L}_{\text{pt}}^{\star}$ for the multi-task ICL loss, the proof of Theorem 7.4 establishes that $\tilde{\mathcal{L}}_{\text{pgd}}^{\star}$ serves as the lower bound for a multi-task ICL loss in a single-layer linear attention model.

CHAPTER 8

Benefits of Unlabeled Data in Multilayer Linear Attention

8.1 Introduction

The success of in-context learning (ICL) is also being extended to multimodal scenarios [223] as well as other modalities such as tabular data [77]. The push toward test-time scaling and long-context models [175, 69] has further boosted the benefits of ICL by allowing the model to ingest a large number of demonstrations. For instance, in "Many-shot in-context learning" paper, [2] demonstrate that pushing more examples into context window can substantially boost the accuracy. MAPLE [35] improves many-shot ICL by pseudo-labeling high-impact unlabeled examples and incorporating them into the prompt. The many-shot ICL setting naturally raises the question of when and how ICL can succeed with weaker supervision. As we can harness longer context models to boost predictive accuracy, we may indeed run out of high-quality demonstrations with verified answers/chain-of-thoughts and may want to utilize weaker data sources. This motivates our central question:

Q: *When and how can transformers learn in context from unlabeled data?*

We primarily investigate this question under a semisupervised ICL (SS-ICL) setting with Gaussian mixture models (GMMs). Formally, given a prompt containing a dataset of feature-label pairs $(\boldsymbol{x}_i, y_i)_{i=1}^n \in \mathbb{R}^d \times \mathbb{R}$ as demonstrations and a query feature $\boldsymbol{x}$ (see Eq. (8.3)), a model trained for ICL learns to predict the corresponding output y given prompt. For ICL with a supervised binary GMM model, we have $\boldsymbol{x}_i \sim \mathcal{N}(\boldsymbol{\mu}_{y_i}, \sigma^2 \boldsymbol{I})$ and $y_i \in \{-1, 1\}$, $i \in [n]$, and the component means $\boldsymbol{\mu}_{\pm 1}$ that parameterize the classification task are sampled from a prior task distribution. This prompt model is well studied under various fully-supervised settings [59, 196, 3, 4, 129, 38, 173] where each demonstration includes a clearly labeled output. In our SS-ICL setting, only m out of n total samples have correct labels ($m \leq n$)

either -1 or 1, and remaining labels are unknown and fed to the model as $y_i = 0$.

In this work, we provide a comprehensive theoretical and empirical study of attention models with varying depths when trained with SS-ICL. Our analysis reveals the importance of *depth*: Despite being able to implement the optimal fully-supervised estimator, single-layer linear attention completely fails to leverage unlabeled examples. In contrast, deeper or looped transformer architectures can emulate strong semi-supervision algorithms, approaching the performance of the Bayes-optimal classifier as depth increases. Informed by the importance of depth/looping, we also devise semisupervision strategies for tabular foundation models. Our specific contributions are:

- ◇ **Landscape of one-layer linear attention (§8.3):** We study the optimization landscape of single-layer linear attention for the SS-ICL problem under an isotropic task prior. We prove that the global minimum of the loss function returns the plug-in estimator (see Eq. (SPI)), i.e., $\hat{y} = \mathrm{sgn}(\boldsymbol{x}^\top \hat{\boldsymbol{\mu}})$ with $\hat{\boldsymbol{\mu}} = \boldsymbol{X}^\top \boldsymbol{y}$, where $\boldsymbol{X} \in \mathbb{R}^{n \times d}$ represents features and $\boldsymbol{y} \in \mathbb{R}^n$ denotes partially-observed labels (with missing entries set to zero) of the ICL demonstrations. This implies that 1-layer model learns Bayes-optimal classifier in the fully-supervised setting, but completely fails to make use of unlabeled data.

- ◇ **Depth is crucial but shallow can suffice (§8.4):** We show that multilayer linear attention can emulate semisupervised learners by implementing polynomial estimators of the form

$$\hat{\boldsymbol{\mu}} = \sum_{i=0}^{K} a_i (\boldsymbol{X}^\top \boldsymbol{X})^i \boldsymbol{X}^\top \boldsymbol{y}. \tag{8.1}$$

  Crucially, an L-layer (or looped) attention can express up to $K = O(3^L)$ powers, highlighting that logarithmic depth suffices to represent high-degree monomials. We provide characterizations of the set of expressible polynomials through different constructions (where each layer gets to update the features or labels of the previous layer). Corroborating these, experiments reveal that shallow models with $L \geq 2$ already achieve strong results and their performance can be approximately predicted through an eigenestimator combining $i = 0$ and ∞ (see (SSPI-k)).

- ◇ **What learner attention emulates?** In Section 8.4.3, we describe how each attention block can update the label estimates by emulating expectation-maximization (for linear attention) or belief propagation (for softmax attention). For instance (8.1) can be interpreted as the model implicitly conducting an *Expectation-Maximization* algo-

rithm: Starting with the supervised estimator $\hat{\boldsymbol{\mu}}_0 = \boldsymbol{X}^\top \boldsymbol{y}$, each term $(\boldsymbol{X}^\top \boldsymbol{X})^i \boldsymbol{X}^\top \boldsymbol{y}$ can be viewed as a sequence of pseudo-labeling (expectation) $\hat{\boldsymbol{y}}_i = \boldsymbol{X}\hat{\boldsymbol{\mu}}_{i-1}$ and training (maximization) $\hat{\boldsymbol{\mu}}_i = \boldsymbol{X}^\top \hat{\boldsymbol{y}}_i$ steps. Corroborating this, we show that softmax-attention and softmax-transformer models similarly benefit from increasing depth and can emulate semisupervised learners competitive with Bayes limit (see Fig. 8.2c).

- ◇ **Applications to Tabular FMs (§8.5):** Tabular foundation models such as TabPFN [77, 78] and TabICL [163] represent a suitable application of theory as they also model the ICL examples with a single token. To harness unlabeled examples, we propose a novel strategy that iteratively creates soft pseudo-labels by *explicitly looping the tabular FM* while controlling validation risk. Focusing on the few-shot learning setting where TabPFN-v2 [78] excels, we demonstrate that our approach can significantly improve predictive performance on various real-world datasets.

8.2 Preliminaries and Problem Setup

We study ICL in the setting of semi-supervised classification, where the in-context demonstrations are drawn from a binary Gaussian mixture model (GMM). We begin by introducing the following core notation: Denote the set $\{1, 2, \cdots, n\}$ as $[n]$ and use bold letters, such as $\boldsymbol{x}$ and $\boldsymbol{X}$, to represent vectors and matrices, respectively. Let $Q(\cdot)$ function return the right tail of the standard normal distribution. We use $\mathrm{sgn}(\cdot)$ denote the sign function which is defined as follows: $\mathrm{sgn}(x) = \begin{cases} 1, & x \geq 0 \\ -1, & x < 0 \end{cases}$.

8.2.1 Semi-supervised Data Model

Consider a d-dimensional semi-supervised binary GMM with n examples $(\boldsymbol{x}_i, y_i)_{i=1}^n$, where $\boldsymbol{x}_i \in \mathbb{R}^d$ denotes the feature vector and $y_i \in \{-1, 0, 1\}$ represents the corresponding observed label, with $y_i = 0$ indicating a missing label, and each label is revealed independently with probability $p \in [0, 1]$. Specifically, the data is generated as follows (for each $i \in [n]$):

$$\boldsymbol{x}_i = y_i^c \cdot \boldsymbol{\mu} + \boldsymbol{\xi}_i \quad , \quad y_i = \begin{cases} y_i^c, & \text{w.p.} \quad p \\ 0, & \text{w.p.} \quad 1-p \end{cases} \quad \text{and} \quad y_i^c = \begin{cases} 1, & \text{w.p.} \quad 1/2 \\ -1, & \text{w.p.} \quad 1/2 \end{cases}. \tag{8.2}$$

Here $\boldsymbol{\mu} \sim \mathrm{Unif}(\mathbb{S}^{d-1})$ denotes the task mean, which is sampled uniformly from the unit sphere, and $\boldsymbol{\xi}_i \sim \mathcal{N}(0, \sigma^2 \boldsymbol{I})$ is the random noise with $\sigma \geq 0$ being the noise level that controls the variability of $\boldsymbol{x}_i$ around its mean. y_i^c denotes the true class label that is uniform over

$\{-1, 1\}$. Observe that $p = 1$ corresponds to fully-supervised learning and $p = 0$ corresponds to fully-unsupervised learning.

8.2.2 In-context Learning and Linear Attention

We build on the setting of [59, 129, 214, 116] and construct the in-context prompts with examples drawn from the model (8.2) as follows.

Prompt Generation Given a task vector $\boldsymbol{\mu} \sim \text{Unif}(\mathbb{S}^{d-1})$, we sample $(n+1)$ in-context demonstrations $(\boldsymbol{x}_i, y_i)_{i=1}^{n+1}$ according to (8.2) and construct the prompt

$$\boldsymbol{Z} = \begin{bmatrix} \boldsymbol{x}_1 & \boldsymbol{x}_2 & \cdots & \boldsymbol{x}_n & \boldsymbol{x} \\ y_1 & y_2 & \cdots & y_n & 0 \end{bmatrix}^\top \in \mathbb{R}^{(n+1)\times(d+1)}. \tag{8.3}$$

We will investigate training a transformer such that given $\boldsymbol{Z}$ as prompt, it correctly predicts the label $y := y_{n+1}^c$ of the query $\boldsymbol{x} := \boldsymbol{x}_{n+1}$ through ICL.

Model Architecture Our work primarily focuses on training of linear attention models. Given any prompt $\boldsymbol{Z} \in \mathbb{R}^{(n+1)\times(d+1)}$, which can be treated as a sequence of $(d+1)$-dimensional tokens, the linear attention mechanism outputs

$$\text{att}(\boldsymbol{Z}; \mathcal{W}) = (\boldsymbol{Z}\boldsymbol{W}_q\boldsymbol{W}_k^\top\boldsymbol{Z}^\top)\boldsymbol{M}\boldsymbol{Z}\boldsymbol{W}_v \tag{8.4}$$

where $\mathcal{W} := \{\boldsymbol{W}_k, \boldsymbol{W}_q, \boldsymbol{W}_v \in \mathbb{R}^{(d+1)\times(d+1)}\}$ denotes the set of the key, query and value weight matrices. Therefore, given the prompt matrix $\boldsymbol{Z} \in \mathbb{R}^{(n+1)\times(d+1)}$ as input, the attention mechanism outputs a $(n+1)$-length sequence (i.e., $\text{att}(\boldsymbol{Z}; \mathcal{W}) \in \mathbb{R}^{(n+1)\times(d+1)}$). Note that the label for the query $\boldsymbol{x}$ is excluded from the prompt $\boldsymbol{Z}$. Similar to [3], we consider a training objective with a mask $\boldsymbol{M} = \begin{bmatrix} \boldsymbol{I}_n & 0 \\ 0 & 0 \end{bmatrix}$ to prevent input tokens from attending to the queries. To ensure that all in-context examples are treated equally and that the model remains invariant to their order/position, we do not apply a causal mask following [3]. In contrast, [118] explores the use of causal masking in multi-layer linear attention and analyzes its impact on the final prediction.

Building upon the single-layer linear attention mechanism of (8.4), we can extend our model to multiple layers to capture more complex patterns. Consider optimizing an L-layer linear attention model and let $\boldsymbol{Z}_\ell$ be the input of ℓth layer, $\ell \in [L]$. Additionally, let $\mathcal{W}_\ell := \{\boldsymbol{W}_{k\ell}, \boldsymbol{W}_{q\ell}, \boldsymbol{W}_{v\ell} \in \mathbb{R}^{(d+1)\times(d+1)}\}$ be the corresponding weight matrices of ℓth layer.

Then, recalling the attention mechanism (8.4), the input prompt of ℓth layer is defined by

$$\boldsymbol{Z}_\ell = \boldsymbol{Z}_{\ell-1} + \mathrm{att}(\boldsymbol{Z}_{\ell-1}; \mathcal{W}_{\ell-1}) \qquad \text{for} \qquad \ell = 2, \dots L, \tag{8.5}$$

and $\boldsymbol{Z}_1 = \boldsymbol{Z}$. We focus on the next-token prediction setting, where the model makes a prediction based on the final query token $[\boldsymbol{x}^\top \; 0]^\top$. Let $\boldsymbol{h} \in \mathbb{R}^{d+1}$ denote the linear prediction head. We define the output of the L-layer linear attention model at the last (query) token as

$$f_{\text{att-}L}(\boldsymbol{Z}) = \boldsymbol{h}^\top \mathrm{att}(\boldsymbol{Z}_L; \mathcal{W}_L)_{[n+1]}. \tag{8.6}$$

Recalling the sign function, the predicted label for $\boldsymbol{x}$ is given by $y_{\text{att-}L}(\boldsymbol{Z}) = \mathrm{sgn}(f_{\text{att-}L}(\boldsymbol{Z}))$.

Model Training With our attention-based architecture established, we now turn to the training procedure and evaluation metrics. Consider the ICL setting where each input prompt $\boldsymbol{Z}$ (cf. (8.3)) corresponds to a randomly sampled task vector $\boldsymbol{\mu} \sim \mathrm{Unif}(\mathbb{S}^{d-1})$ and let $\ell(\cdot) : \mathbb{R} \to \mathbb{R}$ be the loss function. Additionally, define the set of attention weights $\mathcal{W}^{(L)} := \cup_{\ell=1}^{L} \mathcal{W}_\ell \in (\mathbb{R}^{(d+1)\times(d+1)})^{3L}$. The objective of L-layer linear atention takes the following form:

$$\min_{\mathcal{W}^{(L)}, \boldsymbol{h}} \mathcal{L}_{\text{att-}L}(\mathcal{W}^{(L)}, \boldsymbol{h}) \qquad \text{where} \qquad \mathcal{L}_{\text{att-}L}(\mathcal{W}^{(L)}, \boldsymbol{h}) = \mathbb{E}\left[\ell(y, f_{\text{att-}L}(\boldsymbol{Z}))\right]. \tag{8.7}$$

Here, $y = y_{n+1}^c$ and the expectation subsumes the randomness of $\boldsymbol{\mu}$ and $(\boldsymbol{\xi}_i, y_i)_{i=1}^{n+1}$. The search space for $\mathcal{W}^{(L)}$ is $(\mathbb{R}^{(d+1)\times(d+1)})^{3L}$, and for $\boldsymbol{h}$ is $\mathbb{R}^{d+1}$.

8.3 Loss Landscape of One-layer Linear Attention under SS-ICL

Previous work [3, 116, 129] has shown that an optimized single-layer linear attention implements a form of preconditioned gradient descent over the linear in-context demonstrations provided within the prompt. However, to the best of our knowledge, prior studies have not addressed the semi-supervised setting, where some in-context labels are missing. In this section, we analyze the optimization behavior of single-layer linear attention under the semi-supervised binary GMM setting described in Section 8.2, and demonstrate that the single-layer model learns the optimal fully-supervised learner, but fails to utilize the unlabeled data.

We begin with the following optimal supervised label estimator under our problem setting.

Supervised Plug-in (SPI) Estimator The plug-in method is a classical approach for supervised classification problems, aiming to find a linear combination of features that separates different categories. Under our problem setting, it also serves as the asymptotically Bayes-optimal estimator given only labeled data [74, 45]. Consider the binary semi-supervised GMM problem described in (8.2) with dataset $(\boldsymbol{x}_i, y_i)_{i=1}^n$, and let $\mathcal{I} \subset [n]$ represent the indices of labeled samples, e.g., $y_i \neq 0$ for $i \in \mathcal{I}$. The SPI estimator returns the task mean

$$\hat{\boldsymbol{\mu}}_s = \frac{1}{|\mathcal{I}|} \sum_{i \in \mathcal{I}} y_i \boldsymbol{x}_i. \tag{SPI}$$

We next present the following theorem establishes that, under isotropic task prior, optimal single-layer linear attention is equivalent to the SPI estimation.

Theorem 8.1 *Let the prompt (cf.* (8.3)*) be generated as described in Section 8.2.2. Consider the objective (cf.* (8.7)*) with $L = 1$ and squared loss function $\ell(y, \hat{y}) = (y - \hat{y})^2$, and denote the optimal prediction as $y^\star_{att\text{-}1}(\boldsymbol{Z})$. Let $\hat{\boldsymbol{\mu}}_s$ represent the SPI estimator defined in* (SPI)*. Then, for any $\boldsymbol{Z}$ from* (8.3)*, we have*

$$y^\star_{att\text{-}1}(\boldsymbol{Z}) = sgn(\boldsymbol{x}^\top \hat{\boldsymbol{\mu}}_s). \tag{8.8}$$

Additionally, its classification error obeys

$$\begin{aligned} \mathbb{P}(y^\star_{att\text{-}1}(\boldsymbol{Z}) \neq y) &= \mathbb{E}_{g \sim \mathcal{N}(0,1), h \sim \mathcal{X}^2_{d-1}} \left[Q\left(\frac{1 + \epsilon_\sigma g}{\sigma \sqrt{(1 + \epsilon_\sigma g)^2 + \epsilon_\sigma^2 h}} \right) \right] \\ &\leq Q\left(\frac{1 - 10 d \epsilon_\sigma^2}{\sigma} \right) + e^{-d} + e^{-1/8\epsilon_\sigma^2} \end{aligned} \tag{8.9}$$

where we define $\epsilon_\sigma = \sigma / \sqrt{np}$ and $\mathcal{X}^2_d$ defines chi-squared distribution with d degrees of freedom.

The proof of Theorem 8.1 is deferred to Appendix G.1. Eq. (8.8) shows that one-layer linear attention model indeed implements the optimal supervised predictor, assuming access to np labeled examples. Therefore, the classification error corresponds exactly to that of the SPI estimator. The supervised classification problem has been extensively studied [13, 18, 134, 188, 32, 27, 199, 43], with most existing work focusing on a single classification task in asymptotic data or overparameterized regimes. In contrast, within the ICL framework considered in our setting, the task mean $\boldsymbol{\mu}$ is randomly sampled, and the classification error

is computed by averaging over random draws of $\boldsymbol{Z}$, y, and $\boldsymbol{\mu}$. Accordingly, in (8.9), we express the error in a simplified form as an expectation.

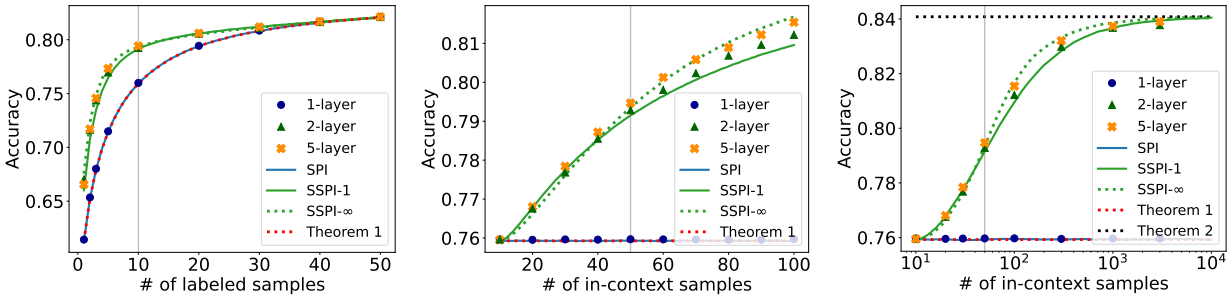
The experimental results in Figure 8.1 support Theorem 8.1, where dark blue circular markers represent the performance of the single-layer linear attention model, blue curves show the classification accuracy of the SPI estimator, and the red dotted curves depict the accuracy $1 - \mathbb{P}(y^{\star}_{\text{att-1}}(\boldsymbol{Z}) \neq y)$ as computed from (8.9). The alignments of these curves empirically validate Theorem 8.1. Implementation details and further discussion are provided in Section 8.5. Based on these results, we reach the following conclusion:

1-layer linear attention learns optimal supervised estimator but doesn't benefit from unlabeled data.

As shown in Figs 8.1b and 8.1c, when the number of labeled samples ($np = 10$) is fixed, increasing the number of unlabeled examples (even up to ~ 10000) has no effect on performance, as the dark blue markers remain at the same level.

At first glance, this may seem counterintuitive—while the data is unlabeled, it still contains information about the classification feature. For instance, the mean of the data points carries relevant information, and one might expect the model to extract and leverage this for better predictions. This expectation is particularly reasonable when a large amount of unlabeled data is available, as the sample covariance matrix approximates the population covariance, i.e., $\mathbb{E}[\boldsymbol{X}^\top \boldsymbol{X}/n] = \boldsymbol{\mu}\boldsymbol{\mu}^\top + \sigma^2 \boldsymbol{I}$ where $\boldsymbol{X} = [\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_n]^\top \in \mathbb{R}^{n \times d}$. The key insight into why single-layer attention fails to leverage unlabeled data lies in the expectation structure. In our isotropic GMM setting where $\boldsymbol{\mu} \sim \text{Unif}(\mathbb{S}^{d-1})$, the sample covariance matrix converges to $\mathbb{E}[\boldsymbol{X}^\top \boldsymbol{X}/n] = \mathbb{E}[\boldsymbol{\mu}\boldsymbol{\mu}^\top] + \sigma^2 \boldsymbol{I} = (1/d + \sigma^2)\boldsymbol{I}$, which contains no task-specific information. The expectation across multiple tasks loses the signal from $\boldsymbol{\mu}$. This explains why single-layer attention, operating in a meta-learning framework across many tasks rather than optimizing for a single fixed task, cannot extract useful information from unlabeled data.

In the following section, we study multi-layer linear attention and demonstrate that it has the ability to propagate $\boldsymbol{X}^\top \boldsymbol{X}$ into deeper layers, thereby enabling the model to utilize the unlabeled data.



(a) $n = 50$, $p \in (0, 1]$. (b) $np = 10$, $n \in \{10 \dots 100\}$. (c) $np = 10$, $n \in \{10 \dots 10000\}$.

Figure 8.1: Experimental results support our theoretical findings presented in Sections 8.3 and 8.4. In all three subfigures, blue, green, and orange markers represent the results of 1-, 2-, and 5-layer linear attention models, respectively. The SPI estimator (cf. (SPI)), SSPI-1, and SSPI-∞ (cf. (SSPI-k)) are shown as blue solid, green solid, and green dotted curves, respectively. The red dotted curves in all subfigures correspond to the single-layer/SPI results described in Eq. (8.9) of Theorem 8.1, while the black dotted line in Fig. 8.1c corresponds to Eq. (8.13) of Theorem 8.2. Additional details and discussion can be found in Sections 8.3, 8.4, and 8.5.

8.4 Multi-layer Attention and the Benefits of Depth

In this section, we explore how deeper attention models can effectively utilize the unlabeled data. Let

$$\boldsymbol{X} = \begin{bmatrix} \boldsymbol{x}_1 & \boldsymbol{x}_2 & \cdots & \boldsymbol{x}_n \end{bmatrix}^\top \in \mathbb{R}^{n \times d} \quad \text{and} \quad \boldsymbol{y} = \begin{bmatrix} y_1 & y_2 & \cdots & y_n \end{bmatrix}^\top \in \mathbb{R}^n. \tag{8.10}$$

8.4.1 L-layer Linear Attention can Implement Degree-$O(3^L)$ Polynomials in $X^\top X$

We first present the following propositions to show that multi-layer as well as looped linear attention can be expressed as a polynomial function of $\boldsymbol{X}^\top \boldsymbol{X}$. This structure allows the models to leverage unlabeled data to improve the estimation of the task mean $\boldsymbol{\mu}$.

Proposition 8.1 *Given an L-layer linear attention model described in Section 8.2.2 with input prompt $\boldsymbol{Z}$ defined in (8.3), one can construct the key, query, value weight matrices and the linear prediction head such that the model outputs (cf. (8.6))*

$$f_{\text{att-}L}(\boldsymbol{Z}) = \boldsymbol{x}^\top \boldsymbol{A} \boldsymbol{X}^\top \boldsymbol{y}. \tag{8.11}$$

Then, the following $\boldsymbol{A}$ matrices are achievable via label and feature updates:

- ***Label propagation:*** $\boldsymbol{A} = c\prod_{\ell=1}^{L-1}\left(\boldsymbol{I} + c_\ell \boldsymbol{X}^\top \boldsymbol{X}\right)$ *for arbitrary constants* $\{c, c_1, \cdots, c_{L-1}\}$*;*
- ***Feature propagation:*** $\boldsymbol{A} = c\left(\boldsymbol{X}^\top \boldsymbol{X}\right)^{3^{L-1}-1}$ *for an arbitrary constant* c*.*

Proposition 8.2 *Consider the same setting as in Proposition 8.1. There exists a single-layer linear attention model whose parameters can be constructed such that, when looped* L *times, its output reproduces that of* (8.11)*, with* $c_\ell \equiv c'$ *for some arbitrary constant* c'*.*

The proofs of Proposition 8.1 and 8.2 are deferred to Appendix G.2.1 and G.2.2. In the following, we provide further clarification on the label and feature propagation.

1. The final prediction of the label propagation process can be rewritten as

$$f_{\text{att-}L}(\boldsymbol{Z}) = c\boldsymbol{x}^\top \boldsymbol{X}^\top \boldsymbol{y}_L \quad \text{where} \quad \boldsymbol{y}_{\ell+1} = (\boldsymbol{I} + c_\ell \boldsymbol{X}\boldsymbol{X}^\top)\boldsymbol{y}_\ell, \quad \text{for} \quad \ell \in [L-1]$$

   with $\boldsymbol{y}_1 = \boldsymbol{y}$. Here, $\boldsymbol{y}_\ell$ can be interpreted as the soft pseudo-labels input to the ℓth layer, and each c_ℓ is parameterized by the attention mechanism in the corresponding layer. Although not exactly equivalent, the L-layer linear attention process shares similarities with the Expectation-Maximization (EM) algorithm for semi-supervised learning, with L iterations of pseudo-labeling and a different label update strategy.

2. In contrast, the feature propagation process yields the final prediction

$$f_{\text{att-}L}(\boldsymbol{Z}) = c\boldsymbol{x}_L^\top \boldsymbol{X}_L^\top \boldsymbol{y} \quad \text{where} \quad \begin{cases} \boldsymbol{X}_{\ell+1} = (\boldsymbol{X}_\ell \boldsymbol{X}_\ell^\top)\boldsymbol{X}_\ell \\ \boldsymbol{x}_{\ell+1} = (\boldsymbol{X}_\ell^\top \boldsymbol{X}_\ell)\boldsymbol{x}_\ell \end{cases} \quad \text{for} \quad \ell \in [L-1]$$

   with $\boldsymbol{X}_1 = \boldsymbol{X}$ and $\boldsymbol{x}_1 = \boldsymbol{x}$. Here, $(\boldsymbol{X}_\ell, \boldsymbol{x}_\ell)$ can be viewed as the input features at the ℓth layer, encoding exponentially higher-order powers of $\boldsymbol{X}^\top \boldsymbol{X}$. This result highlights that a linear attention model requires only $O(\log K)$ layers to represent polynomial functions of degree K.

Our construction for *label propagation* is inherently related to the *gradient descent* emulation capability of linear attention [3]. However, the *feature propagation* construction is fundamentally different and underscores the transformer's capability to implement rapid power iteration over the empirical covariance $\boldsymbol{X}^\top \boldsymbol{X}$. In the above constructions, each attention block with residual connections updates features or labels using one parameter, namely mappings of the form $\boldsymbol{X} \to \boldsymbol{X} + \alpha \boldsymbol{X}\boldsymbol{X}^\top \boldsymbol{X}$ or $\boldsymbol{y} \to \boldsymbol{y} + \beta \boldsymbol{X}\boldsymbol{X}^\top \boldsymbol{y}$. The lemma below shows that, even if the multilayer model can express polynomials of $\boldsymbol{X}^\top \boldsymbol{X}$ with exponential degrees in depth, the expressible manifold of polynomials has dimensionality linear in depth.

Lemma 8.1 (Label + Feature Propagation) *For an L-layer linear attention model, the resulting eventual prediction corresponds to the matrix $\boldsymbol{A}$ in Proposition 8.1 of the form*

$$\boldsymbol{A} = \sum_{\ell=0}^{(3^L-3)/2} a_\ell (\boldsymbol{X}^\top \boldsymbol{X})^\ell. \tag{8.12}$$

The coefficients $\boldsymbol{a} := [a_0 \;\; a_1 \;\; \cdots \;\; a_{(3^L-3)/2}]^\top$ lie on a manifold of dimension at most $2L$ as $\boldsymbol{a}$ can be expressed as $\boldsymbol{a} = g(\boldsymbol{c})$ for some smooth function $g : \mathbb{R}^{2L} \to \mathbb{R}^{(3^L-3)/2}$ with $\boldsymbol{c}$ representing the parameters of individual layers.

8.4.2 Which Semi-supervised Algorithm Does Multi-layer Attention Approximate?

Recall the SPI estimator $\hat{\boldsymbol{\mu}}_s$ from (SPI), and that $\boldsymbol{y}$ denotes the visible labels defined in Section 8.2.1 and (8.10). We have $\hat{\boldsymbol{\mu}}_s = \frac{1}{|\mathcal{I}|}\boldsymbol{X}^\top \boldsymbol{y}$. Motivated by Proposition 8.1 that multi-layer linear attention can implement higher-degree polynomials of $\boldsymbol{X}^\top \boldsymbol{X}$, we introduce the following SSPI estimator, which makes predictions based on the supervised estimate $\hat{\boldsymbol{\mu}}_s$ combined with higher-order debiased term of the form $(\boldsymbol{X}^\top \boldsymbol{X}/n - \sigma^2 \boldsymbol{I})^k$.

Semisupervised Plug-in (SSPI) Estimator Observe that the feature covariance satisfies $\mathbb{E}[\boldsymbol{X}^\top \boldsymbol{X}]/n = \boldsymbol{\mu}\boldsymbol{\mu}^\top + \sigma^2 \boldsymbol{I}$, and the top eigenvector of the centered covariance matrix $(\boldsymbol{X}^\top \boldsymbol{X}/n - \sigma^2 \boldsymbol{I})$ asymptotically aligns with either $\boldsymbol{\mu}$ or $-\boldsymbol{\mu}$. Therefore, with a substantial amount of unlabeled data, we propose the semisupervised plug-in (SSPI) estimator as follows:

$$\hat{\boldsymbol{\mu}}_{ss\text{-}k} = \alpha \hat{\boldsymbol{\mu}}_s + (1-\alpha)(\boldsymbol{X}^\top \boldsymbol{X}/n - \sigma^2 \boldsymbol{I})^k \hat{\boldsymbol{\mu}}_s \tag{SSPI-k}$$

where $\hat{\boldsymbol{\mu}}_s$ is the SPI estimator (cf. (SPI)), and $\alpha \in [0,1]$ controls the trade-off between the fully-supervised and semi-supervised estimators. The optimal choice of α depends on the problem parameters n, d and p. Note that as $k \to \infty$, the term $(\boldsymbol{X}^\top \boldsymbol{X}/n - \sigma^2 \boldsymbol{I})^k$ converges (up to scaling) to a rank-one projection onto the top eigenvector of the debiased covariance matrix, effectively serving as an estimator for $\boldsymbol{\mu}$ (up to sign).

In Figure 8.1, we present the prediction accuracies of 2-layer and 5-layer linear attention models, shown by green and orange markers, respectively. We also evaluate the SSPI algorithm with varying k values, where the green solid curve corresponds to SSPI-1, and the green dotted represents SSPI-∞, both using their respective optimal choices of α. Details on selecting the optimal α values are provided in Section 8.5 and illustrated in Figure 8.2. The

results reveal a close alignment between multi-layer linear attention and SSPI estimators. Notably, the 2-layer model outperforms SSPI-1, due to its ability to implement higher-degree polynomials of $\boldsymbol{X}^\top \boldsymbol{X}$ (cf. Proposition 8.1 and Equation (8.12)). When the sample size is sufficiently large (e.g., $n > 50$ in Figure 8.1b), the top eigenvector provides a more accurate estimate of the task mean, enabling SSPI-∞ to achieve higher accuracy. Furthermore, since the 5-layer model is capable of representing higher-order functions than the 2-layer model, it can better estimate the top eigenvector, resulting in performance that closely matches that of SSPI-∞.

In the following, we analyze the optimal classifier of the form $\operatorname{sgn}(\boldsymbol{x}^\top \boldsymbol{A}\hat{\boldsymbol{\mu}}_s)$ for a GMM, and provide insights into its behavior in the asymptotic regime as $n \to \infty$.

Theorem 8.2 *Consider a binary GMM defined in Section 8.2.1 and suppose that $(\boldsymbol{x}_i, y_i)_{i=1}^{n+1}$ is generated using a fixed $\boldsymbol{\mu}$ following* (8.2)*. Given matrix $\boldsymbol{A} \in \mathbb{R}^{d\times d}$, define prediction*

$$\hat{y}_{\boldsymbol{A}} = sgn(\boldsymbol{x}^\top \boldsymbol{A}\hat{\boldsymbol{\mu}}_s).$$

where $\hat{\boldsymbol{\mu}}_s$ is the SPI estimator defined in (SPI)*. Let $\mathcal{A}^\star := \min_{\boldsymbol{A}\in\mathbb{R}^{d\times d}} \mathbb{P}(\hat{y}_{\boldsymbol{A}} \neq y)$ be its optimal solution set. Then, $\boldsymbol{\mu}\boldsymbol{\mu}^\top \in \mathcal{A}^\star$. Additionally, it obeys*

$$\mathbb{P}(\hat{y}_{\boldsymbol{\mu}\boldsymbol{\mu}^\top} \neq y) = \underbrace{Q(1/\sigma)}_{Bayes\ error} + Q(\sqrt{np}/\sigma) - 2Q(1/\sigma)Q(\sqrt{np}/\sigma). \tag{8.13}$$

Note that, $\mathbb{P}(\hat{y}_{\boldsymbol{\mu}\boldsymbol{\mu}^\top} \neq y)$ depends on np and σ only, regardless of $\boldsymbol{\mu}$ and d.

Theorem 8.3 *Let the prompt $\boldsymbol{Z}$ be generated as described in Section 8.2.2, and consider an L-layer linear attention model with $L \geq 2$ and $n = \infty$. Additionally, let $\hat{\boldsymbol{\mu}}_s$ be the SPI estimator defined in* (SPI)*. There exist model constructions such that for any $\boldsymbol{Z}$ following* (8.3)*, its prediction satisfies*

$$y_{att\text{-}L}(\boldsymbol{Z}) = sgn(\boldsymbol{x}^\top \boldsymbol{\mu}\boldsymbol{\mu}^\top \hat{\boldsymbol{\mu}}_s).$$

The proof follows directly from Proposition 8.1 (label propagation), which shows that multi-layer linear attention can output $\boldsymbol{x}^\top(\boldsymbol{X}^\top\boldsymbol{X}/n - \sigma^2\boldsymbol{I})\hat{\boldsymbol{\mu}}_s$. As $n \to \infty$, the empirical covariance converges to its expectation, i.e., $\boldsymbol{X}^\top\boldsymbol{X}/n - \sigma^2\boldsymbol{I} \to \boldsymbol{\mu}\boldsymbol{\mu}^\top$. The results in Figure 8.1c validate Theorem 8.3, showing that as n becomes large enough (i.e., $n = 10000$), the predictions from both 2-layer and 5-layer linear attention models, as well as the SSPI-1 and SSPI-∞ estimators, closely align with the classification error characterized in Theorem 8.2, depicted by the black dotted line.

Theorem 8.3 establishes that, with infinitely many unlabeled samples, an L-layer linear attention model (for $L \geq 2$) can implement the predictor characterized in Theorem 8.2 using the optimal choice of $\boldsymbol{A}$, thereby achieving the classification error specified in (8.13). In the following, we shift to the non-asymptotic setting where n is finite and analyze the model's performance in this regime.

Theorem 8.4 *Let the prompt $\boldsymbol{Z}$ be generated as described in Section 8.2.2. Consider an L-layer linear attention model with $L \geq 2$ and denote its optimal prediction as $y^\star_{att\text{-}L}(\boldsymbol{Z})$. Additionally, let $\hat{\boldsymbol{\mu}}_s$ be the SPI estimator defined in* (SPI)*. Suppose that the number of labeled samples satisfies $np \geq 8d\sigma^2$ and $n > O(d)$ is sufficiently large. Then, there exists a universal constant $C > 0$ such that the classification error satisfies*

$$\mathbb{P}(y^\star_{att\text{-}L}(\boldsymbol{Z}) \neq y) \leq Q\left(\frac{1 - C\sqrt{d/n}}{\sigma}\right) + e^{-d}.$$

The proof is deferred to Appendix G.2.5. Note that when $n \gg d$, the classification error approaches the Bayes error, i.e., $\mathbb{P}(y^\star_{\text{att-}L}(\boldsymbol{Z}) \neq y) \approx Q(1/\sigma)$.

8.4.3 Multi-layer Attention as Expectation Maximization and Belief Propagation

In Section 8.4.1, we discussed how multi-layer linear attention can express polynomial functions of $\boldsymbol{X}^\top \boldsymbol{X}$. Here, we further explore the connection between multi-layer attention and the Expectation Maximization (EM) algorithm for semi-supervised learning. Beyond linear attention, we also highlight key differences between linear and softmax-based attention mechanisms, particularly in how they implement labeling strategies analogous to those in the EM algorithm.

Consider the following construction of the ℓ-th layer attention weights:

$$\boldsymbol{W}_q = \boldsymbol{W}_k = \begin{bmatrix} \boldsymbol{I}_d & 0 \\ 0 & 0 \end{bmatrix} \quad \text{and} \quad \boldsymbol{W}_v = \begin{bmatrix} 0 & 0 \\ 0 & c_\ell \end{bmatrix}.$$

We examine both linear and softmax attention mechanisms. Let $\mathbb{S}(\cdot)$ denotes the softmax operation that applies on the rows of a matrix. With this, the data update defined in (8.5) becomes:

$$\begin{aligned} &\text{Linear attention:} && \boldsymbol{y}_{\ell+1} = \boldsymbol{y}_\ell + c_\ell \boldsymbol{X}\boldsymbol{X}^\top \boldsymbol{y}_\ell \\ &\text{Softmax attention:} && \boldsymbol{y}_{\ell+1} = \boldsymbol{y}_\ell + c_\ell \mathbb{S}(\boldsymbol{X}\boldsymbol{X}^\top)\boldsymbol{y}_\ell \end{aligned}$$

In the case of linear attention, given the pseudo-labels $\boldsymbol{y}_\ell = [y_1^\ell, y_2^\ell, \ldots, y_n^\ell]$ at layer ℓ, the model estimates the task mean using the SPI algorithm (cf. (SPI)) as $\hat{\boldsymbol{\mu}}_\ell = \boldsymbol{X}^\top \boldsymbol{y}_\ell$. The attention then updates each pseudo-label through the residual rule:

$$y_i^\ell \to y_i^\ell + c_\ell \boldsymbol{x}_i^\top \hat{\boldsymbol{\mu}}_\ell,$$

where c_ℓ is a layer-specific coefficient. Owing to the linearity of this mechanism, the resulting pseudo-labeling strategy aligns with a linear EM-style update.

In contrast, softmax attention computes pairwise similarities via the softmax of dot products. Define $s_{ij} = \frac{e^{\boldsymbol{x}_i^\top \boldsymbol{x}_j}}{\sum_{j \le n} e^{\boldsymbol{x}_i^\top \boldsymbol{x}_j}}$, then each pseudo-label is updated via:

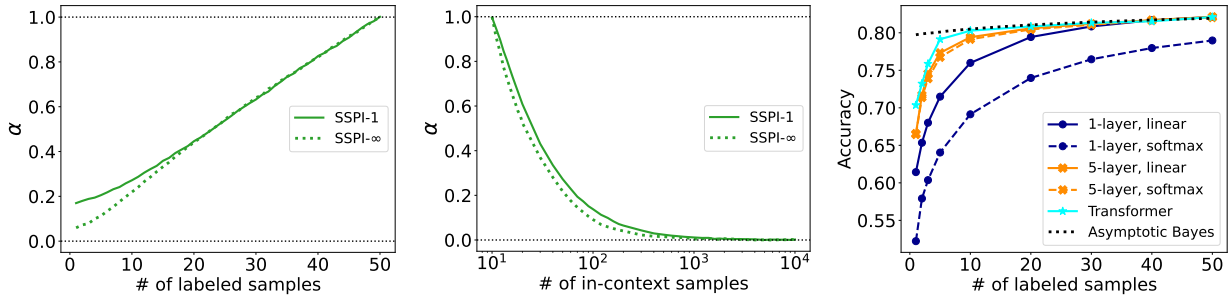
$$y_i^\ell \to y_i^\ell + c_\ell \sum_{j \le n} s_{ij} y_j^\ell.$$

This update is a nonlinear, similarity-weighted pseudo-labeling strategy which can also be viewed as *belief propagation*. The nonlinear nature highlights a key distinction between softmax and linear attention in how they emulate EM–like updates ($s_{ij} = \boldsymbol{x}_i^\top \boldsymbol{x}_j$ for linear attention).

8.5 Experiments

In Sections 8.3 and 8.4, we introduced Figure 8.1 and demonstrated its consistency with our theoretical results. In this section, we describe the experimental setup and implementation details. Additionally, we present further empirical findings to investigate additional questions of interest in Section 8.5.1. Motivated by Proposition 8.2, which suggests that looping can help leverage unlabeled data, Section 8.5.2 introduces an algorithm based on the TabPFN, showing how it can enhance prediction performance by incorporating a small amount of unlabeled data and iterative pseudo-labeling through model looping.

Experimental Setup Following Section 8.2, set $d = 10$ and noise level $\sigma = 1$. All models are trained using Adam optimizer with a learning rate of 10^{-3} for 40,000 epochs, with a batch size of 512. We use logistic loss in our experiments. Since our study focuses on the optimization landscape and model expressivity, and experiments are implemented via gradient descent, we repeat 10 trainings from random initialization and results are presented as the maximal test accuracy among those 10 trails.



(a) $n = 50$, $p \in (0, 1]$. (b) $np = 10$, $n \in \{10 \dots 10000\}$. (c) $n = 50$, $p \in (0, 1]$.

Figure 8.2: Additional experimental results. (a)&(b): Analysis of the optimal α values for the SSPI estimator (cf. (SSPI-k)) under varying (n, p, k). Green solid and dotted curves represent optimal α values for SSPI-1 and SSPI-∞, respectively. The SSPI results shown in Figure 8.1 use the corresponding α values from Figs. 8.2a and 8.2b. (c): Comparison of different model architectures for the SS-ICL problem. Dark blue and orange curves show results for 1-layer and 5-layer attention models, with solid and dashed curves representing linear and softmax attention, respectively. Cyan curves correspond to 5-layer Transformers. The black dotted curve shows the asymptotic Bayes-optimal error (cf. [103]). Results suggest the performance ordering: Transformer $>$ linear attention $>$ softmax attention. Further details are provided in Section 8.5.

8.5.1 Additional Observations

Exploration of Optimal α Values In Section 8.4, we introduced the SSPI-k estimator (cf. (SSPI-k)), but did not discuss the choice of the mixing parameter α, which plays a crucial role in balancing the contribution of the supervised estimator $\hat{\boldsymbol{\mu}}_s$. Specifically, α controls how much weight is given to the purely supervised signal. In the fully supervised case, the optimal choice is $\alpha = 1$, as $\hat{\boldsymbol{\mu}}_s$ corresponds to the optimal estimator.

In Figures 8.2a and 8.2b, we empirically examine the optimal values of α. Given $\boldsymbol{\mu} \sim \text{Unif}(\mathbb{S}^{d-1})$, we define the optimal α as the minimizer of the following cosine similarity-based objective:

$$\alpha^\star := \min_{\alpha \in [0,1]} \mathcal{L}(\alpha) \quad \text{where} \quad \mathcal{L}(\alpha) = 1 - \mathbb{E}[\texttt{cosine\_similarity}(\boldsymbol{\mu}_{ss\text{-}k}, \boldsymbol{\mu})].$$

For each setting, we optimize α using the Adam optimizer for 10,000 epochs with a batch size of 128 and a learning rate of 0.01. The results are shown in Figs 8.2a and 8.2b.

In Figure 8.2a, for both SSPI-1 and SSPI-∞, the optimal α starts near zero when the number of labeled examples is small, reflecting the limited utility of $\hat{\boldsymbol{\mu}}_s$ in low-supervision regimes. As the number of labeled samples increases, α grows approximately linearly and approaches 1 when the problem becomes fully supervised. In Figure 8.2b, when $n = 10$ and

$p = 1$ (i.e., all examples are labeled), the optimal α begins at 1. As n increases and the fraction of unlabeled data grows, α decreases significantly. This trend indicates that as the volume of unlabeled data increases, the SSPI estimator adaptively reduces reliance on the supervised component $\hat{\boldsymbol{\mu}}_s$ and increases reliance on the semi-supervised component, which leverages the structure of the unlabeled data through $\boldsymbol{X}^\top \boldsymbol{X}$.

Comparison Across Different Model Architectures Beyond linear attention, we investigate additional model architectures under our SS-ICL setting. The comparison results are presented in Fig. 8.2c. The softmax attention model uses the same structure described in Section 8.2.2, with the only difference being the addition of a softmax operation in Eq. (8.4). The Transformer model introduces further nonlinearity and capacity by incorporating multilayer perceptrons (MLPs) and layer normalization. The Transformer experiments are conducted with 5-layer models.

When comparing weaker models—such as 1-layer linear (dark blue solid) and softmax (dark blue dashed) attention—we observe that softmax attention consistently underperforms linear attention. Notably, softmax attention fails to match the performance of the optimal supervised estimator, even when all labels are observed (i.e., when the number of labeled samples equals $n = 50$). Furthermore, increasing the depth of softmax attention (orange dashed curve for 5-layer softmax) still does not surpass the performance of 5-layer linear attention (orange solid curve). Among all architectures, the Transformer achieves the best performance due to its increased model capacity and expressiveness. Compared with Fig. 8.1a, where the orange and dark blue markers (linear attention) are identical, the Transformer significantly improves accuracy. This improvement highlights that SSPI, while effective, is not the optimal semi-supervised estimator. Although our semi-supervised setting assumes isotropic data, the characterization of its optimal algorithm remains an open and foundational problem for future exploration. In the figure, we also include the asymptotic Bayes-optimal curve (black dotted; derived from [103]) . As the number of samples increases, the results from linear attention, softmax attention, and Transformer all converge toward this optimal curve. We attribute the initial performance gap, particularly at low values along x-axis (e.g., $np = 1$), to the scarcity of labeled data.

8.5.2 Tabular Experiments

To further investigate how model looping (Proposition 8.2) can improve label prediction, we propose the LoopTabFM algorithm that addresses unlabeled data by iteratively assigning pseudo-labels, with its details outlined in Algorithm 1. Suppose that we are given labeled $\mathcal{D}_{\text{lab}}$ and unlabeled $\mathcal{D}_{\text{unlab}}$ datasets. The overall workflow of the algorithm proceeds as follows:

Algorithm 1 LoopTabFM: Looping Tabular FM with Soft Pseudo-labels

Require: Dataset $\mathcal{D}_{\text{lab}}, \mathcal{D}_{\text{unlab}}$, looping iterations K

procedure LOOPING($\mathcal{D}_{\text{lab}}, \mathcal{D}_{\text{unlab}}, K$)
  $\texttt{FM}_0 \leftarrow$ TabPFN-v2($\mathcal{D}_{\text{lab}}$) ▷ $\texttt{FM}_k$ corresponds to model of Loop-k.
  $\mathcal{D}_{\text{unlab}} \leftarrow \texttt{FM}_0(\mathcal{D}_{\text{unlab}})$ ▷ Assign pseudo labels via $\hat{y}^{\text{soft}} \leftarrow \texttt{FM}_0(\boldsymbol{x} \in \mathcal{D}_{\text{unlab}})$.
  $\texttt{FM}_{\text{best}} \leftarrow \texttt{FM}_0$
  $\mathcal{R}_{\text{val}} = \texttt{Val\_Risk}(\mathcal{D}_{\text{unlab}})$
  for Looping iteration $k = 1, \ldots, K$ **do**
    $\texttt{FM}_k \leftarrow$ TabPFN-v2($\mathcal{D}_{\text{lab}} \cup \mathcal{D}_{\text{unlab}}$)
    $\mathcal{D}_{\text{unlab}} \leftarrow \texttt{FM}_k(\mathcal{D}_{\text{unlab}})$ ▷ Update pseudo labels via $\hat{y}^{\text{soft}} \leftarrow \texttt{FM}_k(\boldsymbol{x} \in \mathcal{D}_{\text{unlab}})$.
    if $\texttt{Val\_Risk}(\mathcal{D}_{\text{unlab}}) < \mathcal{R}_{\text{val}}$ **then**
      $\texttt{FM}_{\text{best}} \leftarrow \texttt{FM}_k$
      $\mathcal{R}_{\text{val}} = \texttt{Val\_Risk}(\mathcal{D}_{\text{unlab}})$
    end if
  end for
  return $\texttt{FM}_{\text{best}}$
end procedure
procedure VAL_RISK($\mathcal{D}_{\text{unlab}}$)
  return $\frac{1}{|\mathcal{D}_{\text{unlab}}|} \sum_i \min(|\hat{y}_i^{\text{soft}} - 1|,\ |\hat{y}_i^{\text{soft}} + 1|)$
  ▷ $\hat{y}^{\text{soft}}$ corresponds to the assigned soft label for feature in $\mathcal{D}_{\text{unlab}}$.
end procedure

1. **Base Model:** Perform ICL using TabPFN on the labeled dataset $\mathcal{D}_{\text{lab}}$ and treat the resulting model as the base model (Loop-0). The corresponding test accuracies are reported in Table 8.1.

2. **Pseudo-Label Assignment:** Using the current model (e.g., Loop-k) to generate predictions for the unlabeled data $\mathcal{D}_{\text{unlab}}$. Assign soft pseudo-labels based on these predictions. Note that the model outputs are scalars (i.e., elements of $\mathbb{R}$) and can be interpreted as soft labels.

3. **Model Update:** Construct a new prompt by combining the labeled examples with their true labels and the unlabeled examples with their assigned soft pseudo-labels. Perform ICL using TabPFN on this combined prompt to obtain an updated model (Loop-$(k+1)$). Repeat this process from Step 2 until the maximum number of looping iterations is reached.

⋆ **Model Validation:** To improve the stability of the looping process, we introduce an additional validation step and retain the model with the lowest validation risk as the final (best) model. Specifically, after assigning soft pseudo-labels to the unlabeled data, i.e., $\mathcal{D}_{\text{unlab}} = \{(\boldsymbol{x}_i, \hat{y}_i^{\text{soft}})_{i=1}^n\}$, we compute the validation risk over these pseudo-labeled

Table 8.1: Comparison of test accuracy (%) between the baseline (Loop-0) and LoopTabFM (Algorithm 1) after 1 to 5 iterations using TabPFN-v2. Each result is averaged over 100 random trials. The highest test accuracy for each dataset is highlighted in bold. The final column reports the relative improvement (%) of Loop-5 over the baseline, computed as (Loop-5 − Loop-0)/Loop-0×100%. Positive signs indicate a performance improvement over the baseline, while negative signs indicate a performance drop.

OpenML ID	# of features	# of samples	Class imbalance	Loop-0	Loop-1	Loop-2	Loop-3	Loop-4	Loop-5	Rel.Imp. (%)
3	36	3196	1.09	58.62	58.63	58.45	58.69	**59.00**	58.97	0.60 (+)
31	20	1000	2.33	**66.18**	65.95	66.05	65.58	65.52	65.07	1.68 (−)
1049	37	1458	7.19	72.00	75.62	79.48	80.31	**81.49**	81.40	13.06 (+)
1067	21	2109	5.47	73.12	76.59	77.94	77.92	78.57	**78.60**	7.50 (+)
1464	4	748	3.20	60.46	63.96	70.20	71.29	**72.26**	72.18	19.38 (+)
1487	72	2534	14.84	82.54	87.67	88.57	88.27	**89.85**	89.56	8.51 (+)
1489	5	5404	2.41	66.40	67.62	**68.30**	68.14	68.21	68.18	2.69 (+)
1494	41	1055	1.96	62.24	63.05	64.62	65.94	**66.07**	66.05	6.12 (+)
40701	20	5000	6.07	66.45	70.65	75.99	**78.18**	78.00	77.70	16.93 (+)
40900	36	5100	67	**98.53**	98.41	98.39	98.39	98.27	98.26	0.28 (−)
40981	14	690	1.25	73.56	74.41	74.67	**74.99**	74.93	74.94	1.88 (+)
40983	5	4839	17.54	79.71	85.04	89.36	**92.94**	92.90	92.75	16.35 (+)
41143	144	2984	1	64.64	64.80	65.06	65.17	**65.29**	65.13	0.76 (+)
41144	259	3140	1.01	50.70	50.63	50.68	50.67	50.71	**50.77**	0.14 (+)
41145	308	5832	1	56.16	**56.28**	56.21	56.24	56.19	56.22	0.12 (+)
41146	20	5124	1	71.26	73.90	75.39	75.84	76.02	**77.07**	8.51 (+)
41156	48	4147	3.03	67.74	69.78	70.64	**71.82**	71.72	71.74	5.90 (+)
Average				68.84	70.76	72.35	72.96	**73.24**	73.21	6.35 (+)

examples as follows:

$$\texttt{Val\_Risk}(\mathcal{D}_{\text{unlab}}) = \frac{1}{n} \sum_{i \in [n]} \min\left(\left| \hat{y}_i^{\text{soft}} - 1 \right|, \left| \hat{y}_i^{\text{soft}} + 1 \right| \right),$$

which penalizes predictions that deviate from confident binary labels ± 1.

We evaluated the effectiveness of our proposed looping strategy by iteratively applying TabPFN-v2 on real-world binary classification benchmarks used in [78]. The results are summarized in Table 8.1, where each entry represents an average over 100 random splits of the dataset, with 80% of the data used as the test set in each split.

For each experiment, we randomly sample 10 labeled and 10 unlabeled examples, ensuring that the labeled set includes at least one example from each class. As a baseline (Loop-0), we apply TabPFN-v2 using only the labeled data. The corresponding test accuracies are reported in the "Loop-0" column of Table 8.1. We compare this to models updated through up to $k \leq 5$ iterations of pseudo-label update, with results shown in the "Loop-k" columns. The final column reports the relative improvement (Rel. Imp.) over the baseline. Our results demonstrate that the looping strategy can significantly improve test accuracy. For instance, on OpenML datasets 1049, 1464, 40701, and 40983, accuracy improves by more than 10% over the baseline using only 10 additional unlabeled samples. The last row of the

table reports average performance across datasets, revealing that the majority of performance gains occur in the first two iterations. This observation aligns with our synthetic experiments using multi-layer models (Figure 8.1), where the improvement from 1-layer to 2-layer is substantially greater than the improvement from 2-layer to 5-layer. These findings highlight that explicitly looping the tabular foundation model to iteratively refine soft pseudo-labels of unlabeled data using only a few iterations can substantially enhance performance.

As shown and discussed, our LoopTabFM algorithm enhances model performance. However, this improvement is not consistent across all datasets. For example, performance drops on the OpenML datasets with IDs 31 and 40900. This may be attributed to factors such as noise levels in the raw data, class imbalance, or other dataset-specific characteristics. In contrast to our synthetic experimental setting, where the model is pretrained in a meta-learning fashion on the distribution of the given dataset, TabPFN is used as a general-purpose pretrained foundation model and applied directly to target datasets in a single-shot inference setting. Prior work [211] has also shown that TabPFN can be sensitive to input length, which may further affect performance consistency. Despite these limitations, our experiments with TabPFN offer an initial insight into how unlabeled data and iterative looping can be leveraged to improve predictive performance. These findings suggest promising future directions, such as designing data-aware looping algorithms that adapt to dataset-specific properties.

CHAPTER 9

Generalization in In-context Learning

9.1 Introduction

In-context learning (ICL) shows the ability of a Transformer (TF) to provide a prediction on an example given a few (input, output) examples within a prompt. As depicted in Figure 9.1, for NLP, the examples may correspond to pairs of (question, answer)'s or translations. Recent works [59, 100] demonstrate that ICL can also be used to infer general functional relationships. For instance, [77] aim to solve supervised learning problems where we feed an entire training dataset $(\boldsymbol{x}_i, f(\boldsymbol{x}_i))_{i=1}^{n-1}$ as the input prompt, expecting that conditioning the TF model on this prompt would allow it to infer an accurate prediction for a new input point $\boldsymbol{x}_n$. As discussed in [59], this provides a supervised-learning flavor to ICL, where the model *implicitly trains* on the data provided within the prompt, and performs inference on test points.

In-context learning	Input prompt	Desired Output
Natural language processing	berry, baya, apple, manzana, banana	plátano
	Japan, mochi, France, croissant, Greece	baklava
Supervised learning $y_i = f(x_i) + \text{noise}$	$x_1, y_1, x_2, \ldots, x_{i-1}, y_{i-1}, x_i$	$f(x_i)$
Dynamical systems $x_{i+1} = f(x_i) + \text{noise}$	$x_1, x_2, x_3, \ldots, x_{i-2}, x_{i-1}, x_i$	$f(x_i)$

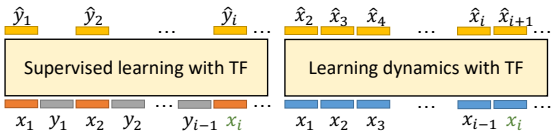
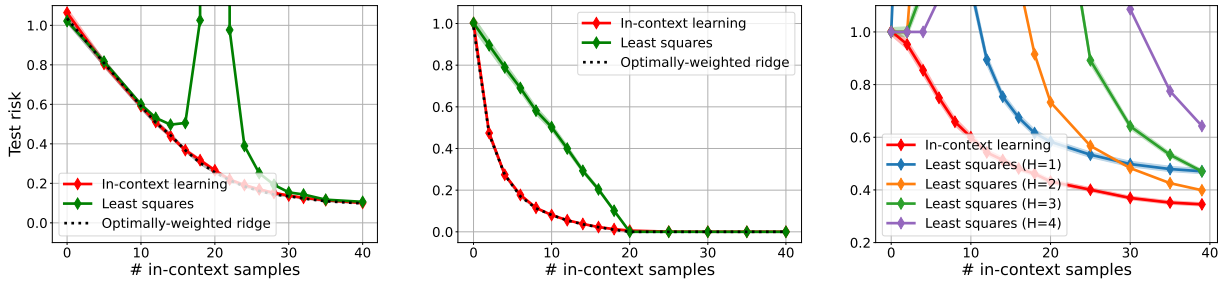


Figure 9.1: Examples of in-context learning. We focus on the lower two settings in the table where a transformer admits a supervised dataset or dynamical system trajectory as a prompt. Then, it autoregressively predicts the output following an input example $\boldsymbol{x}_i$ based on the prompt $(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_i)$.

This chapter formalizes in-context learning from a statistical lens, abstracting the transformer as a learning algorithm where the goal is inferring the correct (input, ouput) functional relationship from prompts. We focus on a meta-learning setting where the model is trained on many tasks, allowing ICL to generalize to both new and previously-seen tasks. Our main contributions are:

- ◇ **Generalization bounds (§9.3 & 9.6):** Suppose the model is trained on T tasks each with a data-sequence containing n examples. During training, each sequence is fed to



(a) Noisy linear regression (b) Linear with covariance prior (c) Partially-observed LDS

Figure 9.2: Examples of implicit model-selection in three ICL settings: (a) Noisy linear regression: $y_i \sim \mathcal{N}(\boldsymbol{x}_i^\top \boldsymbol{\beta}, \sigma^2)$ with $\boldsymbol{x}_i, \boldsymbol{\beta} \sim \mathcal{N}(0, \boldsymbol{I})$. (b) Linear data with covariance prior: $y_i = \boldsymbol{x}_i^\top \boldsymbol{\beta}$ with $\boldsymbol{\beta} \sim \mathcal{N}(0, \boldsymbol{\Sigma})$ with non-isotropic $\boldsymbol{\Sigma}$. (c) Partially observed linear dynamics: $\boldsymbol{x}_t = \boldsymbol{C}\boldsymbol{s}_t$ and $\boldsymbol{s}_{t+1} \sim \mathcal{N}(\boldsymbol{A}\boldsymbol{s}_t, \sigma^2 \boldsymbol{I})$ with randomly sampled $\boldsymbol{C}, \boldsymbol{A}$. Each setting trains a transformer with large number of random regression tasks and evaluates on a new task from the same distribution. In (a) and (b), ICL performances match Bayes-optimal decision (weighted linear ridge regression) that adapt to noise level σ and covariance prior $\boldsymbol{\Sigma}$ on the tasks. (c) shows that ICL outperforms auto-regressive least-squares estimators with varying memory H. Implicit model selection refers to transformer's ability to implement a competitive ML algorithm by leveraging the task prior learned during its training. See Sec 9.7 for experimental details.

the model auto-regressively as depicted in Figure 9.1. By abstracting ICL as an *algorithm learning* problem, we establish a multitask (MTL) generalization rate of $1/\sqrt{nT}$. In order to achieve the proper dependence on the sequence length ($1/\sqrt{n}$ factor), we overcome temporal dependencies by relating generalization to algorithmic stability and we provide numerical and theoretical evidence for the stability of transformer-induced algorithms. Experiments demonstrate that ICL indeed benefits from learning across the full task sequence in line with theory.

- ◇ **From multitask to meta-learning (§9.4):** We provide insights into how our MTL bounds can inform generalization ability of ICL on previously unseen tasks (i.e. transfer learning). Our experiments reveal an intriguing *inductive bias phenomenon*: The gap between the transfer risk and MTL risk is *independent of the model complexity* and only depends on the *task complexity* and the number of tasks T.

- ◇ **Insights on model selection (§9.5):** The *algorithm learning* viewpoint motivates model-selection as a natural feature of in-context learning (e.g. whether to implement a linear model or neural net for prediction). We discuss how this enables the selection of the correct hypothesis set for the tasks when algorithms correspond to empirical risk minimization procedures. We verify this model-selection ability through

extensive experiments as illustrated in Figure 9.2. Among others, the experiments reveal that transformer-based algorithm can learn task covariance as a prior to compete with optimally-weighted ridge and outperform least-squares of different orders to learn partially-observed dynamical systems.

9.2 Preliminaries and Problem Setup

Notation. Let $\mathcal{X}$ be the input feature space, and $\mathcal{Y}$ be the output/label space. We use boldface for vector variables. $[n]$ denotes the set $\{1, 2, \ldots, n\}$. $c, C > 0$ denote absolute constants and $\|\cdot\|_{\ell_p}$ denotes the ℓ_p-norm.

In-context learning setting: We denote a length-m prompt containing $m-1$ in-context examples and the m'th input by $\boldsymbol{x}_{\text{prompt}}^{(m)} = (\boldsymbol{z}_1, \boldsymbol{z}_2, \ldots, \boldsymbol{z}_{m-1}, \boldsymbol{x}_m)$. Here $\boldsymbol{x}_m \in \mathcal{X}$ is the input to predict and $\boldsymbol{z}_i \in \mathcal{Z}$ is the i'th in-context example provided within prompt. Let $\texttt{TF}(\cdot)$ denote a transformer (more generally an auto-regressive model) that admits $\boldsymbol{x}_{\text{prompt}}^{(m)}$ as its input and outputs a label $\hat{\boldsymbol{y}}_m = \texttt{TF}(\boldsymbol{x}_{\text{prompt}}^{(m)})$ in $\mathcal{Y}$. Then the model aims to predict the next output denoted by $\hat{\boldsymbol{y}}_m$. In this work, we investigate two scenarios where prompts are obtained from a sequence of i.i.d. (input, label) pairs or a trajectory of a dynamical system.

• *Independent* $(\boldsymbol{x}, \boldsymbol{y})$ *pairs.* Similar to [59], we draw i.i.d. samples $(\boldsymbol{x}_i, \boldsymbol{y}_i)_{i=1}^n \in \mathcal{Z} = \mathcal{X} \times \mathcal{Y}$ from a data distribution. Then a length-m prompt is written as $\boldsymbol{x}_{\text{prompt}}^{(m)} = (\boldsymbol{x}_1, \boldsymbol{y}_1, \ldots \boldsymbol{x}_{m-1}, \boldsymbol{y}_{m-1}, \boldsymbol{x}_m)$, and the model predicts $\hat{\boldsymbol{y}}_m = \texttt{TF}(\boldsymbol{x}_{\text{prompt}}^{(m)}) \in \mathcal{Y}$ for $1 \le m \le n$.

• *Dynamical systems.* In this setting, the prompt is simply the trajectory generated by a dynamical system, namely, $\boldsymbol{x}_{\text{prompt}}^{(m)} = (\boldsymbol{x}_1, \ldots \boldsymbol{x}_{m-1}, \boldsymbol{x}_m)$ and therefore, $\mathcal{Z} = \mathcal{X} = \mathcal{Y}$. Specifically, we investigate the state observed setting that is governed by dynamics $f(\cdot)$ via $\boldsymbol{x}_{m+1} = f(\boldsymbol{x}_m) + \text{noise}$. Here, $\boldsymbol{y}_m := \boldsymbol{x}_{m+1}$ is the label associated to $\boldsymbol{x}_m$, and the model admits $\boldsymbol{x}_{\text{prompt}}^{(m)}$ as input and predicts the next state $\hat{\boldsymbol{y}}_m := \hat{\boldsymbol{x}}_{m+1} = \texttt{TF}(\boldsymbol{x}_{\text{prompt}}^{(m)}) \in \mathcal{X}$.

We first consider the training phase of ICL where we wish to learn a good $\texttt{TF}(\cdot)$ model through multitask learning (MTL). Suppose we have T tasks associated with data distributions $\{\mathcal{D}_t\}_{t=1}^T$. Each task independently samples a training sequence $\mathcal{S}_t = (\boldsymbol{z}_{ti})_{i=1}^n$ according to its distribution. $\mathcal{S}_{\text{all}} = \{\mathcal{S}_t\}_{t=1}^T$ denote the set of all training sequences. We use $\mathcal{S}_t^{(m)} = (\boldsymbol{z}_{t1}, \ldots, \boldsymbol{z}_{tm})$ to denote a subsequence of $\mathcal{S}_t := \mathcal{S}_t^{(n)}$ for $m \le n$ and $\mathcal{S}^{(0)}$ denotes an empty subsequence.

ICL can be interpreted as an implicit optimization on the subsequence $\mathcal{S}^{(m)} = (\boldsymbol{z}_1, \boldsymbol{z}_2, \ldots, \boldsymbol{z}_m)$ to make prediction on $\boldsymbol{x}_{m+1}$. To model this, we abstract the transformer model as a learning algorithm that maps a sequence of data to a prediction function (e.g. gradient descent, empirical risk minimization). Concretely, let $\mathcal{A}$ be a set of algorithm hypothe-

ses such that algorithm $\texttt{Alg} \in \mathcal{A}$ maps a sequence of form $\mathcal{S}^{(m)}$ into a prediction function $f^{\texttt{Alg}}_{\mathcal{S}^{(m)}} : \mathcal{X} \to \mathcal{Y}$. With this, we represent $\texttt{TF}$ via

$$\texttt{TF}(\boldsymbol{x}^{(m+1)}_{\text{prompt}}) = f^{\texttt{Alg}}_{\mathcal{S}^{(m)}}(\boldsymbol{x}_{m+1}). \tag{9.1}$$

This abstraction is without losing generality as we have the explicit relation $f^{\texttt{Alg}}_{\mathcal{S}^{(m)}}(\boldsymbol{x}) := \texttt{TF}((\mathcal{S}^{(m)}, \boldsymbol{x}))$. Given training sequences $\mathcal{S}_{\text{all}}$ and a loss function $\ell(\boldsymbol{y}, \hat{\boldsymbol{y}})$, the ICL training can be interpreted as searching for the optimal algorithm $\texttt{Alg} \in \mathcal{A}$, and the training objective is formulated as follows:

$$\widehat{\texttt{Alg}} = \arg\min_{\texttt{Alg}\in\mathcal{A}} \widehat{\mathcal{L}}_{\mathcal{S}_{\text{all}}}(\texttt{Alg}) := \frac{1}{T}\sum_{t=1}^{T} \widehat{\mathcal{L}}_t(\texttt{Alg}) \tag{ICL-ERM}$$

$$\text{where} \quad \widehat{\mathcal{L}}_t(\texttt{Alg}) = \frac{1}{n}\sum_{i=1}^{n} \ell(\boldsymbol{y}_{ti}, f^{\texttt{Alg}}_{\mathcal{S}^{(i-1)}_t}(\boldsymbol{x}_{ti})).$$

Here, $\widehat{\mathcal{L}}_t(\texttt{Alg})$ is the training loss of task t and $\widehat{\mathcal{L}}_{\mathcal{S}_{\text{all}}}(\texttt{Alg})$ is the task-averaged MTL loss. Let $\mathcal{L}_t(\texttt{Alg}) = \mathbb{E}_{\mathcal{S}_t}[\widehat{\mathcal{L}}_t(\texttt{Alg})]$ and $\mathcal{L}_{\text{MTL}}(\texttt{Alg}) = \mathbb{E}[\widehat{\mathcal{L}}_{\mathcal{S}_{\text{all}}}(\texttt{Alg})] = \frac{1}{T}\sum_{t=1}^{T} \mathcal{L}_t(\texttt{Alg})$ be the corresponding population risks. Observe that, task-specific loss $\widehat{\mathcal{L}}_t(\texttt{Alg})$ is an empirical average of n terms, one for each prompt $\boldsymbol{x}^{(i)}_{\text{prompt}}$ and the corresponding label $\boldsymbol{y}_i$.

To develop generalization bounds, our primary interest is controlling the gap between empirical and population risks. For the problem (ICL-ERM), we wish to bound the excess MTL risk

$$R_{\text{MTL}}(\widehat{\texttt{Alg}}) = \mathcal{L}_{\text{MTL}}(\widehat{\texttt{Alg}}) - \min_{\texttt{Alg}\in\mathcal{A}} \mathcal{L}_{\text{MTL}}(\texttt{Alg}). \tag{MTL-risk}$$

Following the MTL training (ICL-ERM), we also evaluate the model on previously-unseen tasks; this can be thought of as the transfer learning problem. Concretely, let $\mathcal{D}_{\text{task}}$ be a distribution over tasks and draw a target task $\mathcal{T} \sim \mathcal{D}_{\text{task}}$ with data distribution $\mathcal{D}_{\mathcal{T}}$ and a sequence $\mathcal{S}_{\mathcal{T}} = \{\boldsymbol{z}_i\}_{i=1}^{n} \sim \mathcal{D}_{\mathcal{T}}$. Define the empirical and population risks on $\mathcal{T}$ as $\widehat{\mathcal{L}}_{\mathcal{T}}(\texttt{Alg}) = \frac{1}{n}\sum_{i=1}^{n} \ell(\boldsymbol{y}_i, f^{\texttt{Alg}}_{\mathcal{S}^{(i-1)}_{\mathcal{T}}}(\boldsymbol{x}_i))$ and $\mathcal{L}_{\mathcal{T}}(\texttt{Alg}) = \mathbb{E}_{\mathcal{S}_{\mathcal{T}}}[\widehat{\mathcal{L}}_{\mathcal{T}}(\texttt{Alg})]$. Then the transfer risk of an algorithm $\texttt{Alg}$ is defined as $\mathcal{L}_{\text{TFR}}(\texttt{Alg}) = \mathbb{E}_{\mathcal{T}}[\mathcal{L}_{\mathcal{T}}(\texttt{Alg})]$. With this setup, we are ready to state our main contributions.

9.3 Generalization Guarantees for In-context Learning

In this section, we study the i.i.d. data setting with training sequences $\mathcal{S}_t = (\boldsymbol{x}_{ti}, \boldsymbol{y}_{ti})_{i=1}^{n} \overset{\text{i.i.d.}}{\sim} \mathcal{D}_t$ for $t \in [T]$ and establish generalization bounds for in-context learning. Section 9.6 extends

our results to dynamical systems.

Algorithmic Stability

In ICL a training example $(\boldsymbol{x}_i, \boldsymbol{y}_i)$ in the prompt impacts all future decisions of the algorithm from predictions $i+1$ to n. This necessitates us to control the stability to input perturbation of the learning algorithm emulated by the transformer. Our stability condition is borrowed from the algorithmic stability literature. As stated in [20, 21], the stability level of an algorithm is typically in the order of $1/m$ (for realistic generalization guarantees) where m is the training sample size (in our setting prompt length). This is formalized in the following assumption that captures the variability of the transformer output.

Assumption 9.1 (Error Stability [21]) *Let $\mathcal{S} = (\boldsymbol{x}_1, \boldsymbol{y}_1, \ldots, \boldsymbol{x}_m, \boldsymbol{y}_m)$ be a sequence with $m \geq 1$ and $\mathcal{S}^j$ be the sequence where the j'th sample of $\mathcal{S}$ is replaced by $(\boldsymbol{x}'_j, \boldsymbol{y}'_j)$. Error stability holds for a distribution $(\boldsymbol{x}, \boldsymbol{y}) \sim \mathcal{D}$ if there exists a constant $K > 0$ such that for any $\mathcal{S}, (\boldsymbol{x}'_j, \boldsymbol{y}'_j) \in (\mathcal{X} \times \mathcal{Y})$, $j \in [m]$, and $\mathtt{Alg} \in \mathcal{A}$ we have*

$$\left|\mathbb{E}_{(\boldsymbol{x},\boldsymbol{y})}\left[\ell(\boldsymbol{y}, f_{\mathcal{S}}^{\mathtt{Alg}}(\boldsymbol{x}))\right] - \mathbb{E}_{(\boldsymbol{x},\boldsymbol{y})}\left[\ell(\boldsymbol{y}, f_{\mathcal{S}^j}^{\mathtt{Alg}}(\boldsymbol{x}))\right]\right| \leq \frac{K}{m}. \tag{9.2}$$

Let ρ be a distance metric on $\mathcal{A}$. Pairwise error stability holds if for all $\mathtt{Alg}, \mathtt{Alg}' \in \mathcal{A}$ we have

$$\left|\mathbb{E}_{(\boldsymbol{x},\boldsymbol{y})}\left[\ell(\boldsymbol{y}, f_{\mathcal{S}}^{\mathtt{Alg}}(\boldsymbol{x})) - \ell(\boldsymbol{y}, f_{\mathcal{S}}^{\mathtt{Alg}'}(\boldsymbol{x}))\right] - \mathbb{E}_{(\boldsymbol{x},\boldsymbol{y})}\left[\ell(\boldsymbol{y}, f_{\mathcal{S}^j}^{\mathtt{Alg}}(\boldsymbol{x})) - \ell(\boldsymbol{y}, f_{\mathcal{S}^j}^{\mathtt{Alg}'}(\boldsymbol{x}))\right]\right| \leq \frac{K\rho(\mathtt{Alg}, \mathtt{Alg}')}{m}. \tag{9.3}$$

Here (9.2) is our primary stability condition borrowed from [21] and ensures that all algorithms $\mathtt{Alg} \in \mathcal{A}$ are K-stable. We will also use the stronger pairwise stability condition to develop tighter generalization bounds. The following theorem shows that, under mild assumptions, a multilayer transformer obeys the stability condition (9.2). The proof is deferred to Appendix H.1.1 and Theorem H.1.

Theorem 9.1 *Let $\boldsymbol{x}_{prompt}^{(m)}, \boldsymbol{x}_{prompt}^{\prime(m)}$ be two prompts that differ only at the inputs $\boldsymbol{z}_j = (\boldsymbol{x}_j, \boldsymbol{y}_j)$ with $\boldsymbol{z}'_j = (\boldsymbol{x}'_j, \boldsymbol{y}'_j)$. Assume inputs and labels lie within the unit Euclidean ball in $\mathbb{R}^d$ [1]. Shape these prompts into matrices $\boldsymbol{X}_{prompt}, \boldsymbol{X}'_{prompt} \in \mathbb{R}^{(2m-1)\times d}$ respectively. Let $\mathtt{TF}(\cdot)$ be a transformer composed of D layers followed by a linear head: $\mathtt{TF}(\boldsymbol{x}_{prompt}^{(m)}) = \left\langle \boldsymbol{H}, \boldsymbol{X}_{(D)} \right\rangle$. Here*

[1] Here, we assume $\mathcal{X}, \mathcal{Y} \subset \mathbb{R}^d$, otherwise, inputs and labels are both embedded into d-dimensional vectors of proper size.

$\boldsymbol{X}_{(i)}$ *denotes the i'th layer output with* $\boldsymbol{X}_{(0)} := \boldsymbol{X}_{prompt}$ *and*

$$\boldsymbol{X}_{(i)} = \texttt{Parallel\_MLPs}(att(\boldsymbol{X}_{(i-1)})) \quad where \quad att(\boldsymbol{X}) := \mathbb{S}(\boldsymbol{X}\boldsymbol{W}_i\boldsymbol{X}^\top)\boldsymbol{X}\boldsymbol{W}_{vi},$$

and softmax is row-wise. Assume $\texttt{TF}$ *is normalized to output* $[-1, 1]$ *as follows:* $\|\boldsymbol{W}_v\| \leq 1$, $\|\boldsymbol{W}\| \leq \Gamma/2$, *MLPs obey* $\texttt{MLP}(\boldsymbol{x}) = ReLU(\boldsymbol{M}\boldsymbol{x})$ *with* $\|\boldsymbol{M}\| \leq 1$, *and each row of* $\boldsymbol{H}$ *has Euclidean norm at most* $1/m$. *Then,*

$$|\,\texttt{TF}(\boldsymbol{x}^{(m)}_{prompt}) - \texttt{TF}(\boldsymbol{x}'^{(m)}_{prompt})| \leq \frac{2}{2m-1}((1+\Gamma)e^{\Gamma})^D.$$

Thus, assuming loss $\ell(\boldsymbol{y}, \cdot)$ *is L-Lipschitz, the algorithm induced by* $\texttt{TF}(\cdot)$ *obeys* (9.2) *with* $K = 2L((1+\Gamma)e^{\Gamma})^D$.

A few remarks are in place. First, the dependence on depth is exponential. However this is not as prohibitive for typical transformer architectures which tend to not be very deep. For example the different variants of GPT-2 and BERT have between 12-48 [1] layers. Secondly, if the altered token has an excessive influence on the prediction, this would disrupt the stability condition. Our theorem avoids this via row-wise norm control of the output layer $\boldsymbol{H}$ which ensures that input tokens contribute equally. We anticipate the analysis can be extended to more realistic transformer architectures (e.g. GPT-2 architecture used in our experiments) by incorporating multiple attention-heads and causal masks. Finally, Figure 9.7 provides numerical evidence for multiple ICL settings that demonstrate that stability of GPT-2 architecture's predictions with respect to inputs indeed improves with longer prompt length m in line with our theory.

Generalization Bounds

We are ready to establish generalization bounds by leveraging our stability conditions. We use covering numbers (i.e. metric entropy) to control model complexity.

Definition 9.1 (Covering number) *Let* $\mathcal{Q}$ *be any hypothesis set and* $d(q, q') \geq 0$ *be a distance metric over* $q, q' \in \mathcal{Q}$. *Then,* $\bar{\mathcal{Q}} = \{q_1, \ldots, q_N\}$ *is an* ϵ*-cover of* $\mathcal{Q}$ *with respect to* $d(\cdot, \cdot)$ *if for any* $q \in \mathcal{Q}$, *there exists* $q_i \in \bar{\mathcal{Q}}$ *such that* $d(q, q_i) \leq \epsilon$. *The* ϵ*-covering number* $\mathcal{N}(\mathcal{Q}, d, \epsilon)$ *is the cardinality of the minimal* ϵ*-cover.*

To cover the algorithm space $\mathcal{A}$, we need to introduce a distance metric. We formalize this in terms of the prediction difference between two algorithms on the worst-case data-sequence.

Definition 9.2 (Algorithm distance) *Let* $\mathcal{A}$ *be an algorithm hypothesis set and* $\mathcal{S} = (\boldsymbol{x}_i, \boldsymbol{y}_i)_{i=1}^n$ *be a sequence with* n *examples that is admissible for some task* $t \in$

$[T]$. *For any pair* $\mathtt{Alg}, \mathtt{Alg}' \in \mathcal{A}$, *define the distance metric* $\rho(\mathtt{Alg}, \mathtt{Alg}') := \sup_{\mathcal{S}} \frac{1}{n} \sum_{i=1}^{n} \|f^{\mathtt{Alg}}_{\mathcal{S}^{(i-1)}}(\boldsymbol{x}_i) - f^{\mathtt{Alg}'}_{\mathcal{S}^{(i-1)}}(\boldsymbol{x}_i)\|$.

We note that the distance ρ is controlled by the Lipschitz constant of the transformer architecture (i.e. largest gradient norm with respect to the model weights). Following Definitions 9.1&9.2, the ϵ-covering number of the hypothesis set $\mathcal{A}$ is $\mathcal{N}(\mathcal{A}, \rho, \epsilon)$. This brings us to our main result on MTL risk of (ICL-ERM).

Theorem 9.2 *Suppose* $\mathcal{A}$ *is* K*-stable per Assumption 9.1 and the loss function* $\ell(\boldsymbol{y}, \cdot)$ *is* L*-Lipschitz for all* $\boldsymbol{y} \in \mathcal{Y}$ *and takes values over* $[0, 1]$. *Let* $\widehat{\mathtt{Alg}}$ *be the empirical solution of* (ICL-ERM). *Then, with probability at least* $1 - 2\delta$, *the excess MTL test risk obeys*

$$R_{MTL}(\widehat{\mathtt{Alg}}) \leq \inf_{\epsilon > 0} \left\{ 4L\epsilon + 2(1 + K \log n) \sqrt{\frac{\log \mathcal{N}(\mathcal{A}, \rho, \epsilon) + \log(1/\delta)}{cnT}} \right\}. \tag{9.4}$$

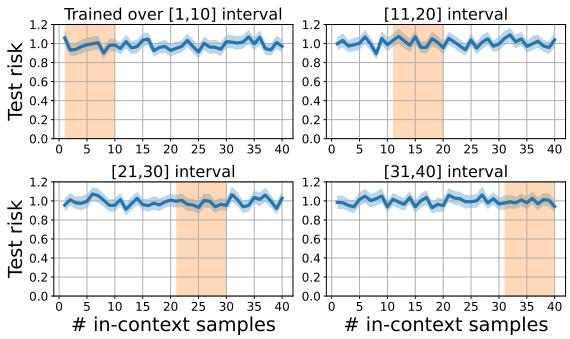
Additionally suppose $\mathcal{A}$ *is* K*-pairwise-stable and set diameter* $D := \sup_{\mathtt{Alg}, \mathtt{Alg}' \in \mathcal{A}} \rho(\mathtt{Alg}, \mathtt{Alg}')$. *Using the convention* $x_+ = \max(x, 1)$, *with probability at least* $1 - 4\delta$,

$$R_{MTL}(\widehat{\mathtt{Alg}}) \leq \inf_{\epsilon > 0} \left\{ 8L\epsilon + \frac{L_+ + K \log n}{\sqrt{cnT}} \Big(\int_{\epsilon}^{D/2} \sqrt{\log \mathcal{N}(\mathcal{A}, \rho, u)} du + D_+ \sqrt{\log(\log(D/\epsilon)/\delta)} \Big) \right\}. \tag{9.5}$$

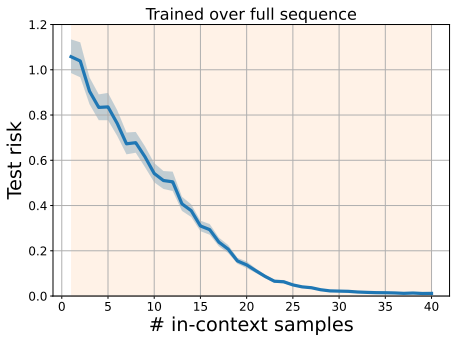
The first bound (9.4) achieves $1/\sqrt{nT}$ rate by covering the algorithm space with resolution ϵ. For Lipschitz architectures with $\dim(\mathcal{A})$ trainable weights we have $\log \mathcal{N}(\mathcal{A}, \rho, \epsilon) \sim \dim(\mathcal{A}) \log(1/\epsilon)$. Thus, up to logarithmic factors, the excess risk is bounded by $\sqrt{\frac{\dim(\mathcal{A})}{nT}}$ and will vanish as $n, T \to \infty$. Recalling (9.1), this can also be seen as a generalization bound for general auto-regressive models that learns from sequential data.

Note that ρ in Def. 9.2 is task-dependent. For instance, suppose tasks are realizable with labels $\boldsymbol{y} = f(\boldsymbol{x})$ and admissible task sequences have the form $\mathcal{S} = (\boldsymbol{x}_i, f(\boldsymbol{x}_i))_{i=1}^n$. Then, ρ will depend on the function class of f (e.g. whether tasks f are linear models, neural nets, etc), specifically, as the function class becomes richer, both ρ and the covering number becomes larger. Under the stronger pairwise-stability, we can obtain a bound in terms of the Dudley entropy integral which arises from a chaining argument. This bound is typically in the same order as the Rademacher complexity of the function class with $T \times n$ samples [198]. Note that achieving $1/\sqrt{T}$ dependence is rather straightforward as tasks are sampled independently. Thus, the main feature of Theorem 9.2 is obtaining the multiplicative $1/\sqrt{n}$ term by overcoming temporal dependencies. Figure 9.3 shows that training with full sequence is indeed critical for ICL accuracy.

Proof sketch. Let $\mathtt{Alg}^\star$ be the population MTL minima. We first decompose $R_{\mathrm{MTL}}(\widehat{\mathtt{Alg}})$



(a) ICL trained with 10 prompts over a range



(b) ICL with all $n = 40$ prompts

Figure 9.3: The value of learning across the full sequence: **Right side:** Standard ICL-ERM where each task trains with all $n = 40$ prompts (full task sequence). **Left side:** ICL-ERM focuses on different parts of the trajectory by fitting $n/4 = 10$ prompts per task over $i \in [1, 10]$ to $[31, 40]$ (highlighted as the orange ranges). We train with $T = 6400$k random linear regression tasks and display the performance on new tasks from the same prior (i.e. transfer risk). Right side learns to solve linear regression via ICL whereas left side fails to do so even when restricted to their target ranges.

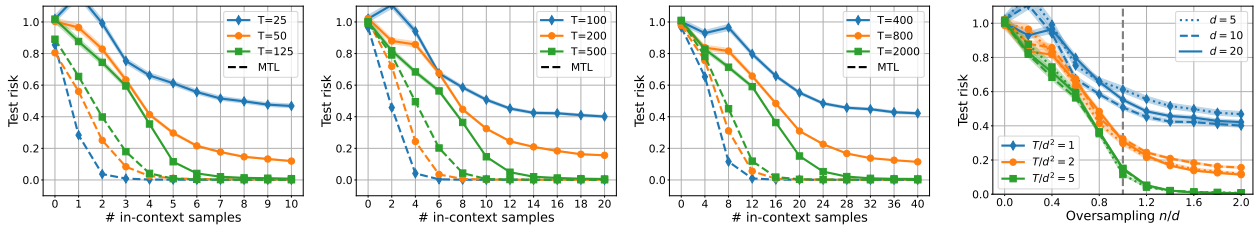
as follows:

$$R_{\mathrm{MTL}}(\widehat{\mathtt{Alg}}) = \underbrace{\mathcal{L}_{\mathrm{MTL}}(\widehat{\mathtt{Alg}}) - \widehat{\mathcal{L}}_{\mathcal{S}_{\mathrm{all}}}(\widehat{\mathtt{Alg}})}_{a} + \underbrace{\widehat{\mathcal{L}}_{\mathcal{S}_{\mathrm{all}}}(\widehat{\mathtt{Alg}}) - \widehat{\mathcal{L}}_{\mathcal{S}_{\mathrm{all}}}(\mathtt{Alg}^\star)}_{b} + \underbrace{\widehat{\mathcal{L}}_{\mathcal{S}_{\mathrm{all}}}(\mathtt{Alg}^\star) - \mathcal{L}_{\mathrm{MTL}}(\mathtt{Alg}^\star)}_{c}$$

where $b \leq 0$ by definition. Regarding (a) and (c), we need a concentration bound on $|\mathcal{L}_{\mathrm{MTL}}(\mathtt{Alg}) - \widehat{\mathcal{L}}_{\mathcal{S}_{\mathrm{all}}}(\mathtt{Alg})|$ for a fixed algorithm $\mathtt{Alg} \in \mathcal{A}$. To achieve this bound, we introduce the sequence of variables $X_{t,i} = \mathbb{E}\left[\frac{1}{n}\sum_{j=1}^{n} \ell(\boldsymbol{y}_{tj}, f^{\mathtt{Alg}}_{\mathcal{S}_t^{(j-1)}}(\boldsymbol{x}_{tj})) \;\middle|\; \mathcal{S}_t^{(i)}\right]$ for $0 \leq i \leq n$, which form a martingale by construction.

The critical stage is upper bounding the martingale differences $|X_{t,i} - X_{t,i-1}|$ through our stability assumption, namely, increments are at most $1 + \sum_{j=i}^{n} \frac{K}{j} \lesssim 1 + K \log n$. Then, we utilize Azuma-Hoeffding's inequality to achieve a concentration bound on $\left|\frac{1}{T}\sum_{t=1}^{T} X_{t,0} - X_{t,n}\right|$, which is equivalent to $|\mathcal{L}_{\mathrm{MTL}}(\mathtt{Alg}) - \widehat{\mathcal{L}}_{\mathcal{S}_{\mathrm{all}}}(\mathtt{Alg})|$. To conclude, we make use of covering/chaining arguments to establish a uniform concentration bound on $|\mathcal{L}_{\mathrm{MTL}}(\mathtt{Alg}) - \widehat{\mathcal{L}}_{\mathcal{S}_{\mathrm{all}}}(\mathtt{Alg})|$ for all $\mathtt{Alg} \in \mathcal{A}$. ■

Multiple sequences per task. Finally consider a setting where each task is associated with M independent sequences with size n. This typically arises in reinforcement learning problems (e.g. dynamical systems in Sec. 9.6) where we collect data through multiple rollouts each leading to independent sequences. In this setting, statistical error rate improves to $1/\sqrt{nMT}$ as discussed in Appendix H.2.1. In the next section, we will contrast MTL vs



(a) $d = 5$ (b) $d = 10$ (c) $d = 20$ (d) $d = 5, 10, 20$ overlayed

Figure 9.4: In Figures (a,b,c,), we plot the $d \in \{5, 10, 20\}$-dimensional results for transfer and MTL risk curves. Figure (d) overlays (a,b,c) to reveal that transfer risks are aligned for fixed $(n/d, T/d^2)$ choice.

transfer learning by letting $M \to \infty$. This way, even if n and T are fixed, the model will fully learn the T source tasks during the MTL phase as the excess risk vanishes with $M \to \infty$.

9.4 Generalization and Inductive Bias on Unseen Tasks

In this section, we explore transfer learning to assess the performance of in-context learning across new tasks: the MTL phase generates a pretrained model $\widehat{\texttt{Alg}}$ trained on T source tasks and we use $\widehat{\texttt{Alg}}$ to predict a target task $\mathcal{T}$. To start with, we consider a standard meta-learning setup where T sources are drawn from the task distribution $\mathcal{D}_{\text{task}}$ and we evaluate the transfer risk on new tasks $\mathcal{T} \sim \mathcal{D}_{\text{task}}$. We are interested in controlling the transfer risk $\mathcal{L}_{\text{TFR}}(\widehat{\texttt{Alg}}) = \mathbb{E}_{\mathcal{T}}[\mathcal{L}_{\mathcal{T}}(\widehat{\texttt{Alg}})]$ in terms of the MTL risk $\mathcal{L}_{\text{MTL}}(\widehat{\texttt{Alg}})$. Since the source tasks are i.i.d, we have $\mathbb{E}_{\text{source}}[\mathcal{L}_{\text{MTL}}(\widehat{\texttt{Alg}})] = \mathcal{L}_{\text{TFR}}(\widehat{\texttt{Alg}})$. More specifically, one can use a standard generalization analysis to control the transfer risk in terms of MTL risk as follows

$$\left|\mathcal{L}_{\text{TFR}}(\widehat{\texttt{Alg}}) - \mathcal{L}_{\text{MTL}}(\widehat{\texttt{Alg}})\right| \lesssim \sqrt{\frac{\dim(\mathcal{A})}{T}}.$$

Here, $\dim(\mathcal{A})$ captures the complexity of the algorithm space (e.g. as formalized in Theorem H.4 in Appendix H.2.3, one can use covering numbers similar to Theorem 9.2). Here, an important distinction with MTL is that transfer risk decays as $1/\text{poly}(T)$ because the unseen tasks induce a distribution shift which, typically, cannot be mitigated with more samples n or more sequences M.

• **Inductive Bias in Transfer Risk.** Before investigating distribution shift, let us consider the following question: While $1/\text{poly}(T)$ behavior may be unavoidable, is it possible that dependence on architectural complexity $\dim(\mathcal{A})$ is avoidable? Perhaps surprisingly, we answer this question affirmatively through experiments on linear regression. In what follows,

during MTL pretraining, we train with $M \to \infty$ independent sequences per task to minimize population MTL risk $\mathcal{L}_{\text{MTL}}(\cdot)$. We then evaluate resulting $\widehat{\texttt{Alg}}$ on different dimensions d and numbers of MTL tasks T. Figures 9.4(a,b,c) display the MTL and transfer risks for dimensions $d = 5, 10, 20$. In each figure, we evaluate the results on $T = \{1, 2, 5\} \times d^2$ and the x-axis moves from 0 to $n = 2d$. Each task has isotropic features, noiseless labels and task vectors $\boldsymbol{\beta} \sim \mathcal{N}(0, \boldsymbol{I}_d)$. Here, our first observation is that, the Figures 9.4(a,b,c) seem (almost perfectly) aligned with each other, that is, each figure exhibits identical MTL and transfer risk curves.

To further elucidate this, Figure 9.4(d) integrates the transfer risk curves from $d = 5, 10, 20$ and overlays them together. This alignment indicates that, for a fixed point $\alpha = n/d$ and $\beta = T/d^2$, the transfer risks remain unchanged. Here, n proportional to d can be attributed to linearity, thus, the more surprising aspect is the dependence on number of tasks T since this indicates that, rather than $\frac{\dim(\mathcal{A})}{T}$, the generalization risk behaves like $\frac{d^2}{T}$. In words, rather than model complexity, what matters is the problem complexity d.

Inductive bias is a natural explanation of this behavior: Intuitively, the MTL pretraining process identifies a favorable algorithm that lies in the span of the source tasks $\Theta_{\text{MTL}} = (\boldsymbol{\beta}_t)_{t=1}^T$. Specifically, while the transformer model can potentially fit MTL tasks through a variety of algorithms, we speculate that the optimization process is implicitly biased (akin to implicit regularization literature [177, 142, 8]) to an algorithm $\texttt{Alg}(\Theta_{\text{MTL}})$. Such bias would explain the lack of $\dim(\mathcal{A})$ dependence since $\texttt{Alg}(\Theta_{\text{MTL}})$ solely depends on the source tasks. We leave the theoretical exploration of the empirical d^2/T behavior to a future work. However, we provide further discussion on why it is rather surprising both from MTL and transfer learning perspectives.

To this end, let us first introduce the optimal estimator for linear regression with Gaussian task prior.

Minimum Bayes risk: The optimal prediction in terms of average risk can be described explicitly [168, 123] under prior $\boldsymbol{\beta} \sim \mathcal{N}(0, \boldsymbol{\Sigma})$ and is given by the optimally-weighted ridge regression solution

$$\hat{\boldsymbol{\beta}} = (\boldsymbol{X}^\top \boldsymbol{X} + \sigma^2 \boldsymbol{\Sigma}^{-1})^{-1} \boldsymbol{X}^\top \boldsymbol{y}, \tag{9.6}$$

Here $\boldsymbol{X} = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n]^\top \in \mathbb{R}^{n \times d}, \boldsymbol{y} = [\boldsymbol{y}_1, \ldots, \boldsymbol{y}_n]^\top \in \mathbb{R}^n$ are the concatenated features and labels obtained from the task sequence and σ^2 is the label noise variance. With this in mind, what is the ideal algorithm $\texttt{Alg}(\Theta_{\text{MTL}})$ based on the (perfect) knowledge of source tasks? Eqn. (9.6) crucially requires the knowledge of the task covariance $\boldsymbol{\Sigma}$ and variance σ^2. Thus, even with the hindsight knowledge that we are dealing with a linear problem, we have

to estimate the task covariance from source tasks. This can be done by constructing the empirical covariance $\hat{\boldsymbol{\Sigma}} = \frac{1}{T}\sum_{i=1}^{T} \boldsymbol{\beta}_i \boldsymbol{\beta}_i^\top$. To make sure that $\hat{\boldsymbol{\Sigma}}$-weighted LS performs $\mathcal{O}(1)$-close to $\boldsymbol{\Sigma}$-weighted LS, we need a spectral norm control, namely, $\|\boldsymbol{\Sigma} - \hat{\boldsymbol{\Sigma}}\|/\lambda_{\min}(\boldsymbol{\Sigma}) \leq \mathcal{O}(1)$. When $\boldsymbol{\Sigma} = \boldsymbol{I}_d$ (as in our experiments) and tasks are isotropic, the latter condition holds with high probability when $T = \Omega(d)$. This is also numerically demonstrated in Figure H.1 in the appendix. This behavior is in contrast to the stronger $T \propto d^2$ requirement we observe for ICL. For instance, $T \propto d^2$ is sufficient to ensure the stronger entrywise control $\|\boldsymbol{\Sigma} - \hat{\boldsymbol{\Sigma}}\|_{\ell_\infty} \leq O(1)$ rather than spectral norm.

Let us now ask the same question for the MTL algorithm: If the transformer perfectly learns the MTL tasks $\Theta_{\text{MTL}} = (\boldsymbol{\beta}_t)_{t=1}^T$, it does not actually need $n = \Omega(d)$ samples to perform well on new prompts drawn from source tasks. To see this, consider the following algorithm: Given a prompt, $\texttt{Alg}(\Theta_{\text{MTL}})$ conducts a discrete search over $(\boldsymbol{\beta}_t)_{t=1}^T$ and returns the source task that best fits to the prompt. Thanks to the discrete search space, it is not hard to see that, we need $n \propto \log(T)$ samples rather than $n \propto d$ (also see Figure H.1). In contrast, based on Figures 9.4(a,b,c), MTL behaves closer to $n \propto d$ empirically. On the other hand, $\texttt{Alg}(\Theta_{\text{MTL}})$ implemented by the transformer is rather intelligent: This is because MTL risks for $d \in \{5, 10, 20\}$ are all strictly better than implementing least-squares[2] and the performance improves as T gets smaller. We leave the thorough exploration of the inductive bias of the MTL training and characterization of $\texttt{Alg}(\Theta_{\text{MTL}})$ as an intriguing future direction.

Exploring transfer risk in terms of distance. Besides drawing source and target tasks from the same $\mathcal{D}_{\text{task}}$, we can also investigate transfer risk in an instance specific fashion. Specifically, the population risk of a new task $\mathcal{T}$ can be bounded as $\mathcal{L}_{\mathcal{T}}(\texttt{Alg}) \leq \mathcal{L}_{\text{MTL}}(\texttt{Alg}) + \text{dist}(\mathcal{T}, (\mathcal{D}_t)_{t=1}^T)$. Here, $\text{dist}(\cdot)$ assesses the (distributional) distance of task $\mathcal{T}$ to the source tasks $(\mathcal{D}_t)_{t=1}^T$ (e.g. [19, 72]). In case of linear tasks, we can simply use the Euclidean distance between task vectors, specifically, the distance of target weights $\boldsymbol{\beta}_{\mathcal{T}}$ to the nearest source task $\text{dist}(\mathcal{T}) = \min_{t\in[T]} \|\boldsymbol{\beta}_{\mathcal{T}} - \boldsymbol{\beta}_t\|$. In Fig. 9.5 we investigate the distance of specific target tasks from source tasks and how the distance affects the transfer performance. Here, all source and target tasks have unit Euclidean norms so that closer distance is equivalent to larger cosine similarity. We again train each MTL task with multiple sequences $M \to \infty$ (as in Fig. 9.4)

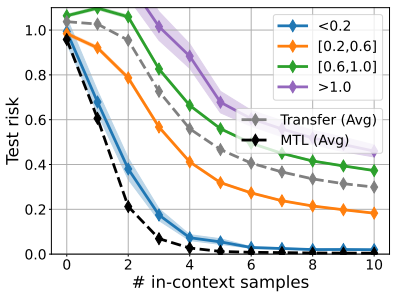


Figure 9.5: Transfer risk as a function of distance to the MTL tasks.

[2]Ordinary least-squares achieves the minimum risk for transfer learning ($T = \infty$) however it is not optimal for finite T.

and use $T = 20$ source tasks with $d = 5$ dimensional regression problems. In a nutshell, Figure 9.5 shows that task similarity (based on ℓ_2 distance) is indeed highly predictive of transfer performance across different distance slices (namely $[0, 0.2], [0.2, 0.6], [0.6, 1], [1, 2]$). An intriguing question is identifying the broader conditions under which transfer risk can be rigorously related to the MTL risk, which is further discussed in Appendix H.2.2.

9.5 Insights on Model Selection

In Section 9.3, we study the generalization error of ICL, which can be eliminated by increasing sample size n or number of sequences M per task. In this section, we will discuss implicit model selection in ICL building on the formalism that transformer is a learning algorithm. Following Figure 9.2 and prior works [59, 100, 77], a plausible assumption is that, transformer can implement ERM algorithms up to a certain accuracy. Then, model selection can be formalized by the selection of the right hypothesis class so that running ERM on that hypothesis class can strike a good bias-variance tradeoff during ICL.

To proceed with our discussion, let us consider the following hypothesis which states that transformer can implement an algorithm competitive with ERM.

Hypothesis 9.1 *Let $\mathbb{F} = (\mathcal{F}_h)_{h=1}^H$ be a family of H hypothesis classes. Let $\mathcal{S} = (\boldsymbol{x}_i, \boldsymbol{y}_i)_{i=1}^n$ be a data-sequence with n examples sampled i.i.d. from $\mathcal{D}$ and let $\mathcal{S}^{(m)} = (\boldsymbol{x}_i, \boldsymbol{y}_i)_{i=1}^m$ be the first m examples. Consider the risk[3] associated to ERM with m samples over $\mathcal{F}_h \in \mathbb{F}$:*

$$\textit{risk}(h, m) = \mathbb{E}_{(\boldsymbol{x}, \boldsymbol{y}, \mathcal{S}^{(m)})}[\ell(\boldsymbol{y}, \hat{f}^{(h)}_{\mathcal{S}^{(m)}}(\boldsymbol{x}))] \quad \textit{where} \quad \hat{f}^{(h)}_{\mathcal{S}^{(m)}} = \arg\min_{f \in \mathcal{F}_h} \frac{1}{m} \sum_{i=1}^m \ell(\boldsymbol{y}_i, f(\boldsymbol{x}_i)),$$

Let $(\epsilon^{h,m}_{\texttt{TF}}) > 0$ be approximation errors associated with $(\mathcal{F}_h)_{h=1}^H$. There exists $\texttt{Alg} \in \mathcal{A}$ such that, for any $m \in [n], h \in [H]$, $f^{\texttt{Alg}}_{\mathcal{S}^{(m)}}$ can approximate ERM in terms of population risk, i.e.

$$\mathbb{E}_{(\boldsymbol{x}, \boldsymbol{y}, \mathcal{S}^{(m)})}[\ell(\boldsymbol{y}, f^{\texttt{Alg}}_{\mathcal{S}^{(m)}}(\boldsymbol{x}))] \leq \textit{risk}(h, m) + \epsilon^{h,m}_{\texttt{TF}}.$$

For model selection purposes, these hypothesis classes can be entirely different ML models, for instance, $\mathcal{F}_1 = \{\text{convolutional-nets}\}$, $\mathcal{F}_2 = \{\text{fully-connected-nets}\}$, and $\mathcal{F}_3 = \{\text{decision-trees}\}$. Alternatively, they can be a nested family useful for capacity control purposes. For instance, Figures 9.2(a,b) are learning covariance/noise priors to implement a constrained-ridge regression. Here $\mathbb{F}$ can be indexed by positive-definite matrices $\boldsymbol{\Sigma}$ with linear classes of the form $\mathcal{F}_{\boldsymbol{\Sigma}} = \{f(\boldsymbol{x}) = \boldsymbol{x}^\top \boldsymbol{\beta} \quad \text{where} \quad \boldsymbol{\beta}^\top \boldsymbol{\Sigma}^{-1} \boldsymbol{\beta} \leq 1\}$.

[3]Since the loss ℓ is bounded by 1, $0 \leq \text{risk}(h, m) \leq 1$ for all m including the scenario $m = 0$ and ERM is vacuous.

Under Hypothesis 1, ICL selects the most suitable class that minimizes the excess risk for each $m \in [n]$.

Following [59, 100, 77], some instructive examples for family $\mathbb{F}$ are

- $\mathbb{F}_\lambda = \{\mathcal{F}_\lambda : \text{Linear model with parameter bounded by } \|\boldsymbol{\beta}\| \leq \lambda\}$
- $\mathbb{F}_{\boldsymbol{\Sigma}} = \{\mathcal{F}_{\boldsymbol{\Sigma}} : \text{Linear model with covariance } \boldsymbol{\Sigma},\ \mathbb{E}[\boldsymbol{\beta}\boldsymbol{\beta}^\top] = \boldsymbol{\Sigma}\}$.
- $\mathbb{F}_s^{\text{sparse}} = \{\mathcal{F}_s : s\text{-sparse linear model}\}$
- $\mathbb{F}_s^{\text{NN}} = \{\mathcal{F}_s : \text{2-layer neural net with width } s\}$
- $\mathbb{F}_s^{\text{RF}} = \{\mathcal{F}_s : \text{Random forest with } s \text{ trees}\}$

Here to simplify, we assumed a discrete family of hypothesis classes, $|\mathbb{F}| = H < \infty$, however, our theory in Section 9.3 is developed for the continuous setting.

Observation 9.1 *Suppose Hypothesis 9.1 holds for a target distribution $\mathcal{D}_\mathcal{T}$. Let $\mathcal{L}_\mathcal{T}^\star := \min_{\texttt{Alg}\in\mathcal{A}} \mathcal{L}_\mathcal{T}(\texttt{Alg})$ be the risk of the optimal algorithm. We have that*

$$\mathcal{L}_\mathcal{T}^\star \leq \frac{1}{n}\sum_{m=0}^{n-1} \min_{h\in[H]} \{risk(h,m) + \epsilon_{\texttt{TF}}^{h,m}\}.$$

Additionally, denote Rademacher complexity of a class $\mathcal{F}$ by $\mathcal{R}_m(\mathcal{F})$. Define the minimum achievable risk over function set $\mathcal{F}_h$ as $\mathcal{L}_h^\star := \min_{f\in\mathcal{F}_h} \mathbb{E}_{\mathcal{D}_\mathcal{T}}[\ell(\boldsymbol{y}, f(\boldsymbol{x}))]$. Since risk$(h,m)$ is controlled by $\mathcal{R}_m(\mathcal{F}_h)$ [133], we have that

$$\mathcal{L}_\mathcal{T}^\star \leq \frac{1}{n}\sum_{m=0}^{n-1} \min_{h\in[H]} \{\mathcal{L}_h^\star + \epsilon_{\texttt{TF}}^{h,m} + \mathcal{O}(\mathcal{R}_m(\mathcal{F}_h))\}.$$

Here, ICL adaptively selects the classes $\arg\min_{h\in[H]}\{\mathcal{L}_h^\star + \mathcal{R}_m(\mathcal{F}_h) + \epsilon_{\texttt{TF}}^{h,m}\}$ to achieve small risk. This is in contrast to training over a single large class $\mathcal{F} = \bigcup_{i=1}^H \mathcal{F}_i$, which would result in a less favorable bound $\approx \min_{h\in[H]} \mathcal{L}_h^\star + \frac{1}{n}\sum_{m=0}^{n-1} \mathcal{R}_m(\mathcal{F})$. A formal version of this statement is provided in Appendix H.2.4. Hypothesis 9.1 assumes a discrete family for simpler exposition ($|\mathbb{F}| = H < \infty$), however, our theory in Section 9.3 allows for the continuous setting.

We emphasize that, in practice, we need to adapt the hypothesis classes for different sample sizes m (typically, more complex classes for larger m). With this in mind, while we have H classes in $\mathbb{F}$, in total we have H^n different ERM algorithms to compete against. This means that VC-dimension of the algorithm class is as large as $n \log H$. This highlights an insightful benefit of our main result: Theorem 9.2 would result in an excess risk $\propto$

$\sqrt{\frac{n \log H}{nT}} = \sqrt{\frac{\log H}{T}}$. In other words, the additional $\times n$ factor achieved through Theorem 9.2 facilitates the adaptive selection of hypothesis classes for each sample size and avoids requiring unreasonably large T.

9.6 Extending Results to Stable Dynamical Systems

Until now, we have studied ICL with sequences of i.i.d. (input, label) pairs. In this section, we investigate the scenario where prompts are obtained from the trajectories of stable dynamical systems, thus, they consist of dependent data. Let $\mathcal{X} \subset \mathbb{R}^d$ and $\mathcal{F} : \mathcal{X} \to \mathcal{X}$ be a hypothesis class elements of which are dynamical systems. During MTL phase, suppose that we are given T tasks associated with $(f_t)_{t=1}^T$ where $f_t \in \mathcal{F}$, and each contains n in-context samples. Then, the data-sequence of t'th task is denoted by $\mathcal{S}_t = (\boldsymbol{x}_{ti})_{i=0}^n$ where $\boldsymbol{x}_{ti} = f_t(\boldsymbol{x}_{t,i-1}) + \boldsymbol{w}_{ti}$, $\boldsymbol{x}_{t0}$ is the initial state, and $\boldsymbol{w}_{ti} \in \mathcal{W} \subset \mathbb{R}^d$ are bounded i.i.d. random noise following some distribution. Then, prompts are given by $\boldsymbol{x}^{(i)}_{\text{prompt}} := (\boldsymbol{x}_0, \boldsymbol{x}_1, \dots \boldsymbol{x}_i)$. Let $\mathcal{S}^{(i)} = \boldsymbol{x}^{(i)}_{\text{prompt}}$, and we can make prediction $\hat{\boldsymbol{x}}_{ti} = f^{\texttt{Alg}}_{\mathcal{S}_t^{(i-1)}}(\boldsymbol{x}_{t,i-1})$. We consider the similar optimization problem as (ICL-ERM).

In order to perform the analysis of generalization error in a dynamical system, we need this system to be stable (which differs from algorithmic stability!). In this work, we use an exponential stability condition that controls the distance between two trajectories initialized from different points [171, 95].

Definition 9.3 ((C_ρ, ρ)-stability) *Denote the state of dynamical system resulting from initial state $\boldsymbol{x}_{t0}$ and $(\boldsymbol{w}_{ti})_{i=1}^m$ by $f_t^{(m)}(\boldsymbol{x}_{t0}) \in \mathcal{X}$. Let $C_\rho \geq 1$ and $\rho \in (0, 1)$ be system related constants. We say that the dynamical system for the task t is (C_ρ, ρ)-stable if, for all $\boldsymbol{x}_{t0}, \boldsymbol{x}'_{t0} \in \mathcal{X}$, $m \geq 1$, and $(\boldsymbol{w}_{ti})_{i \geq 1} \in \mathcal{W}$, we have*

$$\left\| f_t^{(m)}(\boldsymbol{x}_{t0}) - f_t^{(m)}(\boldsymbol{x}'_{t0}) \right\|_{\ell_2} \leq C_\rho \rho^m \left\| \boldsymbol{x}_{t0} - \boldsymbol{x}'_{t0} \right\|_{\ell_2} \tag{9.7}$$

Assumption 9.2 *There exist $\bar{C}_\rho$ and $\bar{\rho} < 1$ such that all dynamical systems $f \in \mathcal{F}$ are $(\bar{C}_\rho, \bar{\rho})$-stable.*

In addition to the stability of the hypothesis set $\mathcal{F}$, we also leverage the algorithmic-stability of the set $\mathcal{A}$ similar to Assumption 9.1. Different from Assumption 9.1, we restrict the variability of the transformer with respect to Euclidean distance metric, similar to the definition of Lipschitz stability.

Assumption 9.3 (Algorithmic-stability for Dynamical Systems) *Let $\mathcal{S} = (\boldsymbol{x}_0, \boldsymbol{x}_1, \dots, \boldsymbol{x}_m)$ be any dynamical sequence with $m \geq 1$ and $\mathcal{S}^j$ be the sequence such*

that $\boldsymbol{w}_j$ is replaced by $\boldsymbol{w}'_j$ ($j = 0$ implies that $\boldsymbol{x}_0$ is replaced by $\boldsymbol{x}'_0$). As a result, starting with the j'th until the m'th index, the sequence $\mathcal{S}^j$ has different samples $(\boldsymbol{x}'_j, \ldots, \boldsymbol{x}'_m)$. There exists a constant $K > 0$ such that for any $\mathcal{S}$, $\boldsymbol{x}'_0 \in \mathcal{X}$, $\boldsymbol{w}'_j \in \mathcal{W}$, and $j \in [m]$, we have

$$\left|\mathbb{E}_{\boldsymbol{w}_{m+1}}\left[\ell(\boldsymbol{x}_{m+1}, f_{\mathcal{S}}^{\texttt{Alg}}(\boldsymbol{x}_m))\right] - \mathbb{E}_{\boldsymbol{w}_{m+1}}\left[\ell(\boldsymbol{x}'_{m+1}, f_{\mathcal{S}^j}^{\texttt{Alg}}(\boldsymbol{x}'_m))\right]\right| \leq \frac{K}{m}\sum_{i=j}^{m} \|\boldsymbol{x}_i - \boldsymbol{x}'_i\|_{\ell_2}\,. \tag{9.8}$$

Theorem H.1 in the appendix justifies this assumption for multilayer transformer architectures. To proceed, we state the main result of this section.

Theorem 9.3 *Suppose the loss function $\ell(\boldsymbol{x}, \cdot) : \mathcal{X} \times \mathcal{X} \to [0, 1]$ is L-Lipschitz for all $\boldsymbol{x} \in \mathcal{X}$ and Assumptions 9.2&9.3 hold. Assume $\mathcal{X}$ and $\mathcal{W}$ are bounded by $\bar{x}, \bar{w}$, respectively. Then, with the same probability, the identical bound as in Theorem 9.2 Eqn.* (9.4) *holds after updating K to be $\bar{K} = 2K\frac{\bar{C}_\rho}{1-\bar{\rho}}(\bar{w} + \bar{x}/\sqrt{n})$.*

The proof of this result is similar to that of Theorem 9.2 and can be found in Appendix H.3. The primary difference occurs when bounding the martingale differences $|X_{t,i} - X_{t,i-1}|$. In the stable dynamical settings, to bound Martingale's increments, we utilize independence of noise and stability of the system instead of using the independence of input and output pairs. One can also obtain an entropy integral bound akin to (9.5) using a proper variation of the pairwise algorithmic-stability condition in Assumption 9.1 Eqn. (9.3).

9.7 Numerical Evaluations

Our experimental setup follows [59]: All ICL experiments are trained and evaluated using the same GPT-2 architecture with 12 layers, 8 attention heads, and 256 dimensional embeddings. Most of our experiments are based on linear regression problems and linear dynamics. We first provide the details and discussions of Figure 9.2 and then provide additional experiments aligned with our findings.

• **Linear regression (Figures 9.2(a,b)).** We consider a d-dimensional linear regression tasks with in-context examples of the form $\boldsymbol{z} = (\boldsymbol{x}, y) \in \mathbb{R}^d \times \mathbb{R}$. Given t'th task $\boldsymbol{\beta}_t$, we generate n i.i.d. samples via $y = \boldsymbol{\beta}_t^\top \boldsymbol{x} + \xi$, where $\boldsymbol{x} \sim \mathcal{N}(0, \boldsymbol{I})$, $\xi \sim \mathcal{N}(0, \sigma^2)$ and σ is the noise level. Tasks are sampled i.i.d. via $\boldsymbol{\beta}_t \sim \mathcal{N}(0, \boldsymbol{\Sigma})$, $t \in [T]$. Results are displayed in Figures 9.2(a)&(b). We set $d = 20$, $n = 40$ and significantly larger T to make sure model is sufficiently trained and we display meta learning results (i.e. on unseen tasks) for both experiments. In Fig. 9.2(a), $\sigma = 1$ and $\boldsymbol{\Sigma} = \boldsymbol{I}$. We also solve ridge-regularized linear

regression (with sample size from 1 to n) over the grid $\lambda = [0.01, 0.05, 0.1, 0.5, 1]$ and display the results of the best λ selection as the optimal ridge curve (Black dotted). Recall from (9.6) that ridge regression is optimal for isotropic task covariance. In Fig. 9.2(b), we set $\sigma = 0$ and $\boldsymbol{\Sigma} = \mathrm{diag}(\left[1, \frac{1}{2^2}, \frac{1}{3^2}, \ldots, \frac{1}{20^2}\right])$. Besides ordinary least squares (Green curve), we also display the optimally-weighted regression according to (9.6) (dotted curve) as $\sigma \to 0$. In both figures, in-context learning results (Red) outperform the least squares results (Green) and are perfectly aligned with optimal ridge/weighted solutions (Black dotted). This in turn provides evidence for the automated model selection ability of transformers by learning task priors.

• **Partially-observed dynamical systems (Figures 9.2(c) & 9.6).** We generate in-context examples $\boldsymbol{z}_i = \boldsymbol{x}_i \in \mathbb{R}^r$, $i \in [n]$ via the partially-observed linear dynamics $\boldsymbol{x}_i = \boldsymbol{C}\boldsymbol{s}_i$, $\boldsymbol{s}_i = \boldsymbol{A}\boldsymbol{s}_{i-1} + \boldsymbol{\xi}_i$ with noise $\boldsymbol{\xi}_i \sim \mathcal{N}(0, \sigma^2 \boldsymbol{I}_d)$ and initial state $\boldsymbol{s}_0 = \boldsymbol{0}$. Each task is parameterized by $\boldsymbol{C} \in \mathbb{R}^{r \times d}$ and $\boldsymbol{A} \in \mathbb{R}^{d \times d}$ which are drawn with i.i.d. $\mathcal{N}(0, 1)$ entries and $\boldsymbol{A}$ is normalized to have spectral radius 0.9. In Fig. 9.2(c), we set $d = 10$, $r = 4$, $\sigma = 0$, $n = 20$ and use sufficiently large T to train the transformer. For comparison, we solve least-squares regression to predict new observations $\boldsymbol{x}_i$ via the most recent H observations for varying window sizes H. Results show that in-context learning outperforms the least-squares results of all orders $H = 1, 2, 3, 4$. In Figure 9.6, we also solve the dynamical problem using optimal ridge regression for different window sizes. This reveals that ICL can also outperform auto-regressive models with optimal ridge tuning, albeit the performance gap is much narrower[4]. It would be interesting to compare ICL performance to a broader class of system identification algorithms (e.g. Hankel nuclear norm, kernel-based, atomic norm [127, 56, 159]) and more broadly understand the extent ICL can inform practically-relevant system identification.

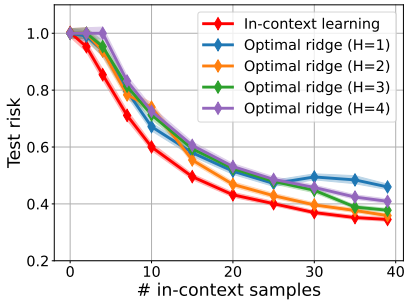


Figure 9.6: Dynamical system experiments. The difference from Fig. 9.2(c) is that we compare ICL to the optimally-tuned ridge regression with different history windows H.

• **Stability analysis.** In Assumption 9.1, we require that transformer-induced algorithms are stable to input perturbations, specifically, we require predictions to vary by at most $\mathcal{O}(1/m)$ where m is the sample size. This was justified in part by Theorem 9.1. To under-

[4]We note that Kalman filter can be used to achieve optimal state estimation (which leverages the full history i.e. $H \to \infty$). However it would require the knowledge of the ground-truth dynamics unlike ICL.

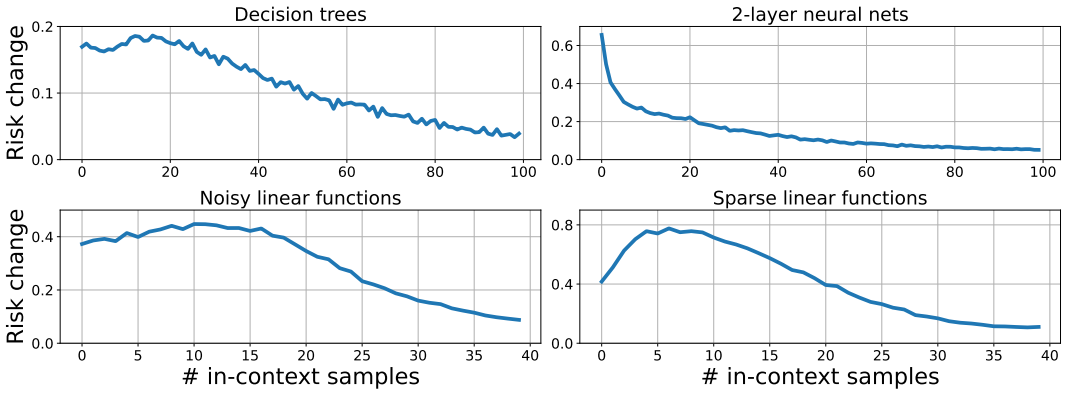


Figure 9.7: Experiments to assess the algorithmic stability Assumption 9.1. Each figure shows the increase in the risk for varying ICL sample sizes after an example in the prompt is modified. We swap an input example in the prompt and assign a flipped label to this new input, e.g., we move from $(\boldsymbol{x}, f(\boldsymbol{x}))$ to $(\boldsymbol{x}', -f(\boldsymbol{x}'))$.

stand empirical stability, we run additional experiments where the results are displayed in Fig. 9.7. We study stability of four function classes: linear models, 3-sparse linear models, decision trees with depth 4, and 2-layer ReLU networks with 100 hidden units, all with input dimension of 20. For each class $\mathcal{F}$, a GPT-2 architecture is trained with large number of random tasks $f \in \mathcal{F}$ and evaluate on new tasks. With the exception of Fig. 2(a), we use the pretrained models provided by [59] and the task sequences are noiseless i.e. sequences obey $y_i = f(\boldsymbol{x}_i)$. As a coarse approximation of the *worst-case* perturbation, we perturb a prompt $\boldsymbol{x}^{(m)}_{\text{prompt}} = (\boldsymbol{x}_1, y_1, \cdots, \boldsymbol{x}_{m-1}, y_{m-1}, \boldsymbol{x}_m)$ as follows. Draw a random point $(\boldsymbol{x}'_1, y'_1) \sim (\boldsymbol{x}_1, y_1)$ and flip its label to obtain $(\boldsymbol{x}'_1, -y'_1)$. We obtain the adversarial prompt via $\bar{\boldsymbol{x}}^{(m)}_{\text{prompt}} = (\boldsymbol{x}'_1, -y'_1, \cdots, \boldsymbol{x}_{m-1}, y_{m-1}, \boldsymbol{x}_m)$[5]. In Figure 9.7, we plot the test risk change between the adversarial and standard prompts. All figures corroborate that, after a certain sample size, the risk change noticeably decreases as the in-context sample size increases. This behavior is in line with Assumption 9.1; however, further investigation and longer context window are required to accurately characterize the stability profile (e.g. to verify whether stability is $\mathcal{O}(1/m)$ or not).

[5]To properly verify Assumption 9.1 one should adversarially optimize $\boldsymbol{x}'_1, \boldsymbol{y}'_1$ and also swap the other indices $m > i > 1$.

CHAPTER 10

Chain-of-Thought and Its Emergent Abilities

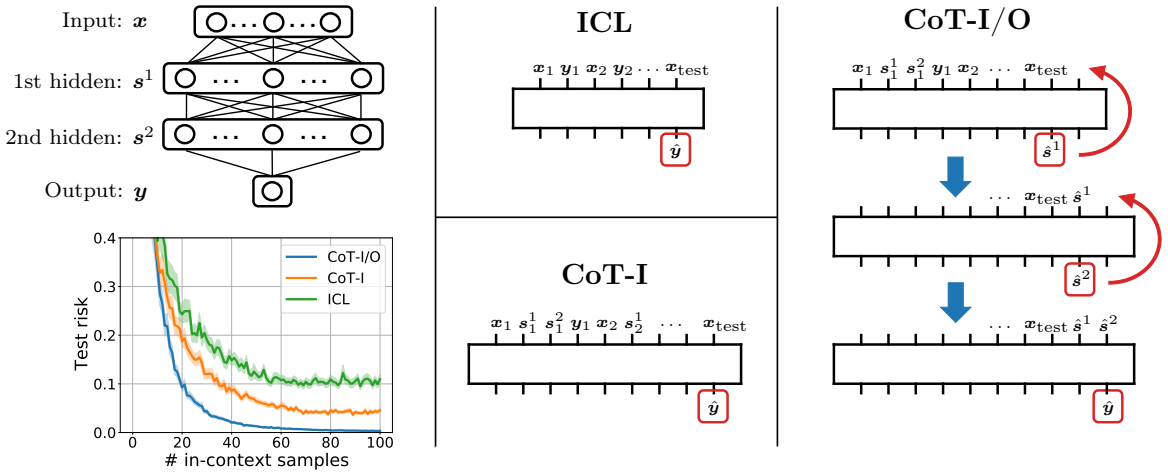


Figure 10.1: An illustration of ICL, CoT-I and CoT-I/O methods, using a 3-layer MLP as an example (top left, where $\boldsymbol{x}$, $\boldsymbol{y}$, $\boldsymbol{s}^1$, $\boldsymbol{s}^2$ denote input, output and hidden features respectivaly). The ICL method utilizes in-context examples in the form of $(\boldsymbol{x}, \boldsymbol{y})$ and makes predictions directly based on the provided $\boldsymbol{x}_{\text{test}}$. Both CoT-I and CoT-I/O methods admit prompts with samples formed by $(\boldsymbol{x}, \boldsymbol{s}^1, \boldsymbol{s}^2, \boldsymbol{y})$. However, CoT-I/O uniquely makes recurrent predictions by re-inputting the intermediate output (as shown on the right). The performance of these methods is shown on the bottom left, with a more detailed discussion available in Section 10.4.

10.1 Introduction

Chain-of-thought prompting [205] is an emergent ability of transformers where the model solves a complex problem [204], by decomposing it into intermediate steps. Intuitively, this underlies the ability of general-purpose language models to accomplish previously-unseen complex tasks by leveraging more basic skills acquired during the pretraining phase. Compositional learning and CoT has enjoyed significant recent success in practical language

modeling tasks spanning question answering, code generation, and mathematical reasoning [158, 82, 212]. In this work, we attempt to demystify some of the mechanics underlying this success and the benefits of CoT in terms of sample complexity and approximation power. To do this we explore the role of CoT in learning multi-layer perceptrons (MLPs) in-context, which we believe can lead to a first set of insightful observations. Throughout, we ask:

Does CoT improve in-context learning of MLPs, and what are the underlying mechanics?

In this chapter, we identify and thoroughly compare three schemes as illustrated in Figure 10.1. (a) ICL: In-context learning from input-output pairs provided in the prompt, (b) CoT-I: Examples in the prompt are augmented with intermediate steps, (c) CoT-I/O: The model also outputs intermediate steps during prediction. Our main contributions are:

- ◇ **Decomposing CoT into filtering and ICL:** As our central contribution, we establish a rigorous and experimentally-supported abstraction that decouples CoT prompting into a *filtering phase* and an *in-context learning (ICL) phase*. In *filtering*, the model attends to the relevant tokens within the prompt based on an instruction. In *ICL*, the model runs inference on the filtered prompt to output a *step* and then moves onto the next *step* in the chain. Our Theorem 10.1 develops a theoretical understanding of this two-step procedure and formalizes how filtering and ICL phases of CoT can be implemented via the self-attention mechanism to learn MLPs.

- ◇ **Approximation and sample complexity:** Through experiments and theory, we establish that intermediate steps in CoT-I improves the sample complexity of learning whereas step-by-step output improves the approximation ability through looping. Specifically, CoT-I/O can learn an MLP with input dimension d and k neurons using $\mathcal{O}(\max(k, d))$ in-context samples by filtering individual layers and solving them via linear regression – in contrast to the $\Omega(kd)$ lower bound without step-augmented prompt. As predicted by our theory, our experiments (see Sec. 10.4) identify a striking universality phenomenon (as k varies) and also demonstrate clear approximation benefits of CoT compared to vanilla ICL.

- ◇ **Accelerated pretraining via learning shortcuts:** We construct deep linear MLPs where each layer is chosen from a discrete set of matrices. This is in contrast to the above setting, where MLP weights can be arbitrary. We show that CoT can dramatically accelerate pretraining by memorizing these discrete matrices and can infer all layers correctly from a *single* demonstration. Notably the pretraining loss goes to zero step-by-step where each step *"learns to filter a layer"*. Together, these

showcase how CoT identifies composable shortcuts to avoid the need for solving linear regression. In contrast, we show that ICL (without CoT) collapses to linear regression performance as it fails to memorize exponentially many candidates (due to lack of composition).

10.2 Preliminaries and Problem Setup

Notation. We denote the set $\{1, 2, \ldots, n\}$ as $[n]$. Vectors and matrices are represented in bold text (e.g., $\boldsymbol{x}, \boldsymbol{A}$), while scalars are denoted in plain text (e.g., y). The input and output domains are symbolized as $\mathcal{X}$ and $\mathcal{Y}$ respectively (unless specified otherwise), and $\boldsymbol{x} \in \mathcal{X}$, $\boldsymbol{y} \in \mathcal{Y}$ denote the input and output. Additionally, let $\mathcal{F}$ be a set of functions from $\mathcal{X}$ to $\mathcal{Y}$. Consider a transition function $f \in \mathcal{F}$ where $\boldsymbol{y} = f(\boldsymbol{x})$. In this section, we explore the properties of learning f, assuming that it can be decomposed into simpler functions $(f_\ell)_{\ell=1}^{L}$, and thus can be expressed as $f = f_L \circ f_{L-1} \circ \cdots \circ f_1$.

10.2.1 In-context Learning

Following the study by [59], the fundamental problem of vanilla in-context learning (ICL) involves constructing a prompt with input-output pairs in the following manner:

$$\boldsymbol{p}_n(f) = (\boldsymbol{x}_i, \boldsymbol{y}_i)_{i=1}^{n} \quad \text{where} \quad \boldsymbol{y}_i = f(\boldsymbol{x}_i). \tag{P-ICL}$$

Here the transition function $f \in \mathcal{F} : \mathcal{X} \to \mathcal{Y}$ remains constant within a single prompt but can vary across prompts, and the subscript n signifies the number of in-context samples contained in the prompt. Considering language translation as an example, f is identified as the target language, and the prompt can be defined as $\boldsymbol{p}(\texttt{Spanish})$ = ((*apple, manzana*), (*ball, pelota*), ...) or $\boldsymbol{p}(\texttt{French})$=((*cat, chat*), (*flower, fleur*), ...). Let `TF` denote any auto-regressive model (e.g., Decoder-only Transformer). The aim of in-context learning is to learn a model that can accurately predict the output, given a prompt $\boldsymbol{p}$ and the test input $\boldsymbol{x}_{\text{test}}$, as shown in the following equation:

$$\texttt{TF}(\boldsymbol{p}_n(\tilde{f}), \boldsymbol{x}_{\text{test}}) \approx \tilde{f}(\boldsymbol{x}_{\text{test}}) \tag{10.1}$$

where $\tilde{f} \in \mathcal{F}$ is the test function which may differ from the functions used during training. Previous work [222, 113] has demonstrated that longer prompts (containing more examples n) typically enhance the performance of the model.

10.2.2 Chain-of-thought Prompt and Prediction

As defined in (P-ICL), the prompt in vanilla ICL only contains input-output pairs of the target function. This demands that the model learns the function $f \in \mathcal{F}$ in one go, which becomes more challenging as $\mathcal{F}$ grows more complex, since larger models and increased prompt length (n) are needed to make correct predictions (as depicted by the green curves in Figures 10.5 and 10.6). Existing studies on chain-of-thought methods (e.g., [205]) observed that prompts containing step-by-step instructions assist the model in decomposing the function and making better predictions. Specifically, consider a function composed of L subfunctions, represented as $f := f_L \circ f_{L-1} \circ \dots f_1$. Each intermediate output can be viewed as a step, enabling us to define a length-n CoT prompt related to f with L steps (expressed with $\boldsymbol{s}^\ell, \ell \in [L]$) as follows:

$$\boldsymbol{p}_n(f) = (\boldsymbol{x}_i, \boldsymbol{s}_i^1, \cdots \boldsymbol{s}_i^{L-1}, \boldsymbol{s}_i^L)_{i=1}^n \quad \text{where} \quad \boldsymbol{s}_i^\ell = f_\ell(\boldsymbol{s}_i^{\ell-1}),\ \ell \in [L]. \tag{P-CoT}$$

Here $\boldsymbol{x}_i = \boldsymbol{s}_i^0$, $\boldsymbol{y}_i = \boldsymbol{s}_i^L$ and $f_\ell \in \mathcal{F}_\ell$, which implies that $f \in \mathcal{F}_L \times \cdots \mathcal{F}_1 := \mathcal{F}$.

Next we introduce two methodologies for making predictions within the CoT framework:

CoT over input only (CoT-I). Contrasted with ICL, CoT-I considers step-by-step instructions as inputs, nonetheless, the prediction for the last token is performed as a single entity. Our experiments indicate that this approach lowers the sample complexity for `TF` to comprehend the function $\tilde{f}$ being learned (see the orange curves in Figures 10.5 and 10.6). The CoT-I prediction aligns with Eq. (10.1) as follows, while the prompt is determined by (P-CoT).

$$\text{One-shot prediction: } \texttt{TF}(\boldsymbol{p}_n(\tilde{f}), \boldsymbol{x}_{\text{test}}) \approx \tilde{f}(\boldsymbol{x}_{\text{test}}). \tag{10.2}$$

CoT over both input and output (CoT-I/O). Despite the fact that CoT-I improves the sample complexity of learning $\tilde{f}$, the `TF` must still possess the capacity to approximate functions from the function class $\mathcal{F}$, given that the prediction is made in one shot. To mitigate this challenge, we consider a scenario where in addition to implementing a CoT prompt, we also carry out CoT predictions. Specifically, for a composed problem with inputs formed via

(P-CoT), the model recurrently makes L-step predictions as outlined below:

$$\begin{aligned}
&\text{Step 1: } \texttt{TF}(\boldsymbol{p}_n(\tilde{f}), \boldsymbol{x}_{\text{test}}) := \hat{\boldsymbol{s}}^1 \\
&\text{Step 2: } \texttt{TF}(\boldsymbol{p}_n(\tilde{f}), \boldsymbol{x}_{\text{test}}, \hat{\boldsymbol{s}}^1) := \hat{\boldsymbol{s}}^2 \\
&\quad \vdots \\
&\text{Setp } L\text{: } \texttt{TF}(\boldsymbol{p}_n(\tilde{f}), \boldsymbol{x}_{\text{test}}, \hat{\boldsymbol{s}}^1 \cdots, \hat{\boldsymbol{s}}^{L-1}) \approx \tilde{f}(\boldsymbol{x}_{\text{test}}),
\end{aligned} \tag{10.3}$$

where at each step (step ℓ), the model outputs an intermediate step ($\hat{\boldsymbol{s}}^\ell$) which is then fed back to the input sequence to facilitate the next-step prediction ($\hat{\boldsymbol{s}}^{\ell+1}$). Following this strategy, the model only needs to learn the union of the sub-function sets, $\bigcup_{\ell=1}^{L} \mathcal{F}_\ell$, whose complexity scales linearly with the number of steps L. Empirical evidence of the benefits of CoT-I/O over ICL and CoT-I in enhancing sample efficiency and model expressivity is reflected in the blue curves shown in Figures 10.5 and 10.6.

10.2.3 Model Training

In Figure 10.1 and Section 10.2, we have discussed vanilla ICL, CoT-I and CoT-I/O methods. Intuitively, ICL can be viewed as a special case of CoT-I (or CoT-I/O) if we assume only one step is performed. Consequently, we will focus on implementing CoT-I and CoT-I/O for model training in the following.

Consider the CoT prompt as in (P-CoT), and assume that $\boldsymbol{x} \sim \mathcal{D}_{\mathcal{X}}$, and $f_\ell \sim \mathcal{D}_\ell, \ell \in [L]$, where L denotes the number of compositions/steps, such that the final prediction should approximate $f(\boldsymbol{x}) = f_L(f_{L-1} \dots f_1(\boldsymbol{x})) := \boldsymbol{y} \in \mathcal{Y}$. We define $\ell(\hat{\boldsymbol{y}}, \boldsymbol{y}) : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$ as a loss function. For simplicity, we assume $f_\ell(\dots f_1(\boldsymbol{x})) \in \mathcal{Y}$, $\ell \in [L]$. Let N represent the in-context window of TF, which implies that TF can only admit a prompt containing up to N in-context samples. Generally, our goal is to ensure high prediction performance given any length-n prompt, where $n \in [N]$. To this end, we train the model using prompts with length from 1 to N equally and aim to minimize the averaged risk over different prompt size. Assuming the model TF is parameterized by $\boldsymbol{\beta}$ and considering meta learning problem, the objective functions for CoT-I and CoT-I/O are defined as follows.

$$\hat{\boldsymbol{\beta}}^{\text{CoT-I}} = \arg\min_{\boldsymbol{\beta}} \mathbb{E}_{(\boldsymbol{x}_n)_{n=1}^N, (f_\ell)_{\ell=1}^L} \left[\frac{1}{N} \sum_{n=1}^{N} \ell(\hat{\boldsymbol{y}}_n, f(\boldsymbol{x}_n)) \right]$$

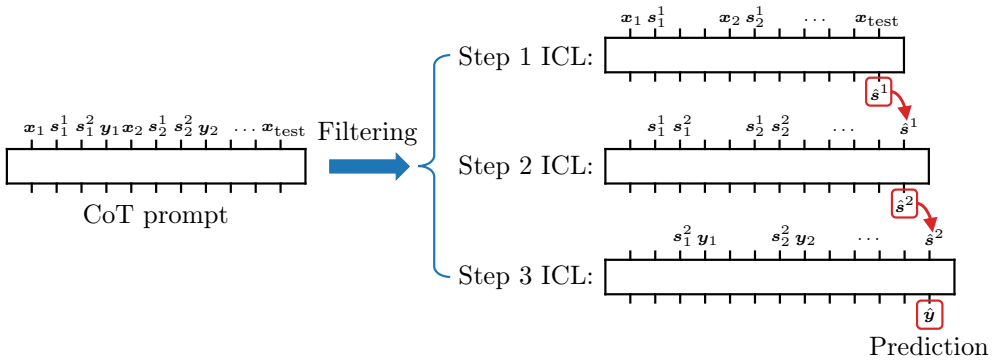


Figure 10.2: Depiction of Theorem 10.1 that formalizes CoT as a Filtering + ICL procedure. In this illustration, we utilize in-context samples as depicted in Figure 10.1, characterizing the CoT prompt by $(\boldsymbol{x}, \boldsymbol{s}^1, \boldsymbol{s}^2, \boldsymbol{y})$. We show a transformer $\texttt{TF}$, composed of $\texttt{TF}_{\texttt{LR}} \circ \texttt{TF}_{\texttt{BE}}$. $\texttt{TF}_{\texttt{BE}}$ facilitates the implementation of filtering on the CoT prompt (refer to (P-CoT), illustrated on the left side of the figure), subsequently generating vanilla ICL prompts (refer to (P-ICL), illustrated on the right side of the figure). During each step of the ICL process, $\texttt{TF}_{\texttt{LR}}$ employs gradient descent techniques to address and solve sub-problems, exemplified by linear regression in the context of solving MLPs.

and

$$\hat{\boldsymbol{\beta}}^{\text{CoT-I/O}} = \arg\min_{\boldsymbol{\beta}} \mathbb{E}_{(\boldsymbol{x}_n)_{n=1}^N, (f_\ell)_{\ell=1}^L} \left[\frac{1}{NL} \sum_{n=1}^{N} \sum_{\ell=1}^{L} \ell(\hat{\boldsymbol{s}}_n^\ell, \boldsymbol{s}_n^\ell) \right]$$

where $\hat{\boldsymbol{y}}_n = \texttt{TF}(\boldsymbol{p}_n(f), \boldsymbol{x}_n)$ and $\hat{\boldsymbol{s}}_n^\ell = \texttt{TF}(\boldsymbol{p}_n(f), \boldsymbol{x}_n \cdots \boldsymbol{s}_n^{\ell-1})$. $\boldsymbol{p}_n(f)$ is given by (P-CoT), and as mentioned previously, $\boldsymbol{s}^0 = \boldsymbol{x}$ and $\boldsymbol{s}^L = \boldsymbol{y}$. All $\boldsymbol{x}$ and f_ℓ are independent, and we take the expectation of the risk over their respective distributions.

10.3 Provable Approximation of MLPs via CoT

In this section, we present our theoretical findings that demonstrate how CoT-I/O can execute filtering over the CoT prompt, thereby learning a 2-layer MLP with input dimension of d and hidden dimension of k, akin to resolving k d-dimensional ReLU problems and 1 k-dimensional linear regression problem. Subsequently, in Section 10.4.1, we examine the performance of CoT-I/O when learning 2-layer random MLPs. Our experiments indicate that CoT-I/O needs only $O(\max(d, k))$ in-context samples to learn the corresponding MLP.

We state our main contribution of establishing a result that decouples CoT-based in-context learning (CoT-I/O) into two phases: (1) *Filtering Phase:* Given a prompt that contains features of multiple MLP layers, retrieve only the features related to a target layer to create an ICL prompt. (2) *ICL Phase:* Given filtered prompt, learn the target layer weights through gradient descent. Combining these two phases, and looping over all layers,

we will show that there exists a transformer architecture such that CoT-I/O can provably approximate a multilayer MLP up to a given resolution. An illustration is provided in Figure 10.2. To state our result, we assume access to an oracle that performs linear regression and consider the consider the condition number of the data matrix.

Definition 10.1 (MLP and condition number) *Consider a multilayer MLP defined by the recursion* $\boldsymbol{s}_i^\ell = \phi(\boldsymbol{W}_\ell \boldsymbol{s}_i^{\ell-1})$ *for* $\ell \in [L]$, $i \in [n]$ *and* $\boldsymbol{s}_i^0 = \boldsymbol{x}_i$. *Here* $\phi(x) := \max(\alpha x, x)$ *is a Leaky-ReLU activation with* $1 \geq \alpha > 0$. *Define the feature matrix* $\boldsymbol{T}_\ell = [\boldsymbol{s}_1^\ell \ \ldots \ \boldsymbol{s}_n^\ell]^\top$ *and define its condition number* $\kappa_\ell = \sigma_{\max}(\boldsymbol{T}_\ell)/\sigma_{\min}(\boldsymbol{T}_\ell)$ *(with* $\sigma_{\min} := 0$ *for fat matrices) and* $\kappa_{\max} = \max_{0 \leq \ell < L} \kappa_\ell$.

Assumption 10.1 (Oracle Model) *We assume access to a transformer* $\texttt{TF}_{\texttt{LR}}$ *which can run* T *steps of gradient descent on the quadratic loss* $\mathcal{L}(\boldsymbol{w}) = \sum_{i=1}^n (y_i - \boldsymbol{w}^\top \boldsymbol{x}_i)^2$ *given a prompt of the form* $(\boldsymbol{x}_1, y_1, \ldots, \boldsymbol{x}_n, y_n)$.

We remark that this assumption is realistic and has been formally established by earlier work [4, 62]. Our CoT abstraction builds on these to demonstrate that CoT-I/O can call a blackbox TFmodel to implement a compositional function when combined with filtering.

We now present our main theoretical contribution. Our result provides a transformer construction that first filters a particular MLP layer from the prompt through the attention mechanism, then applies in-context learning, and repeats this procedure to approximate the MLP output. The precise statement is deferred to the supplementary material.

Theorem 10.1 (CoT⇔Filtering+ICL) *Consider a CoT prompt* $\boldsymbol{p}_n(f)$ *generated from an L-layer MLP* $f(\cdot)$ *as described in Definition 10.1, and assume given test example* $(\boldsymbol{x}_{test}, \boldsymbol{s}_{test}^1, \ldots \boldsymbol{s}_{test}^L)$. *For any resolution* $\epsilon > 0$, *there exists* $\delta = \delta(\epsilon)$, *iteration choice* $T = \mathcal{O}(\kappa_{\max}^2 \log(1/\epsilon))$, *and a backend transformer construction* $\texttt{TF}_{\texttt{BE}}$ *such that the concatenated transformer* $\texttt{TF} = \texttt{TF}_{\texttt{LR}} \circ \texttt{TF}_{\texttt{BE}}$ *implements the following: Let* $(\hat{\boldsymbol{s}}^i)_{i=0}^{\ell-1}$ *denote the first* $\ell - 1$ *CoT-I/O outputs of* TF *where* $\hat{\boldsymbol{s}}^0 = \boldsymbol{x}_{test}$ *and set* $\boldsymbol{p}[\ell] = (\boldsymbol{p}_n(f), \boldsymbol{x}_{test}, \hat{\boldsymbol{s}}^1 \ldots \hat{\boldsymbol{s}}^{\ell-1})$. *At step* ℓ, TF *implements*

1. ***Filtering.*** *Define the filtered prompt with input/output features of layer* ℓ,

$$\boldsymbol{p}_n^{filter} = \begin{pmatrix} \ldots \mathbf{0}, \boldsymbol{s}_1^{\ell-1}, \mathbf{0} \ \ldots \mathbf{0}, \boldsymbol{s}_n^{\ell-1}, \mathbf{0} \ \ldots \mathbf{0}, \hat{\boldsymbol{s}}^{\ell-1} \\ \ldots \mathbf{0}, \ \mathbf{0}, \ \boldsymbol{s}_1^\ell \ldots \mathbf{0}, \ \mathbf{0}, \ \boldsymbol{s}_n^\ell \ldots \mathbf{0}, \ \mathbf{0} \end{pmatrix}.$$

   There exists a fixed projection matrix $\boldsymbol{\Pi}$ *that applies individually on tokens such that the backend output obeys* $\|\boldsymbol{\Pi}(\texttt{TF}_{\texttt{BE}}(\boldsymbol{p}[\ell])) - \boldsymbol{p}_n^{filter}\| \leq \delta$.

2. ***Gradient descent.*** *The combined model obeys* $\|\texttt{TF}(\boldsymbol{p}[\ell]) - \boldsymbol{s}_{test}^\ell\| \leq \ell \cdot \epsilon / L$.

$\texttt{TF}_{\texttt{BE}}$ has constant number of layers independent of n and T. Consequently, after L rounds of CoT-I/O, $\texttt{TF}$outputs $f(\boldsymbol{x}_{test})$ up to ϵ accuracy.

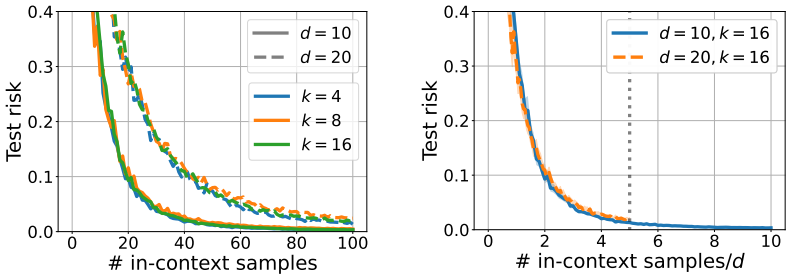
Remark 10.1 *Note that, this result effectively shows that, with a sufficiently good blackbox transformer $\texttt{TF}_{\texttt{LR}}$ (per Assumption 10.1), CoT-I/O can learn an L-layer MLP using in-context sample size $n > \max_{\ell \in [L]} d_\ell$ where d_ℓ is the input dimension of ℓth layer. This is assuming condition number $\kappa_{\max}$ of the problem is finite as soon as all layers become over-determined. Consequently, CoT-I/O needs $\max(k, d)$ sample complexity to learn a two layer MLP. This provides a formal justification for the observation that empirical CoT-I/O performance is agnostic to k as long as $k \leq d$.*

We provide the concrete filtering statements based on the transformer architecture in Appendix I.1, and the key components of our construction are the following: (i) Inputs are projected through the embedding layer in which a set of encodings, an enumeration of the tokens $(1, 2, \ldots, N)$, an enumeration of the layers $(1, 2, \ldots, L)$ and an identifier for each layer already predicted are all attached. Notice that this "modification" to the input only depends on the sequence length and is agnostic to the token to-be-predicted. This allows for an automated looping over L predictions. (ii) We use this information to extract the sequence length N and the current layer ℓ to-be-predicted. (iii) With these at hand, we construct an 'if-then' type of function using the ReLU layers to filter out the samples that are not needed for the prediction.

10.4 Experiments with 2-layer Random MLPs

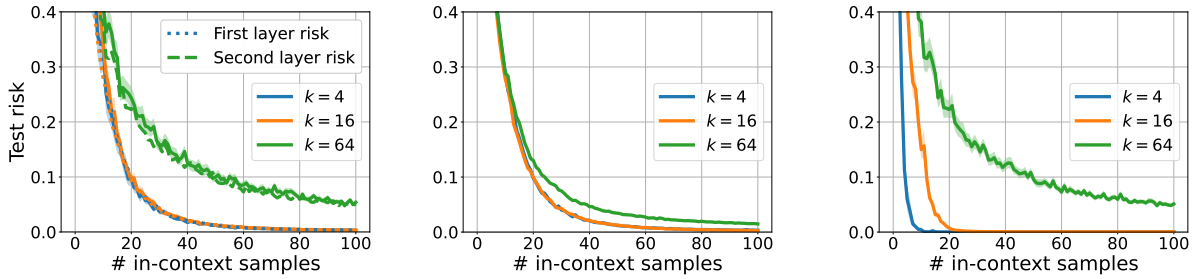
For a clear exposition, we initially focus on two-layer MLPs, which represent 2-step tasks (e.g., $L = 2$). We begin by validating Theorem 10.1 using the CoT-I/O method, demonstrating that in-context learning for a 2-layer MLP with d input dimensions and k hidden neurons requires $O(\max(d, k))$ samples. The results are presented in Section 10.4.1. Subsequently, in Section 10.4.2, we compare three different methods: ICL, CoT-I, and CoT-I/O. The empirical evidence highlights the advantages of CoT-I/O, showcasing its ability to reduce sample complexity and enhance model expressivity.

Dataset. Consider 2-layer MLPs with input $\boldsymbol{x} \in \mathbb{R}^d$, hidden feature (step-1 output) $\boldsymbol{s} \in \mathbb{R}^k$, and output $y \in \mathbb{R}$. Here, $\boldsymbol{s} = f_1(\boldsymbol{x}) := (\boldsymbol{W}\boldsymbol{x})_+$ and $y = f_2(\boldsymbol{s}) := \boldsymbol{v}^\top \boldsymbol{s}$, with $\boldsymbol{W} \in \mathbb{R}^{k \times d}$ and $\boldsymbol{v} \in \mathbb{R}^k$ being the parameters of the first and second layer, and $(x)_+ = \max(x, 0)$ being ReLU activation. The function is composed as $y = \boldsymbol{v}^\top (\boldsymbol{W}\boldsymbol{x})_+$. We define the function distributions as follows: each entry of $\boldsymbol{W}$ is sampled via $\boldsymbol{W}_{ij} \sim \mathcal{N}(0, \frac{2}{k})$, and $\boldsymbol{v} \sim \mathcal{N}(0, \boldsymbol{I}_k)$,



(a) CoT-I/O: $d = 10$ *vs* 20 (b) Alignment in Fig. 10.3a

Figure 10.3: We solve 2-layer MLPs with varying input dimension $d \in \{10, 20\}$ and hidden neuron size $k \in \{4, 8, 16\}$.



(a) CoT-I/O: composed risk (b) Risk of first layer (c) Risk of second layer

Figure 10.4: We solve 2-layer MLPs with $d = 10$ and $k \in \{4, 16, 64\}$. We then decouple the composed risk of predicting 2-layer MLPs into risks of individual layers.

with inputs being randomly sampled through $\boldsymbol{x} \sim \mathcal{N}(0, \boldsymbol{I}_d)$[1]. We apply the quadratic loss in our experiments. To avoid the implicit bias due to distribution shift, both training and test datasets are generated following the same strategy.

10.4.1 Empirical Evaluation of CoT-I/O Performance

To investigate how MLP task impacts CoT-I/O performance, we train 2-layer MLPs with varying input dimensions (d) and hidden layer sizes (k). The results are presented in Figures 10.3 and 10.4, and all experiments utilize small GPT-2 models for training[2].

CoT-I/O performance is agnostic to k when $k \leq d$ (Figure 10.3). In Fig. 10.3a, we train MLPs with $d = 10, 20$ and $k = 4, 8, 16$. Solid and dashed curves represent the CoT-I/O test risk of $d = 10$ and 20 respectively for varying in-context samples. The results

[1]Following this strategy for data generation, the expected norms of $\boldsymbol{x}$, $\boldsymbol{s}$ and y are equivalent, and the risk curves displayed in the figures are normalized for comparison.

[2]Our code is available at https://github.com/yingcong-li/Dissecting-CoT.

indicate that an increase in d will amplifies the number of samples needed for in in-context learning, while the performance remains unaffected by changes in $k \in \{4, 8, 16\}$. To further scrutinize the impact of d on CoT-I/O accuracy, in Fig. 10.3b, we adjust the horizontal axis by dividing it by the input dimension d, and superimpose both $d = 10, k = 16$ (blue solid) and $d = 20, k = 16$ (orange dashed) results. This alignment of the two curves implies that the in-context sample complexity of CoT-I/O is linearly dependent on d.

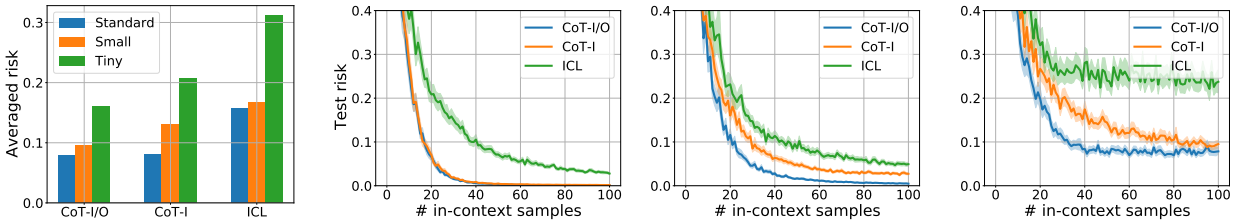
Large k dominates CoT-I/O performance (Figure 10.4). We further investigate the circumstances under which k begins to govern the CoT-I/O performance. In Figure 10.4a, we replicate the same experiments with $d = 10$, but train with wider MLPs ($k = 64$). Blue, orange and green curves represent results for $k = 4, 16, 64$ respectively. Since the hidden dimension $k = 64$ is larger, learning the second layer requires more hidden features ($\boldsymbol{s}$), thus $N = 100$ in-context samples (providing 100 $\boldsymbol{s}$s) are insufficient to fully restore the second layer, leading to performance gaps between $k = 4, 16$ and $k = 64$. To quantify the existing gaps, we conduct single-step evaluations for both the first and the second layers, with the results shown in Figures 10.4b and 10.4c. Specifically, let $\boldsymbol{p}_n(\tilde{f})$ be a test prompt containing n in-context samples where $\tilde{f}$ represents any arbitrary 2-layer MLP. Given a test sample $(\boldsymbol{x}_{\text{test}}, \boldsymbol{s}_{\text{test}}, y_{\text{test}})$, the layer predictions are performed as follows.

$$\text{1st layer prediction: } \texttt{TF}(\boldsymbol{p}_n(\tilde{f}), \boldsymbol{x}_{\text{test}}) := \hat{\boldsymbol{s}},$$
$$\text{2nd layer prediction: } \texttt{TF}(\boldsymbol{p}_n(\tilde{f}), \boldsymbol{x}_{\text{test}}, \boldsymbol{s}_{\text{test}}) := \hat{y}.$$

The test risks are calculated by $\|\hat{\boldsymbol{s}} - \boldsymbol{s}_{\text{test}}\|^2$ and $(\hat{y} - y_{\text{test}})^2$. The risks illustrated in the figures are normalized for comparability (refer to the appendix for more details). Evidence from Fig. 10.4b and 10.4c shows that while increasing k does not affect the first layer's prediction, it does augment the number of samples required to learn the second layer. Moreover, by plotting the first layer risks of $k = 4, 16$ (blue/orange dotted) and second layer risk of $k = 64$ (green dashed) in Fig. 10.4a, we can see that they align with the CoT-I/O composed risks. This substantiates the hypothesis that CoT-I/O learns 2-layer MLP through compositional learning of separate layers.

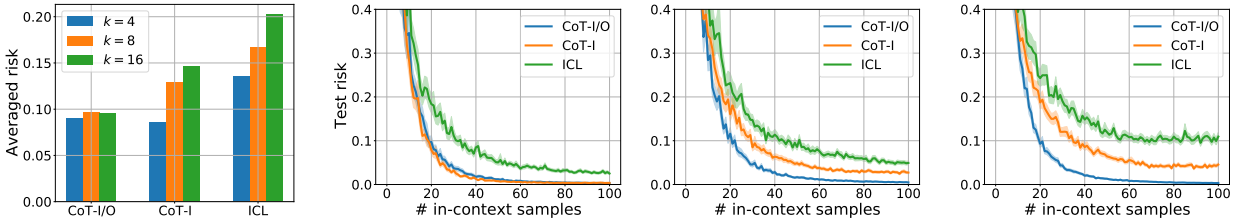
10.4.2 Comparative Analysis of ICL, CoT-I and CoT-I/O

Varying model sizes (Figure 10.5). We initially assess the benefits of CoT-I/O over ICL and CoT-I across different `TF` models. With $d = 10$ and $k = 8$ fixed, we train three different GPT-2 models: standard, small and tiny GPT-2. The small GPT-2 has 6 layers, 4 attention heads per layer and 128 dimensional embeddings. The standard GPT-2 consists of twice the



(a) Averaged risk (b) Standard GPT-2 (c) Small GPT-2 (d) Tiny GPT-2

Figure 10.5: Comparison of three methods for solving 2-layer MLPs using different GPT-2's.

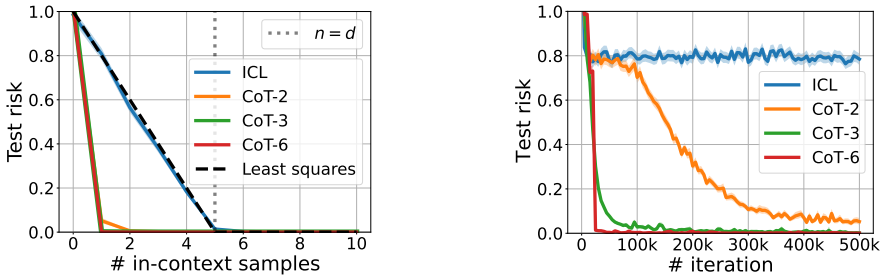


(a) Averaged risk (b) $k = 4$ (c) $k = 8$ (d) $k = 16$

Figure 10.6: Comparison of three methods for solving 2-layer MLPs with different widths.

number of layers, attention heads and embedding dimensionality compared to the small GPT-2, and tiny GPT-2, on the other hand, possesses only half of these hyperparameters compared to the small GPT-2. We evaluate the performance using prompts containing n in-context samples, where n ranges from 1 to N ($N = 100$). The associated test risks are displayed in Figs. 10.5b, 10.5c and 10.5d. The blue, orange and green curves correspond to CoT-I/O, CoT-I and ICL, respectively. In Fig. 10.5a, we present the averaged risks. The results show that using CoT-I/O, the small GPT-2 can solve 2-layer MLPs with approximately 60 samples, while CoT-I requires the standard GPT-2. Conversely, ICL is unable to achieve zero test risk even with the standard GPT-2 model and up to 100 samples. This indicates that to learn 2-layer MLPs in a single shot, ICL requires at least $\mathcal{O}(dk+d)$ samples to restore all function parameters. Conversely, CoT-I and CoT-I/O can leverage implicit samples contained in the CoT prompt. Let $f_1 \in \mathcal{F}_1$ (first layer) and $f_2 \in \mathcal{F}_2$ (second layer). By comparing the performance of CoT-I and CoT-I/O, it becomes evident that the standard GPT-2 is capable of learning the composed function $f = f_2 \circ f_1 \in \mathcal{F}$, which the small GPT-2 cannot express.

Varying MLP widths (Figure 10.6). Next, we explore how different MLP widths impact the performance (by varying the hidden neuron size $k \in \{4, 8, 16\}$). The corresponding results are depicted in Figure 10.6. The blue, orange and green curves in Fig. 10.6b, 10.6c and 10.6d correspond to hidden layer sizes of $k = 4$, 8, and 16, respectively. Fig. 10.6a



(a) Point-to-point meta prediction (b) One-shot prediction over time

Figure 10.7: Evaluations over deep linear MLPs using CoT-I/O and ICL where CoT-X represents the X-step CoT-I/O. Fig. 10.7a illustrates point-to-point meta results where the model is trained with substantial number of samples. In contrast, Fig. 10.7b displays the one-shot performance (with only one in-context sample provided) when making predictions during training.

displays the averaged risks. We keep $d = 10$, $N = 100$ fixed and train with the small GPT-2 model. As discussed in Section 10.4.1, CoT-I/O can learn a 2-layer MLP using around 60 samples for all $k = 4, 8, 16$ due to its capability to deconstruct composed functions. However, CoT-I can only learn the narrow MLPs with $k = 4$, and ICL is unable to learn any of them. Moreover, we observe a substantial difference in the performance of ICL and CoT-I with varying k (e.g., see averaged risks in Fig. 10.6a). This can be explained by the fact that enlarging k results in more complex $\mathcal{F}_1$ and $\mathcal{F}_2$, thus making the learning of $\mathcal{F} = \mathcal{F}_2 \times \mathcal{F}_1$ more challenging for ICL and CoT-I.

10.5 Further Investigation on Deep Linear MLPs

In Section 10.4, we have discussed the approximation benefits of CoT-I/O and how it in-context learns 2-layer random MLPs by parallel learning of k d-dimensional ReLU and 1 k-dimensional linear regression. In this section, we investigate the capability of CoT-I/O in learning longer compositions. For brevity, we will use CoT to refer to CoT-I/O in the rest of the discussion.

Dataset. Consider L-layer linear MLPs with input $\boldsymbol{x} \in \mathbb{R}^d \sim \mathcal{N}(0, \boldsymbol{I}_d)$, and output generated by $\boldsymbol{y} = \boldsymbol{W}_L \boldsymbol{W}_{L-1} \cdots \boldsymbol{W}_1 \boldsymbol{x}$, where the ℓth layer is parameterized by $\boldsymbol{W}_\ell \in \mathbb{R}^{d \times d}$, $\ell \in [L]$. In this work, to better understand the emerging ability of CoT, we assume that each layer draws from the same discrete sub-function set $\bar{\mathcal{F}} = \{\bar{\boldsymbol{W}}_k : \bar{\boldsymbol{W}}_k^\top \bar{\boldsymbol{W}}_k = \boldsymbol{I}, k \in [K]\}$[3]. Therefore, to learn the L-layer neural net, CoT only needs to learn $\bar{\mathcal{F}}$ with $|\bar{\mathcal{F}}| = K$, whereas ICL

[3]This assumption ensures that the norm of the feature remains constant across layers ($\|\boldsymbol{x}\| = \|\boldsymbol{y}\| = \|\boldsymbol{s}^\ell\|$), enabling fair evaluation across different layers.

needs to learn the function set $\bar{\mathcal{F}}^L$, which contains K^L random matrices.

Composition ability of CoT (Figures 10.7). Set $d = 5$, $L = 6$ and $K = 4$. At each round, we randomly select L matrices $\boldsymbol{W}_\ell$, $\ell \in [L]$ from $\bar{\mathcal{F}}$ so that for any input $\boldsymbol{x}$, we can form a chain

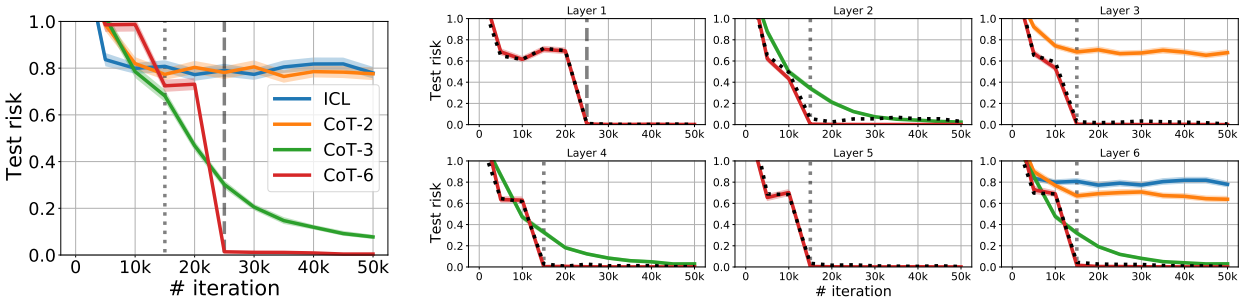
$$\boldsymbol{x} \to \boldsymbol{s}^1 \to \boldsymbol{s}^2 \cdots \to \boldsymbol{s}^6 := \boldsymbol{y},$$

where $\boldsymbol{s}^\ell = \boldsymbol{W}_\ell \boldsymbol{s}^{\ell-1}$, $\ell \in [L]$ and $\boldsymbol{s}^0 := \boldsymbol{x}$. Let CoT-$X$ denote X-step CoT-I/O method. For example, the in-context sample of CoT-6 has form of $(\boldsymbol{x}, \boldsymbol{s}^1, \boldsymbol{s}^2, \ldots \boldsymbol{s}^5, \boldsymbol{y})$, which contains all the intermediate outputs from each layer; while CoT-3, CoT-2 have prompt samples formed as $(\boldsymbol{x}, \boldsymbol{s}^2, \boldsymbol{s}^4, \boldsymbol{y})$ and $(\boldsymbol{x}, \boldsymbol{s}^3, \boldsymbol{y})$ respectively. In this setting, ICL is also termed as CoT-1, as its prompt contains only $(\boldsymbol{x}, \boldsymbol{y})$ pairs. To solve the length-6 chain, CoT-X needs to learn a model that can remember $4^{6/X}$ matrices. Therefore, ICL is face a significantly challenge sincd it needs to remember $4^6 = 4,096$ matrices (all combinations of the 4 matrices used for training and testing) compared to just 4 for CoT-6.

We train small GPT-2 models using the CoT-2/-3/-6 and ICL methods, and present the results in Fig. 10.7a. As evident from the figure, the performance curves of CoT-2 (orange), CoT-3 (green) and CoT-6 (red) overlap, and they can all make precise predictions in one shot (given an in-context example $n = 1$). It seems that `TF` has effectively learned to remember up to 64 matrices (for CoT-2) and compose up to 6 layers (for CoT-6). However, ICL (blue) struggles to learn the 6-layer MLPs in one shot. The black dashed curve shows the solution for linear regression $y = \boldsymbol{\beta}^\top \boldsymbol{x}$ computed directly via least squares given n random training samples, where $\boldsymbol{x}$ is the input and y is from the output of the 6-layer MLPs (e.g., $\boldsymbol{y}[0]$). The test risks for $n = 1, \ldots 10$ are plotted (in Fig. 10.7a), which show that the ICL curve aligns with the least squares performance. This implies that, instead of remembering all $4,096$ matrices, ICL solves the problem from the linear regression phase.

In addition to the meta-learning results which highlight the approximation benefits of multi-step CoT, we also investigate the convergence rate of CoT-2/-3/-6 and ICL, with results displayed in Fig. 10.7b. We test the one-shot performance during training and find that CoT-6 converges fastest. This is because it has the smallest sub-function set, and given the same tasks (e.g., deep neural nets), shortening the chain leads to slower convergence. This supports the evidence that taking more steps facilitates faster and more effective learning of complex problems.

Evidence of Filtering (Figure 10.8). As per Theorem 10.1 and the appendix, transformers can perform filtering over CoT prompts, and the results from 2-layer MLPs align with our theoretical findings. However, can we explicitly observe filtering behaviors? In



(a) Composed risk

(b) Risk of each layer output

Figure 10.8: Fig. 10.8a is generated by magnifying the initial 50k iterations of Fig. 10.7b, and we decouple the composed risks from predicting 6-layer linear MLPs into predictions for each layer, and the results are depicted in Fig. 10.8b. Additional implementation details can be found in Section 10.5.

Fig. 10.8a, we display the results of the first 50k iterations from Fig. 10.7b, and observe risk drops in CoT-6 (red) at the 15k and 25k iteration (shown as grey dotted and dashed lines). Subsequently, in Fig. 10.8b, we plot the test risk of each layer prediction (by feeding the model with correct intermediate features not the predicted ones), where CoT-6 (red) predicts the outputs from all 6 layers $(\boldsymbol{s}^1, \cdots, \boldsymbol{s}^L)$. From these figures, we can identify risk drops when predicting different layers, which appear at either 15k (for layer 2, 3, 4, 5, 6) or 25k (for layer 1) iteration. This implies that the model learns to predict each step/layer function independently. Further, we test the filtering evidence of the ℓth layer by filling irrelevant positions with random features. Specifically, an in-context example is formed by

$$(\boldsymbol{z}^0, \cdots, \boldsymbol{s}^{\ell-1}, \boldsymbol{s}^{\ell}, \boldsymbol{z}^{\ell+1}, \ldots \boldsymbol{z}^L), \quad \text{where} \quad \boldsymbol{s}^{\ell} = \boldsymbol{W}_{\ell}(\boldsymbol{s}^{\ell-1}) \quad \text{and} \quad \boldsymbol{z} \sim \mathcal{N}(0, \boldsymbol{I}_d).$$

The test risks are represented by black dotted curves in Fig. 10.8b, which aligned precisely with the CoT-6 curves (red). This signifies that each layer's prediction concentrate solely on the corresponding intermediate steps in the prefix, while disregarding irrelevant features. This observation provides evidence that the process of filtering is indeed performed.

CHAPTER 11

Conclusion and Future Work

The rapid advancement of artificial intelligence (AI) and machine learning (ML) has led to models capable of performing tasks once considered exclusive to human cognition, such as complex mathematical reasoning. While these models demonstrate impressive empirical performance across a wide range of applications, they often lack a comprehensive theoretical foundation. Many are developed through empirical trial and error, leaving fundamental questions about their underlying mechanisms unanswered.

In light of these challenges, this thesis aims to advance the theoretical understanding of language models, with a focus on their optimization, architecture, and emergent abilities. As a brief overview, this thesis presents the following investigations:

Chapters 2–4: Optimization Dynamics of Softmax Attention. The attention mechanism, a core component of Transformer-based language models, plays a crucial role in enabling parallelization over sequential data. However, the underlying reasons for its effectiveness remain underexplored. This thesis establishes a concrete theoretical framework to explain the optimization dynamics of softmax attention, revealing a surprising connection to a foundational concept in machine learning: the Support Vector Machine (SVM).

Chapters 5–8: Optimization Landscape of Alternative Recurrent Models. Despite the success of softmax attention, its computational and memory costs are significant. Recent alternatives (e.g., Mamba) offer efficient inference while maintaining competitive performance. This thesis analyzes the optimization Landscape of several such architectures, including Linear Attention, State Space Models, and Gated Linear Attention.

Chapters 9–10: Emergent Abilities of Language Models. LLMs exhibit emergent capabilities such as the ability to perform unseen tasks without explicit parameter updates. Beyond optimization analysis, this thesis investigates these emergent behaviors, including in-context learning and chain-of-thought reasoning, and offers theoretical explanations for how and why such behaviors arise.

This thesis opens several avenues for further research, which we outline below:

- While this thesis establishes the asymptotic convergence of gradient descent for single-layer softmax attention, a finer-grained non-asymptotic analysis remains an open question. Understanding convergence rates could offer valuable insights into the influence of learning rate, initialization, and optimization method. Moreover, the directional behavior of gradient descent from arbitrary initialization is not yet fully characterized. A key question is: *Where does gradient descent converge directionally for 1-layer self-attention models?* Investigating this could clarify the role of over-parameterization and further develop the notion of locally optimal directions, both of which appear to influence training behavior in practice.

- The scope of this thesis does not fully capture the complexity of ICL in large language models. In particular, it does not address more challenging settings such as question answering, where in-context demonstrations must be integrated with general knowledge acquired during pretraining. Future work could explore how specific contexts might activate or suppress behaviors learned during pretraining, including undesirable ones. Additionally, while this thesis introduces a theoretical model for retrieval-augmented ICL, real-world settings require accounting for the quality and relevance of retrieved demonstrations. Analyzing the effects of retrieval noise on final predictions is an important direction for extending this work.

- While our theoretical and experimental results validate the effectiveness of CoT, the current analysis is limited to in-distribution settings. Future work could extend this understanding to tasks with greater structural complexity, longer compositional chains, and more diverse subproblem types, including autoregressive reasoning and symbolic problem solving. Additionally, our proposed two-stage framework, filtering followed by in-context learning, offers an interpretable explanation of how CoT reduces in-context complexity, but its alignment with empirical behavior in real-world applications such as code generation and mathematical reasoning remains to be fully understood. We have shown that CoT can rely on a linear regression oracle to approximate an MLP, raising the important question of whether LLMs can achieve similar expressivity without CoT, and what the corresponding theoretical lower and upper bounds would be.

APPENDIX A

Appendix for Chapter 2

A.1 Proofs for Section 2.2

A.1.1 Preliminaries on the Training Risk

By our assumption $\psi : \mathbb{R}^d \to \mathbb{R}$ and $\ell : \mathbb{R} \to \mathbb{R}$ are differentiable functions. Recall the objective

$$\mathcal{L}(\boldsymbol{p}, \boldsymbol{W}) = \frac{1}{n} \sum_{i=1}^{n} \ell \left(y_i \cdot \psi(\boldsymbol{X}_i^\top \mathbb{S}(\boldsymbol{K}_i \boldsymbol{p})) \right) \tag{A.1}$$

with the generic prediction model $\psi(\boldsymbol{X}^\top \mathbb{S}(\boldsymbol{K}\boldsymbol{p}))$ and $\boldsymbol{K} = \boldsymbol{X}\boldsymbol{W}^\top$.

Here, we write down the gradients of $\boldsymbol{W}$ and $\boldsymbol{p}$ in (A.1) to highlight the connection. Set $\boldsymbol{q} := \boldsymbol{W}^\top \boldsymbol{p}$, $\boldsymbol{z}\{\boldsymbol{X}\} := \boldsymbol{X}^\top \mathbb{S}(\boldsymbol{K}\boldsymbol{p})$, and $\boldsymbol{a}\{\boldsymbol{X}\} := \boldsymbol{K}\boldsymbol{p}$. Given $\boldsymbol{X}$ and using $\boldsymbol{K} = \boldsymbol{X}\boldsymbol{W}^\top$, we have that

$$\nabla_{\boldsymbol{q}} \psi(\boldsymbol{p}, \boldsymbol{W}) = \boldsymbol{X}^\top \mathbb{S}'(\boldsymbol{a}\{\boldsymbol{X}\}) \boldsymbol{X} \cdot \psi'(\boldsymbol{z}\{\boldsymbol{X}\}), \tag{A.2a}$$

$$\nabla_{\boldsymbol{p}} \psi(\boldsymbol{p}, \boldsymbol{W}) = \boldsymbol{W} \nabla_{\boldsymbol{q}} \psi(\boldsymbol{p}, \boldsymbol{W}), \tag{A.2b}$$

$$\nabla_{\boldsymbol{W}} \psi(\boldsymbol{p}, \boldsymbol{W}) = \boldsymbol{p} \nabla_{\boldsymbol{q}}^\top \psi(\boldsymbol{p}, \boldsymbol{W}), \tag{A.2c}$$

where

$$\mathbb{S}'(\boldsymbol{a}\{\boldsymbol{X}\}) = \operatorname{diag}(\mathbb{S}(\boldsymbol{a}\{\boldsymbol{X}\})) - \mathbb{S}(\boldsymbol{a}\{\boldsymbol{X}\}) \mathbb{S}(\boldsymbol{a}\{\boldsymbol{X}\})^\top \in \mathbb{R}^{T \times T}.$$

Setting $\psi(\boldsymbol{z}) = \boldsymbol{v}^\top \boldsymbol{z}$ for linear head, we obtain

$$\nabla_{\boldsymbol{q}} \psi(\boldsymbol{p}, \boldsymbol{W}) = \boldsymbol{X}^\top \mathbb{S}'(\boldsymbol{a}\{\boldsymbol{X}\}) \gamma, \tag{A.3a}$$

$$\nabla_{\boldsymbol{p}} \psi(\boldsymbol{p}, \boldsymbol{W}) = \boldsymbol{W} \nabla_{\boldsymbol{q}} \psi(\boldsymbol{p}, \boldsymbol{W}) = \boldsymbol{K}^\top \mathbb{S}'(\boldsymbol{a}\{\boldsymbol{X}\}) \gamma, \tag{A.3b}$$

$$\nabla_{\boldsymbol{W}} \psi(\boldsymbol{p}, \boldsymbol{W}) = \boldsymbol{p} \nabla_{\boldsymbol{q}}^\top \psi(\boldsymbol{p}, \boldsymbol{W}) = \boldsymbol{p}\boldsymbol{v}^\top \boldsymbol{X}^\top \mathbb{S}'(\boldsymbol{a}\{\boldsymbol{X}\}) \boldsymbol{X}. \tag{A.3c}$$

Recalling (A.2b) and (A.2c), and defining $\ell_i' := \ell'(y_i \cdot \psi(\boldsymbol{z}\{\boldsymbol{X}_i\})) \in \mathbb{R}$, we have that

$$\nabla_{\boldsymbol{p}} \mathcal{L}(\boldsymbol{p}, \boldsymbol{W}) = \frac{1}{n} \sum_{i=1}^{n} \ell_i' \cdot y_i \cdot \boldsymbol{W} \nabla_{\boldsymbol{q}} \psi(\boldsymbol{p}, \boldsymbol{W}), \tag{A.4a}$$

$$\nabla_{\boldsymbol{W}} \mathcal{L}(\boldsymbol{p}, \boldsymbol{W}) = \frac{1}{n} \sum_{i=1}^{n} \ell_i' \cdot y_i \cdot \boldsymbol{p} \nabla_{\boldsymbol{q}}^{\top} \psi(\boldsymbol{p}, \boldsymbol{W}). \tag{A.4b}$$

Setting $\psi(\boldsymbol{z}) = \boldsymbol{v}^{\top} \boldsymbol{z}$ for linear head and $\gamma_i = y_i \cdot \boldsymbol{X}_i \boldsymbol{v}$, we obtain

$$\nabla_{\boldsymbol{p}} \mathcal{L}(\boldsymbol{p}, \boldsymbol{W}) = \frac{1}{n} \sum_{i=1}^{n} \ell_i' \cdot \boldsymbol{K}_i^{\top} \mathbb{S}'(\boldsymbol{a}\{\boldsymbol{X}_i\}) \gamma_i, \tag{A.5a}$$

$$\nabla_{\boldsymbol{W}} \mathcal{L}(\boldsymbol{p}, \boldsymbol{W}) = \boldsymbol{p} \left(\frac{1}{n} \sum_{i=1}^{n} \ell_i' \cdot \gamma_i^{\top} \mathbb{S}'(\boldsymbol{a}\{\boldsymbol{X}_i\}) \boldsymbol{X}_i \right). \tag{A.5b}$$

Lemma A.1 (Key Lemma) *For any $\boldsymbol{p}, \boldsymbol{q} \in \mathbb{R}^d$, let $\boldsymbol{a} = \boldsymbol{K}\boldsymbol{q}$, $\boldsymbol{s} = \mathbb{S}(\boldsymbol{K}\boldsymbol{p})$, and $\gamma = \boldsymbol{X}\boldsymbol{v}$. Set*

$$\Gamma = \sup_{t,\tau \in [T]} |\gamma_t - \gamma_\tau| \quad \text{and} \quad A = \sup_{t \in [T]} \|\boldsymbol{k}_t\| \cdot \|\boldsymbol{q}\|.$$

We have that

$$\left| \boldsymbol{a}^{\top} \mathrm{diag}(\boldsymbol{s}) \gamma - \boldsymbol{a}^{\top} \boldsymbol{s} \boldsymbol{s}^{\top} \gamma - \sum_{t \geq 2}^{T} (\boldsymbol{a}_1 - \boldsymbol{a}_t) \boldsymbol{s}_t (\gamma_1 - \gamma_t) \right| \leq 2\Gamma A (1 - \boldsymbol{s}_1)^2.$$

Proof. Set $\bar{\gamma} = \sum_{t=1}^{T} \gamma_t \boldsymbol{s}_t$. We have

$$\gamma_1 - \bar{\gamma} = \sum_{t \geq 2}^{T} (\gamma_1 - \gamma_t) \boldsymbol{s}_t, \quad \text{and} \quad |\gamma_1 - \bar{\gamma}| \leq \Gamma (1 - \boldsymbol{s}_1).$$

Then,

$$\begin{aligned} \boldsymbol{a}^{\top} \mathrm{diag}(\boldsymbol{s}) \gamma - \boldsymbol{a}^{\top} \boldsymbol{s} \boldsymbol{s}^{\top} \gamma &= \sum_{t=1}^{T} \boldsymbol{a}_t \gamma_t \boldsymbol{s}_t - \sum_{t=1}^{T} \boldsymbol{a}_t \boldsymbol{s}_t \sum_{t=1}^{T} \gamma_t \boldsymbol{s}_t \\ &= \boldsymbol{a}_1 \boldsymbol{s}_1 (\gamma_1 - \bar{\gamma}) - \sum_{t \geq 2}^{T} \boldsymbol{a}_t \boldsymbol{s}_t (\bar{\gamma} - \gamma_t). \end{aligned}$$

Since

$$\left| \sum_{t \geq 2}^{T} \boldsymbol{a}_t \boldsymbol{s}_t (\bar{\gamma} - \gamma_t) - \sum_{t \geq 2}^{T} \boldsymbol{a}_t \boldsymbol{s}_t (\gamma_1 - \gamma_t) \right| \leq A\Gamma (1 - \boldsymbol{s}_1)^2,$$

we obtain[1]

$$\begin{aligned}\boldsymbol{a}^\top \mathrm{diag}(\boldsymbol{s})\gamma - \boldsymbol{a}^\top \boldsymbol{s}\boldsymbol{s}^\top \gamma &= \boldsymbol{a}_1 \boldsymbol{s}_1(\gamma_1 - \bar{\gamma}) - \sum_{t\geq 2}^{T} \boldsymbol{a}_t \boldsymbol{s}_t(\gamma_1 - \gamma_t) \pm A\Gamma(1-\boldsymbol{s}_1)^2 \\ &= \boldsymbol{a}_1 \boldsymbol{s}_1 \sum_{t\geq 2}^{T}(\gamma_1 - \gamma_t)\boldsymbol{s}_t - \sum_{t\geq 2}^{T} \boldsymbol{a}_t \boldsymbol{s}_t(\gamma_1 - \gamma_t) \pm A\Gamma(1-\boldsymbol{s}_1)^2 \\ &= \sum_{t\geq 2}^{T}(\boldsymbol{a}_1 \boldsymbol{s}_1 - \boldsymbol{a}_t)\boldsymbol{s}_t(\gamma_1 - \gamma_t) \pm A\Gamma(1-\boldsymbol{s}_1)^2 \\ &= \sum_{t\geq 2}^{T}(\boldsymbol{a}_1 - \boldsymbol{a}_t)\boldsymbol{s}_t(\gamma_1 - \gamma_t) \pm 2A\Gamma(1-\boldsymbol{s}_1)^2.\end{aligned}$$

Here, $\pm$ on the right handside uses the fact that

$$\left|\sum_{t\geq 2}^{T}(\boldsymbol{a}_1 \boldsymbol{s}_1 - \boldsymbol{a}_1)\boldsymbol{s}_t(\gamma_1 - \gamma_t)\right| \leq (1-\boldsymbol{s}_1)A\Gamma \sum_{t\geq 2}^{T} \boldsymbol{s}_t = (1-\boldsymbol{s}_1)^2 A\Gamma.$$

■

A.1.2 Proof of Lemma 2.1

Proof. Let us prove the result for a general step size sequence $(\eta_t)_{t\geq 0}$. On the same training data $(y_i, \boldsymbol{X}_i)_{i=1}^n$, recall the objectives $\tilde{\mathcal{L}}(\boldsymbol{p}) = \frac{1}{n}\sum_{i=1}^n \ell(y_i \cdot \psi(\boldsymbol{X}_i^\top \mathbb{S}(\boldsymbol{X}_i \boldsymbol{p})))$ and $\mathcal{L}(\boldsymbol{W}) = \frac{1}{n}\sum_{i=1}^n \ell(y_i \cdot \psi(\boldsymbol{X}_i^\top \mathbb{S}(\boldsymbol{X}_i \boldsymbol{W}^\top \boldsymbol{u})))$. Suppose claim is true till iteration t. For iteration $t+1$, using $\boldsymbol{W}(t)^\top \boldsymbol{u} = \boldsymbol{p}(t)$, define and observe that

$$\begin{aligned}\boldsymbol{S}_i &= \mathbb{S}'(\boldsymbol{X}_i \boldsymbol{W}(t)^\top \boldsymbol{u}) = \mathbb{S}'(\boldsymbol{X}_i \boldsymbol{p}(t)), \\ \boldsymbol{s}_i &= \mathbb{S}(\boldsymbol{X}_i \boldsymbol{W}(t)^\top \boldsymbol{u}) = \mathbb{S}(\boldsymbol{X}_i \boldsymbol{p}(t)), \\ \boldsymbol{z}\{\boldsymbol{X}_i\} &= \boldsymbol{X}_i^\top \mathbb{S}(\boldsymbol{X}_i \boldsymbol{p}(t)) = \boldsymbol{X}_i^\top \mathbb{S}(\boldsymbol{X}_i \boldsymbol{W}(t)^\top \boldsymbol{u}),\end{aligned}$$

for all $i \in [n]$.

[1] For simplicity, we use $\pm$ on the right hand side to denote the upper and lower bounds.

Thus, using (A.4), we have that

$$\nabla\tilde{\mathcal{L}}(\boldsymbol{p}(t)) = \frac{1}{n}\sum_{i=1}^{n} \ell_i' \cdot y_i \cdot \boldsymbol{X}_i^\top \boldsymbol{S}_i \boldsymbol{X}_i \cdot \psi'(\boldsymbol{z}\{\boldsymbol{X}_i\}),$$
$$\nabla\mathcal{L}(\boldsymbol{W}(t)) = \boldsymbol{u}\left(\frac{1}{n}\sum_{i=1}^{n} \ell_i' \cdot y_i \cdot \boldsymbol{X}_i^\top \boldsymbol{S}_i \boldsymbol{X}_i \cdot \psi'(\boldsymbol{z}\{\boldsymbol{X}_i\})\right)^\top.$$

Consequently, we found that gradient is rank-1 with left singular space equal to $\boldsymbol{u}$, i.e.,

$$\nabla\mathcal{L}(\boldsymbol{W}(t)) = \boldsymbol{u}\nabla^\top\tilde{\mathcal{L}}(\boldsymbol{p}(t)).$$

Since $\boldsymbol{W}(t)$'s left singular space is guaranteed to be in $\boldsymbol{u}$ (including $\boldsymbol{W}(0)$ by initialization), we only need to study the right singular vector. Using the induction till t, this yields

$$\begin{aligned}\boldsymbol{W}(t+1)^\top\boldsymbol{u} &= \boldsymbol{W}(t)^\top\boldsymbol{u} - \eta_t\|\boldsymbol{u}\|^{-2}\nabla^\top\mathcal{L}(\boldsymbol{W}(t))\boldsymbol{u}\\ &= \boldsymbol{p}(t) - \eta_t\|\boldsymbol{u}\|^{-2}\boldsymbol{u}^\top\boldsymbol{u}\nabla\tilde{\mathcal{L}}(\boldsymbol{p}(t))\\ &= \boldsymbol{p}(t+1).\end{aligned}$$

This concludes the induction. ∎

A.2 Proofs for Section 2.3

A.2.1 Descent and Global Gradient Correlation Conditions

The lemma below identifies conditions under which $\boldsymbol{p}^{\mathrm{mm}\star}$ is a global descent direction for $\mathcal{L}(\boldsymbol{p})$.

Lemma A.2 *Suppose $\ell(\cdot)$ is a strictly decreasing differentiate loss function and Assumption 2.2 holds. Then, for all $\boldsymbol{p} \in \mathbb{R}^d$, the training loss* (ERM) *obeys $\langle\nabla\mathcal{L}(\boldsymbol{p}), \boldsymbol{p}^{mm\star}\rangle < 0$.*

Proof. Set

$$\gamma_i = y_i \cdot \boldsymbol{X}_i\boldsymbol{v}, \quad \boldsymbol{a}_i = \boldsymbol{K}_i\boldsymbol{p}, \quad \bar{\boldsymbol{a}}_i = \boldsymbol{K}_i\boldsymbol{p}^{\mathrm{mm}\star}, \quad \text{and} \quad \ell_i' = \ell'\left(\gamma_i^\top\mathbb{S}(\boldsymbol{K}_i\boldsymbol{p})\right). \tag{A.6}$$

Let us recall the gradient evaluated at $\boldsymbol{p}$ which is given by

$$\nabla\mathcal{L}(\boldsymbol{p}) = \frac{1}{n}\sum_{i=1}^{n} \ell_i' \cdot \boldsymbol{K}_i^\top\mathbb{S}'(\boldsymbol{a}_i)\gamma_i. \tag{A.7}$$

This implies that

$$\langle \nabla \mathcal{L}(\boldsymbol{p}), \boldsymbol{p}^{\mathrm{mm}\star} \rangle = \frac{1}{n} \sum_{i=1}^{n} \ell_i' \cdot \langle \bar{\boldsymbol{a}}_i, \mathbb{S}'(\boldsymbol{a}_i)\gamma_i \rangle \,. \tag{A.8}$$

To proceed, we will prove that individual summands are all strictly negative. To show that, without losing generality, let us focus on the first input and drop the subscript i for cleaner notation. This yields

$$\langle \bar{\boldsymbol{a}}, \mathbb{S}'(\boldsymbol{a})\gamma \rangle = \bar{\boldsymbol{a}}^\top \mathrm{diag}(\mathbb{S}(\boldsymbol{a}))\gamma - \bar{\boldsymbol{a}}^\top \mathbb{S}(\boldsymbol{a})\mathbb{S}(\boldsymbol{a})^\top \gamma. \tag{A.9}$$

Without losing generality, assume optimal token is the first one and γ_t is a constant for all $t \geq 2$.

To proceed, we will prove the following: Suppose $\gamma = \gamma_{t\geq 2}$ is constant, $\gamma_1, \bar{\boldsymbol{a}}_1$ are the largest indices of $\gamma, \bar{\boldsymbol{a}}$. Then, for any $\boldsymbol{s}$ obeying $\sum_{t\in[T]} \boldsymbol{s}_t = 1, \boldsymbol{s}_t \geq 0$, we have that $\bar{\boldsymbol{a}}^\top \mathrm{diag}(\boldsymbol{s})\gamma - \bar{\boldsymbol{a}}^\top \boldsymbol{s}\boldsymbol{s}^\top \gamma > 0$. To see this, we write

$$\begin{aligned}
\bar{\boldsymbol{a}}^\top \mathrm{diag}(\boldsymbol{s})\gamma - \bar{\boldsymbol{a}}^\top \boldsymbol{s}\boldsymbol{s}^\top \gamma &= \sum_{t=1}^{T} \bar{\boldsymbol{a}}_t \gamma_t \boldsymbol{s}_t - \sum_{t=1}^{T} \bar{\boldsymbol{a}}_t \boldsymbol{s}_t \sum_{t=1}^{T} \gamma_t \boldsymbol{s}_t \\
&= \left(\bar{\boldsymbol{a}}_1 \gamma_1 \boldsymbol{s}_1 + \gamma \sum_{t\geq 2}^{T} \bar{\boldsymbol{a}}_t \boldsymbol{s}_t \right) - \Big(\gamma_1 \boldsymbol{s}_1 + \gamma(1 - \boldsymbol{s}_1) \Big) \left(\bar{\boldsymbol{a}}_1 \boldsymbol{s}_1 + \sum_{t\geq 2}^{T} \bar{\boldsymbol{a}}_t \boldsymbol{s}_t \right) \\
&= \bar{\boldsymbol{a}}_1 (\gamma_1 - \gamma) \boldsymbol{s}_1 (1 - \boldsymbol{s}_1) + \Big(\gamma - (\gamma_1 \boldsymbol{s}_1 + \gamma(1 - \boldsymbol{s}_1)) \Big) \sum_{t\geq 2}^{T} \bar{\boldsymbol{a}}_t \boldsymbol{s}_t \\
&= \bar{\boldsymbol{a}}_1 (\gamma_1 - \gamma) \boldsymbol{s}_1 (1 - \boldsymbol{s}_1) - (\gamma_1 - \gamma) \boldsymbol{s}_1 \sum_{t\geq 2}^{T} \bar{\boldsymbol{a}}_t \boldsymbol{s}_t \\
&= (\gamma_1 - \gamma)(1 - \boldsymbol{s}_1) \boldsymbol{s}_1 \left[\bar{\boldsymbol{a}}_1 - \frac{\sum_{t\geq 2}^{T} \bar{\boldsymbol{a}}_t \boldsymbol{s}_t}{\sum_{t\geq 2}^{T} \boldsymbol{s}_t} \right].
\end{aligned} \tag{A.10}$$

To proceed, let $\boldsymbol{\gamma}_{\mathrm{gap}} = \gamma_1 - \gamma$ and $\boldsymbol{a}_{\mathrm{gap}} = \bar{\boldsymbol{a}}_1 - \max_{t\geq 2} \boldsymbol{a}_t$. With these, we obtain

$$\bar{\boldsymbol{a}}^\top \mathrm{diag}(\boldsymbol{s})\gamma - \bar{\boldsymbol{a}}^\top \boldsymbol{s}\boldsymbol{s}^\top \gamma \geq \boldsymbol{a}_{\mathrm{gap}} \boldsymbol{\gamma}_{\mathrm{gap}} \boldsymbol{s}_1 (1 - \boldsymbol{s}_1). \tag{A.11}$$

Note that

$$\begin{aligned} \boldsymbol{a}^i_{\text{gap}} &\geq \inf_{t \neq \text{opt}_i} (\boldsymbol{k}_{i\text{opt}_i} - \boldsymbol{k}_{it})^\top \boldsymbol{p}^{\text{mm}\star} \geq 1, \\ \boldsymbol{\gamma}^i_{\text{gap}} &= \inf_{t \neq \text{opt}_i} \gamma_{i\text{opt}_i} - \gamma_{it} > 0, \\ \boldsymbol{s}_{i1}(1 - \boldsymbol{s}_{i1}) &> 0. \end{aligned}$$

On the other hand, by our assumption $\ell_i' < 0$. Hence, infimum'ing (A.11) over all inputs, multiplying by ℓ_i' and using (A.8) give the desired result. ■

Lemma A.3 (Gradient Correlation Conditions) *Consider $n = 1$ and let $\boldsymbol{p}^{mm} = \boldsymbol{p}^{mm\star}$ be ($\boldsymbol{p}$-SVM) solution separating $\alpha = opt$ from remaining tokens of input $\boldsymbol{X}$. Suppose $\ell(\cdot)$ is a strictly decreasing differentiate loss function and Assumption 2.2 holds. For any choice of $\pi > 0$, there exists $R := R_\pi$ such that, for any $\boldsymbol{p}$ with $\|\boldsymbol{p}\| \geq R$, we have*

$$\left\langle \nabla \mathcal{L}(\boldsymbol{p}), \frac{\boldsymbol{p}}{\|\boldsymbol{p}\|} \right\rangle \geq (1 + \pi) \left\langle \nabla \mathcal{L}(\boldsymbol{p}), \frac{\boldsymbol{p}^{mm}}{\|\boldsymbol{p}^{mm}\|} \right\rangle.$$

Above, observe that as $R \to \infty$, we eventually get to set $\pi = 0$.

Proof. The proof is similar to Lemma A.2 at a high-level. However, we also need to account for the impact of $\boldsymbol{p}$ besides $\boldsymbol{p}^{\text{mm}}$ in the gradient correlation. The main goal is showing that $\boldsymbol{p}^{\text{mm}}$ is the near-optimal descent direction, thus, $\boldsymbol{p}$ cannot significantly outperform it.

To proceed, let $\bar{\boldsymbol{p}} = \|\boldsymbol{p}^{\text{mm}}\|\boldsymbol{p}/\|\boldsymbol{p}\|$, $M = \sup_t \|\boldsymbol{k}_t\|$, $\Theta = 1/\|\boldsymbol{p}^{\text{mm}}\|$, $\boldsymbol{s} = \mathbb{S}(\boldsymbol{K}\boldsymbol{p})$, $\boldsymbol{a} = \boldsymbol{K}\bar{\boldsymbol{p}}$, $\bar{\boldsymbol{a}} = \boldsymbol{K}\boldsymbol{p}^{\text{mm}}$. Without losing generality assume opt = 1. Set $\gamma = \gamma_{t\geq 2}$. Repeating the proof of Lemma A.2 yields

$$\begin{aligned} \langle \nabla \mathcal{L}(\boldsymbol{p}), \boldsymbol{p}^{\text{mm}} \rangle &= \ell' \cdot (\gamma_1 - \gamma)(1 - \boldsymbol{s}_1)\boldsymbol{s}_1 \left[\bar{\boldsymbol{a}}_1 - \frac{\sum_{t\geq 2}^T \bar{\boldsymbol{a}}_t \boldsymbol{s}_t}{\sum_{t\geq 2}^T \boldsymbol{s}_t} \right], \\ \langle \nabla \mathcal{L}(\boldsymbol{p}), \bar{\boldsymbol{p}} \rangle &= \ell' \cdot (\gamma_1 - \gamma)(1 - \boldsymbol{s}_1)\boldsymbol{s}_1 \left[\boldsymbol{a}_1 - \frac{\sum_{t\geq 2}^T \boldsymbol{a}_t \boldsymbol{s}_t}{\sum_{t\geq 2}^T \boldsymbol{s}_t} \right]. \end{aligned}$$

Given π, for sufficiently large R, we wish to show that

$$\boldsymbol{a}_1 - \frac{\sum_{t\geq 2}^T \boldsymbol{a}_t \boldsymbol{s}_t}{\sum_{t\geq 2}^T \boldsymbol{s}_t} \leq (1 + \pi) \cdot \left[\bar{\boldsymbol{a}}_1 - \frac{\sum_{t\geq 2}^T \bar{\boldsymbol{a}}_t \boldsymbol{s}_t}{\sum_{t\geq 2}^T \boldsymbol{s}_t} \right]. \tag{A.12}$$

We consider two scenarios.

Scenario 1: $\|\bar{\boldsymbol{p}} - \boldsymbol{p}^{\mathrm{mm}}\| \le \epsilon := \pi/(2M)$. In this scenario, for any token, we find that

$$|\boldsymbol{a}_t - \bar{\boldsymbol{a}}_t| = |\boldsymbol{k}_t^\top(\bar{\boldsymbol{p}} - \boldsymbol{p}^{\mathrm{mm}})| \le M\|\bar{\boldsymbol{p}} - \boldsymbol{p}^{\mathrm{mm}}\| \le M\epsilon.$$

Consequently, we obtain

$$\bar{\boldsymbol{a}}_1 - \frac{\sum_{t\ge 2}^T \bar{\boldsymbol{a}}_t \boldsymbol{s}_t}{\sum_{t\ge 2}^T \boldsymbol{s}_t} \ge \boldsymbol{a}_1 - \frac{\sum_{t\ge 2}^T \boldsymbol{a}_t \boldsymbol{s}_t}{\sum_{t\ge 2}^T \boldsymbol{s}_t} - 2M\epsilon = \boldsymbol{a}_1 - \frac{\sum_{t\ge 2}^T \boldsymbol{a}_t \boldsymbol{s}_t}{\sum_{t\ge 2}^T \boldsymbol{s}_t} - \pi.$$

Also noticing $\bar{\boldsymbol{a}}_1 - \frac{\sum_{t\ge 2}^T \bar{\boldsymbol{a}}_t \boldsymbol{s}_t}{\sum_{t\ge 2}^T \boldsymbol{s}_t} \ge 1$ (thanks to $\boldsymbol{p}^{\mathrm{mm}}$ satisfying ≥ 1 margin), this implies (A.12).
Scenario 2: $\|\bar{\boldsymbol{p}} - \boldsymbol{p}^{\mathrm{mm}}\| \ge \epsilon := \pi/(2M)$. In this scenario, for some $\nu = \nu(\epsilon)$ and $\tau \ge 2$, we have that

$$\bar{\boldsymbol{p}}^\top(\boldsymbol{k}_1 - \boldsymbol{k}_\tau) = \boldsymbol{a}_1 - \boldsymbol{a}_\tau \le 1 - 2\nu.$$

Here $\tau = \arg\max_{t\ge 2} \bar{\boldsymbol{p}}^\top \boldsymbol{k}_t$ denotes the nearest point to $\boldsymbol{k}_1$. Recall that $\boldsymbol{s} = \mathbb{S}(\bar{R}\boldsymbol{a})$ where $\bar{R} = \|\boldsymbol{p}\|/\|\boldsymbol{p}^{\mathrm{mm}}\|$. To proceed, split the tokens into two groups: Let $\mathcal{N}$ be the group of tokens obeying $\bar{\boldsymbol{p}}^\top(\boldsymbol{k}_1 - \boldsymbol{k}_t) \le 1 - \nu$ for $t \in \mathcal{N}$ and $[T] - \mathcal{N}$ be the rest. Observe that

$$\frac{\sum_{t\in[T]-\mathcal{N}} \boldsymbol{s}_t}{\sum_{t\ge 2}^T \boldsymbol{s}_t} \le \frac{\sum_{t\in[T]-\mathcal{N}} \boldsymbol{s}_t}{\boldsymbol{s}_\tau} \le T\frac{e^{\nu\bar{R}}}{e^{2\nu\bar{R}}} = Te^{-\bar{R}\nu}.$$

Set $\bar{M} = M/\Theta$ and note that $\|\boldsymbol{a}_t\| \le \|\boldsymbol{p}^{\mathrm{mm}}\| \cdot \|\boldsymbol{k}_t\| \le \bar{M}$. Using $\bar{\boldsymbol{p}}^\top(\boldsymbol{k}_1 - \boldsymbol{k}_t) \le 1 - \nu$ over $t \in \mathcal{N}$ and plugging in the above bound, we obtain

$$\begin{aligned}\frac{\sum_{t\ge 2}^T (\boldsymbol{a}_1 - \boldsymbol{a}_t)\boldsymbol{s}_t}{\sum_{t\ge 2}^T \boldsymbol{s}_t} &= \frac{\sum_{t\in\mathcal{N}} (\boldsymbol{a}_1 - \boldsymbol{a}_t)\boldsymbol{s}_t}{\sum_{t\ge 2}^T \boldsymbol{s}_t} + \frac{\sum_{t\in[T]-\mathcal{N}} (\boldsymbol{a}_1 - \boldsymbol{a}_t)\boldsymbol{s}_t}{\sum_{t\ge 2}^T \boldsymbol{s}_t} \\ &\le 1 - \nu + 2\bar{M}Te^{-\bar{R}\nu}.\end{aligned}$$

Using the fact that $\bar{\boldsymbol{a}}_1 - \frac{\sum_{t\ge 2}^T \bar{\boldsymbol{a}}_t \boldsymbol{s}_t}{\sum_{t\ge 2}^T \boldsymbol{s}_t} \ge 1$, the above implies (A.12) with $\pi' = 2\bar{M}Te^{-\bar{R}\nu} - \nu$. To proceed, choose $R_\pi = \nu^{-1}\Theta^{-1}\log(2\bar{M}T/\pi)$ to ensure $\pi' \le \pi$. ∎

The following lemma states the descent property of gradient descent for $\mathcal{L}(\boldsymbol{p})$ under Assumption 2.1. It is important to note that although the infimum of the optimization problem is $\mathcal{L}^*$, it is not achieved at any finite $\boldsymbol{p}$. Additionally, there are no finite critical points $\boldsymbol{p}$.

Lemma A.4 *Under Assumption 2.1, the function $\mathcal{L}(\boldsymbol{p})$ is L_p-smooth, where*

$$L_p := \frac{1}{n}\sum_{i=1}^{n}\left(M_0\|\boldsymbol{v}\|^2\|\boldsymbol{W}\|^2\|\boldsymbol{X}_i\|^4 + 3M_1\|\boldsymbol{v}\|\ \|\boldsymbol{W}\|^2\|\boldsymbol{X}_i\|^3\right). \tag{A.13}$$

Furthermore, if $\eta \le 1/L_p$, then, for any initialization $\boldsymbol{p}(0)$, with the GD sequence $\boldsymbol{p}(t+1) = \boldsymbol{p}(t) - \eta\nabla\mathcal{L}(\boldsymbol{p}(t))$, we have

$$\mathcal{L}(\boldsymbol{p}(t+1)) - \mathcal{L}(\boldsymbol{p}(t)) \le -\frac{\eta}{2}\left\|\nabla\mathcal{L}(\boldsymbol{p}(t))\right\|^2, \tag{A.14}$$

for all $t \ge 0$. This implies that

$$\sum_{t=0}^{\infty}\left\|\nabla\mathcal{L}\left(\boldsymbol{p}(t)\right)\right\|^2 < \infty, \quad \text{and} \quad \lim_{t\to\infty}\left\|\nabla\mathcal{L}\left(\boldsymbol{p}\left(t\right)\right)\right\|^2 = 0. \tag{A.15}$$

Proof. Recall that we defined $\gamma_i = y_i \cdot \boldsymbol{X}_i\boldsymbol{v}$ and $\boldsymbol{a}_i = \boldsymbol{K}_i\boldsymbol{p}$. The gradient of $\mathcal{L}(\boldsymbol{p})$ is given by

$$\nabla\mathcal{L}(\boldsymbol{p}) = \frac{1}{n}\sum_{i=1}^{n}\ell'\left(\gamma_i^\top\mathbb{S}(\boldsymbol{K}_i\boldsymbol{p})\right)\cdot\boldsymbol{K}_i^\top\mathbb{S}'(\boldsymbol{a}_i)\gamma_i.$$

Note that for any $\boldsymbol{p}\in\mathbb{R}^d$, the Jacobian of $\mathbb{S}(\boldsymbol{K}_i\boldsymbol{p})$ is given by

$$\frac{\partial\mathbb{S}(\boldsymbol{K}_i\boldsymbol{p})}{\partial\boldsymbol{p}} = \mathbb{S}'(\boldsymbol{K}_i\boldsymbol{p})\boldsymbol{K}_i = \left(\mathrm{diag}(\mathbb{S}(\boldsymbol{K}_i\boldsymbol{p})) - \mathbb{S}(\boldsymbol{K}_i\boldsymbol{p})\mathbb{S}(\boldsymbol{K}_i\boldsymbol{p})^\top\right)\boldsymbol{K}_i. \tag{A.16}$$

The Jacobian (A.16) together with the definition of the softmax function $\mathbb{S}(\cdot)$ implies that $\|\partial\mathbb{S}(\boldsymbol{K}_i\boldsymbol{p})/\partial\boldsymbol{p}\| \le \|\boldsymbol{K}_i\|$. Hence, for any $\boldsymbol{p},\dot{\boldsymbol{p}}\in\mathbb{R}^d$, we have

$$\|\mathbb{S}(\boldsymbol{K}_i\boldsymbol{p}) - \mathbb{S}(\boldsymbol{K}_i\dot{\boldsymbol{p}})\| \le \|\boldsymbol{K}_i\|\ \|\boldsymbol{p}-\dot{\boldsymbol{p}}\|, \tag{A.17a}$$

and

$$\begin{aligned}\|\mathbb{S}'(\boldsymbol{K}_i\boldsymbol{p}) - \mathbb{S}'(\boldsymbol{K}_i\dot{\boldsymbol{p}})\| &\le \|\mathrm{diag}(\mathbb{S}(\boldsymbol{K}_i\boldsymbol{p})) - \mathrm{diag}(\mathbb{S}(\boldsymbol{K}_i\dot{\boldsymbol{p}}))\| \\ &\quad + \left\|\mathbb{S}(\boldsymbol{K}_i\boldsymbol{p})\mathbb{S}(\boldsymbol{K}_i\boldsymbol{p})^\top - \mathbb{S}(\boldsymbol{K}_i\dot{\boldsymbol{p}})\mathbb{S}(\boldsymbol{K}_i\dot{\boldsymbol{p}})^\top\right\| \\ &\le 3\|\boldsymbol{K}_i\|\ \|\boldsymbol{p}-\dot{\boldsymbol{p}}\|. \end{aligned} \tag{A.17b}$$

Here, the last inequality uses the fact that $|ab-cd| \le |d||a-c| + |a||b-d|$.

Next, for any $\boldsymbol{p}, \dot{\boldsymbol{p}} \in \mathbb{R}^d$, we have

$$\begin{aligned}
\|\nabla\mathcal{L}(\boldsymbol{p}) - \nabla\mathcal{L}(\dot{\boldsymbol{p}})\| &\leq \frac{1}{n}\sum_{i=1}^{n} \left\|\ell'\left(\gamma_i^\top \mathbb{S}(\boldsymbol{K}_i\boldsymbol{p})\right)\cdot \boldsymbol{K}_i^\top \mathbb{S}'(\boldsymbol{K}_i\boldsymbol{p})\gamma_i - \ell'\left(\gamma_i^\top \mathbb{S}(\boldsymbol{K}_i\dot{\boldsymbol{p}})\right)\cdot \boldsymbol{K}_i^\top \mathbb{S}'(\boldsymbol{K}_i\dot{\boldsymbol{p}})\gamma_i\right\| \\
&\leq \frac{1}{n}\sum_{i=1}^{n} \left\|\boldsymbol{K}_i^\top \mathbb{S}'(\boldsymbol{K}_i\dot{\boldsymbol{p}})\gamma_i\right\| \ \left|\ell'\left(\gamma_i^\top \mathbb{S}(\boldsymbol{K}_i\boldsymbol{p})\right) - \ell'\left(\gamma_i^\top \mathbb{S}(\boldsymbol{K}_i\dot{\boldsymbol{p}})\right)\right| \\
&+ \frac{1}{n}\sum_{i=1}^{n} \left|\ell'(\gamma_i^\top \mathbb{S}(\boldsymbol{K}_i\boldsymbol{p}))\right| \ \left\|\boldsymbol{K}_i^\top \mathbb{S}'(\boldsymbol{K}_i\boldsymbol{p})\gamma_i - \boldsymbol{K}_i^\top \mathbb{S}'(\boldsymbol{K}_i\dot{\boldsymbol{p}})\gamma_i\right\| \\
&\leq \frac{1}{n}\sum_{i=1}^{n} M_0\|\gamma_i\|^2 \ \|\boldsymbol{K}_i\| \ \|\mathbb{S}(\boldsymbol{K}_i\boldsymbol{p}) - \mathbb{S}(\boldsymbol{K}_i\dot{\boldsymbol{p}})\| \\
&+ \frac{1}{n}\sum_{i=1}^{n} M_1\|\gamma_i\| \ \|\boldsymbol{K}_i\| \ \|\mathbb{S}'(\boldsymbol{K}_i\boldsymbol{p}) - \mathbb{S}'(\boldsymbol{K}_i\dot{\boldsymbol{p}})\|, \qquad \text{(A.18)}
\end{aligned}$$

where the second inequality follows from the fact that $|ab - cd| \leq |d||a - c| + |a||b - d|$ and the third inequality uses Assumption 2.1.

Substituting (A.17a) and (A.17b) into (A.18), we get

$$\begin{aligned}
\|\nabla\mathcal{L}(\boldsymbol{p}) - \nabla\mathcal{L}(\dot{\boldsymbol{p}})\| &\leq \frac{1}{n}\sum_{i=1}^{n} \left(M_0\|\gamma_i\|^2\|\boldsymbol{K}_i\|^2 + 3M_1\|\boldsymbol{K}_i\|^2\|\gamma_i\|\right)\|\boldsymbol{p} - \dot{\boldsymbol{p}}\| \\
&\leq \frac{1}{n}\sum_{i=1}^{n} \left(M_0\|\boldsymbol{v}\|^2\|\boldsymbol{W}\|^2\|\boldsymbol{X}_i\|^4 + 3M_1\|\boldsymbol{v}\|\|\boldsymbol{W}\|^2\|\boldsymbol{X}_i\|^3\right)\|\boldsymbol{p} - \dot{\boldsymbol{p}}\| \\
&\leq L_p\|\boldsymbol{p} - \dot{\boldsymbol{p}}\|,
\end{aligned}$$

where L_p is defined in (A.13).

The remaining proof follows standard gradient descent analysis (see e.g. [177, Lemma 10]). Since $\mathcal{L}(\boldsymbol{p})$ is L_p-smooth, we get

$$\begin{aligned}
\mathcal{L}(\boldsymbol{p}(t+1)) &\leq \mathcal{L}(\boldsymbol{p}(t)) + \nabla\mathcal{L}(\boldsymbol{p}(t))^\top(\boldsymbol{p}(t+1) - \boldsymbol{p}(t)) + \frac{L_p}{2}\|\boldsymbol{p}(t+1) - \boldsymbol{p}(t)\|^2 \\
&= \mathcal{L}(\boldsymbol{p}(t)) - \eta\|\nabla\mathcal{L}(\boldsymbol{p}(t))\|^2 + \frac{L_p\eta^2}{2}\|\nabla\mathcal{L}(\boldsymbol{p}(t))\|^2 \\
&= \mathcal{L}(\boldsymbol{p}(t)) - \eta\left(1 - \frac{L_p\eta}{2}\right)\|\nabla\mathcal{L}(\boldsymbol{p}(t))\|^2 \\
&\leq \mathcal{L}(\boldsymbol{p}(t)) - \frac{\eta}{2}\|\nabla\mathcal{L}(\boldsymbol{p}(t))\|^2,
\end{aligned}$$

where the last inequality follows from our assumption on the stepsize.

The above inequality implies that

$$\sum_{t=0}^{\infty} \|\nabla \mathcal{L}(\boldsymbol{p}(t))\|^2 \leq \frac{2}{\eta} \left(\mathcal{L}(\boldsymbol{p}(0)) - \mathcal{L}^*\right), \tag{A.19}$$

where the right hand side is upper bounded by a finite constant. This is because, by Assumption 2.1, $\mathcal{L}(\boldsymbol{p}(0)) < \infty$ and $\mathcal{L}^* \leq \mathcal{L}(\boldsymbol{p}(t))$, where $\mathcal{L}^*$ denotes the minimum objective.

Finally, (A.19) yields the expression (A.15). ∎

A.2.2 Local Gradient Correlation Conditions

Let $(\mathcal{T}_i)_{i=1}^n$ be the sets of all support indices per Definition 2.2. Let $\bar{\mathcal{T}}_i := [T] - \mathcal{T}_i - \{\alpha_i\}$ be the set of non-SVM-neighbor token indices for $i \in [n]$. Let $\mathcal{I} \subseteq [n]$ be the sample index set that contains samples with support for locally optimal tokens as per Definition 2.2, and $\bar{\mathcal{I}}$ be the sample index set without support tokens, that is

$$\mathcal{I} = \left\{ i \in [n] \;\middle|\; \mathcal{T}_i \neq \emptyset \right\}, \quad \text{and} \quad \bar{\mathcal{I}} = \left\{ i \in [n] \;\middle|\; \mathcal{T}_i = \emptyset \right\} = [n] - \mathcal{I}. \tag{A.20}$$

Let

$$\Theta = \frac{1}{\|\boldsymbol{p}^{\mathrm{mm}}\|}, \tag{A.21a}$$

$$\delta = \min\left\{ \frac{1}{2} \min_{i \in [n]} \min_{\tau \in \bar{\mathcal{T}}_i} \left((\boldsymbol{k}_{i\alpha_i} - \boldsymbol{k}_{i\tau})^\top \boldsymbol{p}^{\mathrm{mm}} - 1 \right), \frac{1}{4} \right\}, \tag{A.21b}$$

$$A = \max\left\{ \max_{i \in [n], t \in [T]} \frac{\|\boldsymbol{k}_{it}\|}{\Theta}, 1 \right\}, \tag{A.21c}$$

$$\mu(\delta) = \frac{\delta^2}{32 A^2}. \tag{A.21d}$$

When $\bar{\mathcal{T}}_i = \emptyset$ for all $i \in [n]$ (i.e. globally-optimal indices), we set $\delta = \infty$ as all non-neighbor related terms will disappear. Recall scores $\gamma_{it} = y_i \cdot \boldsymbol{v}^\top \boldsymbol{x}_{it}$. We define the global scalar

$$\Gamma = \sup_{i \in [n], t, \tau \in [T]} |\gamma_{it} - \gamma_{i\tau}|. \tag{A.22a}$$

We define the score gaps over $i \in \mathcal{I}$ and support indices:

$$\gamma_i^{\mathrm{gap}} = \gamma_{i\alpha_i} - \max_{t \in \mathcal{T}_i} \gamma_{it}, \qquad \bar{\gamma}_i^{\mathrm{gap}} = \gamma_{i\alpha_i} - \min_{t \in \mathcal{T}_i} \gamma_{it},$$
$$\gamma_{\min}^{\mathrm{gap}} = \min_{i \in \mathcal{I}} \gamma_i^{\mathrm{gap}}, \quad \text{and} \quad \bar{\gamma}_{\max}^{\mathrm{gap}} = \max_{i \in \mathcal{I}} \bar{\gamma}_i^{\mathrm{gap}}. \tag{A.22b}$$

Recall $\boldsymbol{s}_i = \mathbb{S}(\boldsymbol{K}_i \boldsymbol{p})$. We set

$$S_i = \sum_{t \in \mathcal{T}_i} \boldsymbol{s}_{it}, \quad \text{and} \quad Q_i = \sum_{\tau \in \bar{\mathcal{T}}_i} \boldsymbol{s}_{i\tau}.$$

Lemma A.5 *For any $\mu \le \mu(\delta)$ with $\mu(\delta)$ defined in* (A.21d)*, $\boldsymbol{q} \in \mathtt{cone}_\mu(\boldsymbol{p}^{mm})$ with $\|\boldsymbol{q}\| = \|\boldsymbol{p}^{mm}\|$, and for all $i \in [n]$, $t \in \mathcal{T}_i$, and $\tau \in \bar{\mathcal{T}}_i$, we have*

$$(\boldsymbol{k}_{it} - \boldsymbol{k}_{i\tau})^\top \boldsymbol{q} \ge \frac{3}{2}\delta > 0, \tag{A.23a}$$

$$(\boldsymbol{k}_{i\alpha_i} - \boldsymbol{k}_{i\tau})^\top \boldsymbol{q} \ge 1 + \frac{3}{2}\delta, \tag{A.23b}$$

$$1 + \frac{1}{2}\delta \ge (\boldsymbol{k}_{i\alpha_i} - \boldsymbol{k}_{it})^\top \boldsymbol{q} \ge 1 - \frac{1}{2}\delta, \tag{A.23c}$$

Proof. To show (A.23), we note that for any $\boldsymbol{q} \in \mathtt{cone}_\mu(\boldsymbol{p}^{\mathrm{mm}})$ such that $\|\boldsymbol{q}\| = \|\boldsymbol{p}^{\mathrm{mm}}\|$,

$$\begin{aligned}
\|\boldsymbol{q} - \boldsymbol{p}^{\mathrm{mm}}\| &= \sqrt{2\|\boldsymbol{p}^{\mathrm{mm}}\|^2 - 2\langle \boldsymbol{q}, \boldsymbol{p}^{\mathrm{mm}} \rangle} \\
&= \|\boldsymbol{p}^{\mathrm{mm}}\| \sqrt{2 - 2\left\langle \frac{\boldsymbol{q}}{\|\boldsymbol{p}^{\mathrm{mm}}\|}, \frac{\boldsymbol{p}^{\mathrm{mm}}}{\|\boldsymbol{p}^{\mathrm{mm}}\|} \right\rangle} \\
&\le \frac{1}{\Theta}\sqrt{2 - 2(1-\mu)} \\
&\le \frac{\delta}{4A\Theta}.
\end{aligned} \tag{A.24}$$

Here, the first inequality follows from (2.8), and the last equality uses (A.21d).

We will proceed to show a bound on $(\boldsymbol{k}_{it_1} - \boldsymbol{k}_{it_2})^\top (\boldsymbol{q} - \boldsymbol{p}^{\mathrm{mm}})$ for any $i \in [n]$ and any token indices $t_1, t_2 \in [T]$. For any token index $t \in [T]$, we have

$$\begin{aligned}
\left|\boldsymbol{k}_{it}^\top (\boldsymbol{q} - \boldsymbol{p}^{\mathrm{mm}})\right| &\le \|\boldsymbol{q} - \boldsymbol{p}^{\mathrm{mm}}\| \|\boldsymbol{k}_{it}\| \\
&\le \frac{\delta}{4A\Theta} \cdot \|\boldsymbol{k}_{it}\| \\
&\le \frac{\delta}{4A\Theta} \cdot \Theta A \\
&= \frac{\delta}{4}.
\end{aligned} \tag{A.25}$$

Here, the second inequality follows from (A.24); and the third inequality is obtained from (A.21c).

Hence, it follows from (A.25) that

$$\text{for all } \ t_1, t_2 \in [T] \ \text{ and any } \ i \in [n], \qquad \left|(\boldsymbol{k}_{it_1} - \boldsymbol{k}_{it_2})^\top (\boldsymbol{q} - \boldsymbol{p}^{\mathrm{mm}})\right| \le \frac{\delta}{2}. \tag{A.26}$$

Note that $\boldsymbol{p}^{\mathrm{mm}}$ is the solution of ($\boldsymbol{p}$-SVM). Hence, from the definition of δ in (A.21b), we have

$$\delta \le \frac{1}{2} \min_{i\in[n]} \min_{\tau\in\bar{\mathcal{T}}_i} \left((\boldsymbol{k}_{i\alpha_i} - \boldsymbol{k}_{i\tau})^\top \boldsymbol{p}^{\mathrm{mm}} - 1 \right) \le \frac{1}{2} \min_{i\in[n]} \min_{t\in\mathcal{T}_i} \min_{\tau\in\bar{\mathcal{T}}_i} (\boldsymbol{k}_{it} - \boldsymbol{k}_{i\tau})^\top \boldsymbol{p}^{\mathrm{mm}}.$$

This, together with (A.26), implies that for all $t \in \mathcal{T}_i$ and $\tau \in \bar{\mathcal{T}}_i$,

$$\begin{aligned}
(\boldsymbol{k}_{it} - \boldsymbol{k}_{i\tau})^\top \boldsymbol{q} &\ge (\boldsymbol{k}_{it} - \boldsymbol{k}_{i\tau})^\top \boldsymbol{p}^{\mathrm{mm}} + (\boldsymbol{k}_{it} - \boldsymbol{k}_{i\tau})^\top (\boldsymbol{q} - \boldsymbol{p}^{\mathrm{mm}}) \\
&\ge 2\delta - \frac{1}{2}\delta = \frac{3}{2}\delta, \\
(\boldsymbol{k}_{i\alpha_i} - \boldsymbol{k}_{i\tau})^\top \boldsymbol{q} &\ge (\boldsymbol{k}_{i\alpha_i} - \boldsymbol{k}_{i\tau})^\top \boldsymbol{p}^{\mathrm{mm}} + (\boldsymbol{k}_{i\alpha_i} - \boldsymbol{k}_{i\tau})^\top (\boldsymbol{q} - \boldsymbol{p}^{\mathrm{mm}}) \\
&\ge 1 + 2\delta - \frac{1}{2}\delta = 1 + \frac{3}{2}\delta, \\
\left| (\boldsymbol{k}_{i\alpha_i} - \boldsymbol{k}_{it})^\top \boldsymbol{q} - 1 \right| &= \left| (\boldsymbol{k}_{i\alpha_i} - \boldsymbol{k}_{it})^\top \boldsymbol{p}^{\mathrm{mm}} + (\boldsymbol{k}_{i\alpha_i} - \boldsymbol{k}_{it})^\top (\boldsymbol{q} - \boldsymbol{p}^{\mathrm{mm}}) - 1 \right| \\
&= \left| (\boldsymbol{k}_{i\alpha_i} - \boldsymbol{k}_{it})^\top (\boldsymbol{q} - \boldsymbol{p}^{\mathrm{mm}}) \right| \le \frac{1}{2}\delta.
\end{aligned}$$

This completes the proof.

■

In the following lemma, we demonstrate the existence of parameters $\mu > 0$ and $\bar{R}_\mu > 0$ such that when R_μ is sufficiently large, there are no stationary points within $\mathcal{C}_{\mu,\bar{R}_\mu}(\boldsymbol{p}^{\mathrm{mm}})$ defined in (2.8).

Lemma A.6 *Suppose Assumption 2.1 on the loss function ℓ holds. Let $\boldsymbol{\alpha} = (\alpha_i)_{i=1}^n$ be locally optimal tokens (Definition 2.2). Let $\boldsymbol{p}^{mm} = \boldsymbol{p}^{mm}(\boldsymbol{\alpha})$ denote the SVM solution via ($\boldsymbol{p}$-SVM). There exists $\mu \in (0, \mu(\delta)]$ such that for sufficiently large $\bar{R}_\mu$: For all $\boldsymbol{q} \in \mathtt{cone}_\mu(\boldsymbol{p}^{mm})$ with $\|\boldsymbol{q}\| = \|\boldsymbol{p}^{mm}\|$ and $\boldsymbol{p} \in \mathcal{C}_{\mu,\bar{R}_\mu}(\boldsymbol{p}^{mm})$, there exist dataset-dependent positive constants C, $\hat{C}$, and c such that*

$$C \cdot \frac{1}{n} \sum_{i\in\mathcal{I}} (1 - \boldsymbol{s}_{i\alpha_i}) \ge -\left\langle \nabla\mathcal{L}(\boldsymbol{p}), \boldsymbol{q} \right\rangle \ge c \cdot \frac{1}{n} \sum_{i\in\mathcal{I}} (1 - \boldsymbol{s}_{i\alpha_i}) > 0, \tag{A.27a}$$

$$\|\nabla\mathcal{L}(\boldsymbol{p})\| \le A\hat{C} \cdot \frac{1}{n} \sum_{i\in\mathcal{I}} (1 - \boldsymbol{s}_{i\alpha_i}), \tag{A.27b}$$

$$-\left\langle \frac{\boldsymbol{q}}{\|\boldsymbol{q}\|}, \frac{\nabla\mathcal{L}(\boldsymbol{p})}{\|\nabla\mathcal{L}(\boldsymbol{p})\|} \right\rangle \ge \frac{c}{\hat{C}} \cdot \frac{\Theta}{A} > 0. \tag{A.27c}$$

Here, $\boldsymbol{s}_{i\alpha_i} = (\mathbb{S}(\boldsymbol{K}_i\boldsymbol{p}))_{\alpha_i}$; $\mathcal{I} \subseteq [n]$ is defined in (A.20); Θ and A are defined in (A.21).

Proof. Let $R := \|\boldsymbol{p}\|$,

$$\ell_i' = \ell'(y_i \cdot \boldsymbol{v}^\top \boldsymbol{X}_i^\top \mathbb{S}(\boldsymbol{K}_i\boldsymbol{p})), \quad \boldsymbol{a}_i = \boldsymbol{K}_i\boldsymbol{q}, \quad \tilde{\boldsymbol{a}}_i = \boldsymbol{K}_i\boldsymbol{p}, \quad \boldsymbol{s}_i = \mathbb{S}(\tilde{\boldsymbol{a}}_i). \tag{A.28}$$

Before the main proof of the lemma, we provide bounds for $|\boldsymbol{a}_{it}|$, Q_i, S_i, and Q_i/S_i. Note that using (A.21c), for all $\boldsymbol{q} \in \mathbb{R}^{d\times d}$ with $\|\boldsymbol{q}\| = \|\boldsymbol{p}^{\mathrm{mm}}\|$, we have

$$\max_{i\in[n],\ t\in[T]}\{|\boldsymbol{a}_{it}|\} \le \max_{i\in[n],\ t\in[T]}\{\|\boldsymbol{q}\|\ \|\boldsymbol{k}_{it}\|\} \le A. \tag{A.29}$$

Here, the last inequality uses (A.23c).

Assume, without loss of generality, $\alpha_i = 1$ for all $i \in [n]$. Let Θ, δ, and $\mu(\delta)$ be defined in (A.21a), (A.21b), and (A.21d), respectively. Choose $\mu \in (0, \mu(\delta)]$ so that it will satisfy the requirement of Lemma A.5. Note that $\mathcal{T}_i$ can be empty for some $i \in [n]$. In this case, we set $S_i = 0$. Similarly, $\bar{\mathcal{T}}_i$ can also be empty for some $i \in [n]$, in which case we set $Q_i = 0$. Note that for all $\boldsymbol{p} \in \mathtt{cone}_\mu(\boldsymbol{p}^{\mathrm{mm}})$, $\|\boldsymbol{p}^{\mathrm{mm}}\|\boldsymbol{p}/\|\boldsymbol{p}\|$ satisfies the requirement of Lemma A.5. Hence, we can provide bounds for Q_i and S_i in the following. If Q_i is positive, for any $\boldsymbol{p} \in \mathtt{cone}_\mu(\boldsymbol{p}^{\mathrm{mm}})$ and with $\bar{\mathcal{T}}_i$ as the set of all non-support indices, we have

$$Q_i = \frac{\sum_{t\in\bar{\mathcal{T}}_i} e^{\boldsymbol{k}_{it}^\top \boldsymbol{p}}}{\sum_{t=1}^{T} e^{\boldsymbol{k}_{it}^\top \boldsymbol{p}}} \le \frac{\sum_{t\in\bar{\mathcal{T}}_i} e^{\boldsymbol{k}_{it}^\top \boldsymbol{p}}}{e^{\boldsymbol{k}_{i1}^\top \boldsymbol{p}}} \le T e^{-R\Theta\left(1+\frac{3}{2}\delta\right)}. \tag{A.30a}$$

Here, the last inequality uses our definition $R = \|\boldsymbol{p}\|$ for all $\boldsymbol{p} \in \mathtt{cone}_\mu(\boldsymbol{p}^{\mathrm{mm}})$, $\Theta = 1/\|\boldsymbol{p}^{\mathrm{mm}}\|$, and (A.23b).

Similarly, in the case S_i is positive, for any $\boldsymbol{p} \in \mathtt{cone}_\mu(\boldsymbol{p}^{\mathrm{mm}})$ and $\mathcal{T}_i$ as the set of all support indices, we have

$$S_i = \frac{\sum_{t\in\mathcal{T}_i} e^{\boldsymbol{k}_{it}^\top \boldsymbol{p}}}{\sum_{t=1}^{T} e^{\boldsymbol{k}_{it}^\top \boldsymbol{p}}} \ge \frac{\sum_{t\in\mathcal{T}_i} e^{\boldsymbol{k}_{it}^\top \boldsymbol{p}}}{T e^{\boldsymbol{k}_{i1}^\top \boldsymbol{p}}} \ge \frac{1}{T} e^{-R\Theta(1+\frac{1}{2}\delta)}, \tag{A.30b}$$

$$S_i = \frac{\sum_{t\in\mathcal{T}_i} e^{\boldsymbol{k}_{it}^\top \boldsymbol{p}}}{\sum_{t=1}^{T} e^{\boldsymbol{k}_{it}^\top \boldsymbol{p}}} \le \frac{\sum_{t\in\mathcal{T}_i} e^{\boldsymbol{k}_{it}^\top \boldsymbol{p}}}{e^{\boldsymbol{k}_{i1}^\top \boldsymbol{p}}} \le T e^{-R\Theta(1-\frac{1}{2}\delta)}. \tag{A.30c}$$

Here, the last inequalities in (A.30b) and (A.30c) follow from (A.23c).

Further, for positive S_i and any $\boldsymbol{p} \in \mathtt{cone}_\mu(\boldsymbol{p}^{\mathrm{mm}})$, from (A.23a), we have

$$\frac{Q_i}{S_i} = \frac{\sum_{t\in\bar{\mathcal{T}}_i} e^{\boldsymbol{k}_{it}^\top \boldsymbol{p}}}{\sum_{t\in\mathcal{T}_i} e^{\boldsymbol{k}_{it}^\top \boldsymbol{p}}} \le T e^{-\frac{3}{2}R\Theta\delta}. \tag{A.30d}$$

Proof of (A.27a):

To proceed, we write the gradient correlation following (A.7) and (A.10) as

$$
\begin{aligned}
-\langle \nabla \mathcal{L}(\boldsymbol{p}), \boldsymbol{q} \rangle &= \frac{1}{n} \sum_{i=1}^{n} (-\ell_i') \cdot \boldsymbol{a}_i^\top \mathbb{S}'(\tilde{\boldsymbol{a}}_i) \gamma_i, \\
&= \frac{1}{n} \sum_{i \in \mathcal{I}} (-\ell_i') \cdot \boldsymbol{a}_i^\top \mathbb{S}'(\tilde{\boldsymbol{a}}_i) \gamma_i + \frac{1}{n} \sum_{i \in \bar{\mathcal{I}}} (-\ell_i') \cdot \boldsymbol{a}_i^\top \mathbb{S}'(\tilde{\boldsymbol{a}}_i) \gamma_i. \qquad \text{(A.31)}
\end{aligned}
$$

Directly applying Lemma A.1, for any $i \in [n]$, we obtain

$$
\left| \boldsymbol{a}_i^\top \operatorname{diag}(\boldsymbol{s}_i) \gamma_i - \boldsymbol{a}_i^\top \boldsymbol{s}_i \boldsymbol{s}_i^\top \gamma_i - \sum_{t \geq 2}^{T} (\boldsymbol{a}_{i1} - \boldsymbol{a}_{it}) \boldsymbol{s}_{it} (\gamma_{i1} - \gamma_{it}) \right| \leq 2\Gamma A (1 - \boldsymbol{s}_{i1})^2, \qquad \text{(A.32a)}
$$

where Γ is defined in (A.22a).

To proceed, let us decouple the non-support indices within $\sum_{t \geq 2}^{T} (\boldsymbol{a}_{i1} - \boldsymbol{a}_{it}) \boldsymbol{s}_{it} (\gamma_{i1} - \gamma_{it})$. Recall $\bar{\mathcal{T}}_i$ is the set of all non-support indices per Definition 2.2. From (A.29), we have

$$
\left| \sum_{t \in \bar{\mathcal{T}}_i} (\boldsymbol{a}_{i1} - \boldsymbol{a}_{it}) \boldsymbol{s}_{it} (\gamma_{i1} - \gamma_{it}) \right| \leq 2 Q_i \Gamma A. \qquad \text{(A.32b)}
$$

It follows from (A.32a) and (A.32b) for $\mathcal{T}_i$ as the set of all support indices,

$$
\left| \boldsymbol{a}_i^\top \operatorname{diag}(\boldsymbol{s}_i) \gamma_i - \boldsymbol{a}_i^\top \boldsymbol{s}_i \boldsymbol{s}_i^\top \gamma_i - \sum_{t \in \mathcal{T}_i} (\boldsymbol{a}_{i1} - \boldsymbol{a}_{it}) \boldsymbol{s}_{it} (\gamma_{i1} - \gamma_{it}) \right| \leq 2\Gamma A \left((1 - \boldsymbol{s}_{i1})^2 + Q_i \right). \qquad \text{(A.33)}
$$

Recall $\boldsymbol{a}_{i1} - \boldsymbol{a}_{it} = (\boldsymbol{k}_{i1} - \boldsymbol{k}_{it})^\top \boldsymbol{q}$ for all $\boldsymbol{q} \in \texttt{cone}_\mu(\boldsymbol{p}^{\mathrm{mm}})$ with $\|\boldsymbol{q}\| = \|\boldsymbol{p}^{\mathrm{mm}}\|$. Hence, it follows from (A.23c) that

$$
\left(1 + \frac{\delta}{2}\right) \sum_{t \in \mathcal{T}_i} \boldsymbol{s}_{it} (\gamma_{i1} - \gamma_{it}) \geq \sum_{t \in \mathcal{T}_i} (\boldsymbol{a}_{i1} - \boldsymbol{a}_{it}) \, \boldsymbol{s}_{it} \, (\gamma_{i1} - \gamma_{it}) \geq \left(1 - \frac{\delta}{2}\right) \sum_{t \in \mathcal{T}_i} \boldsymbol{s}_{it} (\gamma_{i1} - \gamma_{it}). \qquad \text{(A.34)}
$$

Next we consider bounding (A.34) for samples from $\mathcal{I}$ and $\bar{\mathcal{I}}$, separably.

Case 1: $i \in \mathcal{I}$ (Samples with support tokens). From (A.22b), we get

$$
\left(1 + \frac{\delta}{2}\right) \bar{\gamma}_i^{\mathrm{gap}} \sum_{t \in \mathcal{T}_i} \boldsymbol{s}_{it} \geq \sum_{t \in \mathcal{T}_i} (\boldsymbol{a}_{i1} - \boldsymbol{a}_{it}) \boldsymbol{s}_{it} (\gamma_{i1} - \gamma_{it}) \geq \left(1 - \frac{\delta}{2}\right) \gamma_i^{\mathrm{gap}} \sum_{t \in \mathcal{T}_i} \boldsymbol{s}_{it}. \qquad \text{(A.35)}
$$

Here, the inequalities follow from (A.22b) and (A.23c).

Note that $S_i = \sum_{t \in \mathcal{T}_i} \boldsymbol{s}_{it}$. The above inequality together with (A.23) implies that

$$\boldsymbol{a}_i^\top \operatorname{diag}(\boldsymbol{s}_i)\gamma_i - \boldsymbol{a}_i^\top \boldsymbol{s}_i \boldsymbol{s}_i^\top \gamma_i \le \left(1+\frac{\delta}{2}\right) S_i \bar{\gamma}_i^{\text{gap}} + 2\Gamma A\left((1-\boldsymbol{s}_{i1})^2 + Q_i\right), \tag{A.36a}$$

$$\boldsymbol{a}_i^\top \operatorname{diag}(\boldsymbol{s}_i)\gamma_i - \boldsymbol{a}_i^\top \boldsymbol{s}_i \boldsymbol{s}_i^\top \gamma_i \ge \left(1-\frac{\delta}{2}\right) S_i \gamma_i^{\text{gap}} - 2\Gamma A\left((1-\boldsymbol{s}_{i1})^2 + Q_i\right). \tag{A.36b}$$

We claim that for each sample $i \in \mathcal{I}$, S_i dominates $((1-\boldsymbol{s}_{i1})^2 + Q_i)$ for large R. Specifically, we aim for

$$\left(1-\frac{\delta}{2}\right) S_i \cdot \gamma_i^{\text{gap}} \ge 8\Gamma A \max\left((1-\boldsymbol{s}_{i1})^2, Q_i\right), \tag{A.37a}$$

which holds if and only if

$$S_i \ge 8\left(1-\frac{\delta}{2}\right)\frac{\Gamma A}{\gamma_i^{\text{gap}}} \max\left((1-\boldsymbol{s}_{i1})^2, Q_i\right). \tag{A.37b}$$

It follows from (A.30),

$$R \ge R_1 := \frac{2\log(T)}{3\delta\Theta}, \quad \text{implies that} \quad Q_i \le S_i \qquad \text{for all } i \in \mathcal{I}. \tag{A.38}$$

Consequently,

$$(1-\boldsymbol{s}_{i1})^2 = (Q_i + S_i)^2 \le 4S_i^2 \le 4S_i T e^{-R\Theta\left(1-\frac{1}{2}\delta\right)}. \tag{A.39}$$

Combining these, our goal is achieved by ensuring

$$S_i \ge 8\left(1-\frac{\delta}{2}\right)\frac{\Gamma A}{\gamma_i^{\text{gap}}} \max\left(4S_i T e^{-R\Theta(1-\frac{1}{2}\delta)}, S_i T e^{-R\Theta\frac{3}{2}\delta}\right), \tag{A.40}$$

or equivalently,

$$1 \ge 32\left(1-\frac{\delta}{2}\right)\frac{\Gamma A}{\gamma_i^{\text{gap}}} T e^{-R\Theta\frac{3}{2}\delta}.$$

This in turn is ensured for all inputs $i \in \mathcal{I}$ by choosing

$$R \ge \max(R_1, R_2), \quad \text{where} \quad R_2 := \frac{2}{3\delta\Theta}\log\left(\frac{16\,(2-\delta)\,T\Gamma A}{\gamma_{\min}^{\text{gap}}}\right). \tag{A.41}$$

Here, R_1 is defined in (A.38), and $\gamma_{\min}^{\text{gap}} = \min_{i\in\mathcal{I}} \gamma_i^{\text{gap}}$ is defined in (A.22b) as the global scalar which is the worst case score gap over all inputs.

With the choice of R in (A.41), we guaranteed

$$2\left(1+\frac{\delta}{2}\right)S_i\bar{\gamma}_i^{\text{gap}} \geq \boldsymbol{a}_i^\top \text{diag}(\boldsymbol{s}_i)\gamma_i - \boldsymbol{a}_i^\top \boldsymbol{s}_i\boldsymbol{s}_i^\top\gamma_i \geq \frac{1}{2}\left(1-\frac{\delta}{2}\right)S_i\gamma_i^{\text{gap}}. \tag{A.42}$$

Here, the first inequality follows since $Q_i \leq S_i$ by the choice of R in (A.41) and the last inequality is due to (A.37).

Let $-\ell'_{\min}$ and $-\ell'_{\max}$ be the min and max values negative loss derivative admits over the ball $[-A, A]$, respectively. Note that $\max_{i\in\mathcal{I}}\bar{\gamma}_i^{\text{gap}} > 0$ and $\min_{i\in\mathcal{I}}\gamma_i^{\text{gap}} > 0$ are dataset dependent constants. Then, using (A.22b), (A.42), and the choice of R in (A.41), the first term in (A.31) can be bounded as

$$(-\ell'_{\max})\cdot\frac{(2+\delta)\bar{\gamma}_{\max}^{\text{gap}}}{n}\sum_{i\in\mathcal{I}}S_i \geq -\frac{1}{n}\sum_{i\in\mathcal{I}}\ell'_i\cdot\boldsymbol{a}_i^\top\mathbb{S}'(\tilde{\boldsymbol{a}}_i)\gamma_i \geq (-\ell'_{\min})\cdot\frac{(2-\delta)\gamma_{\min}^{\text{gap}}}{4n}\sum_{i\in\mathcal{I}}S_i. \tag{A.43}$$

Case 2: $i \in \bar{\mathcal{I}}$ (Samples without support tokens). We next bound the second term in (A.31) by showing that it is dominated by the first term in (A.31) for large R. Note that if $i \in \bar{\mathcal{I}}$, then $S_i = 0$ and it follows from (A.36) that for all $i \in \bar{\mathcal{I}}$,

$$\left|\boldsymbol{a}_i^\top \text{diag}(\boldsymbol{s}_i)\gamma_i - \boldsymbol{a}_i^\top \boldsymbol{s}_i\boldsymbol{s}_i^\top\gamma_i\right| \leq 2\Gamma A\left((1-\boldsymbol{s}_{i1})^2 + Q_i\right) = 2\Gamma A\left(Q_i^2 + Q_i\right) \leq 4\Gamma A Q_i, \tag{A.44}$$

where the last inequality uses $(1-\boldsymbol{s}_{i1})^2 = (Q_i+S_i)^2 = Q_i^2 \leq Q_i$.

Recall $-\ell'_{\max}$ be the max values negative loss derivative admits over the ball $[-A, A]$. It follows from (A.31) that

$$(-\ell'_{\max})\cdot\frac{4\Gamma A}{n}\sum_{i\in\bar{\mathcal{I}}}Q_i \geq -\frac{1}{n}\sum_{i\in\bar{\mathcal{I}}}\ell'_i\cdot\boldsymbol{a}_i^\top\mathbb{S}'(\tilde{\boldsymbol{a}}_i)\gamma_i \geq \ell'_{\max}\cdot\frac{4\Gamma A}{n}\sum_{i\in\bar{\mathcal{I}}}Q_i. \tag{A.45}$$

Using (A.43) and (A.45), we will show that

$$(-\ell'_{\min})\cdot\frac{(2-\delta)\gamma_{\min}^{\text{gap}}}{4n}\sum_{i\in\mathcal{I}}S_i + \ell'_{\max}\cdot\frac{4\Gamma A}{n}\sum_{i\in\bar{\mathcal{I}}}Q_i \geq 0. \tag{A.46}$$

The above inequality together with (A.30a) and (A.30b) implies that

$$-\frac{(2-\delta)\gamma_{\min}^{\text{gap}}\ell'_{\min}}{8}|\mathcal{I}|\cdot\frac{1}{T}e^{-R\Theta(1+\frac{1}{2}\delta)} \geq -4\Gamma A\ell'_{\max}|\bar{\mathcal{I}}|\cdot Te^{-R\Theta(1+\frac{3}{2}\delta)}.$$

This gives the following choice of R

$$R \geq R_3 := \frac{1}{\delta\Theta} \log\left(\frac{32|\bar{\mathcal{I}}|T^2\Gamma A\ell'_{\max}}{(2-\delta)|\mathcal{I}|\gamma^{\text{gap}}_{\min}\ell'_{\min}}\right). \tag{A.47}$$

Taking both *Case 1* and *Case 2*, and combining (A.41) and (A.47), by choosing

$$R \geq \max\left(R_1, R_2, R_3\right), \tag{A.48}$$

we obtain

$$\begin{aligned}
-\frac{2(2+\delta)\gamma^{\text{gap}}_{\max}\ell'_{\max}}{n}\sum_{i\in\mathcal{I}} S_i &\geq -\frac{1}{n}\sum_{i\in\mathcal{I}} \ell'_i \cdot \boldsymbol{a}_i^\top \mathbb{S}'(\tilde{\boldsymbol{a}}_i)\gamma_i - \frac{1}{n}\sum_{i\in\bar{\mathcal{I}}} \ell'_i \cdot \boldsymbol{a}_i^\top \mathbb{S}'(\tilde{\boldsymbol{a}}_i)\gamma_i \\
&\geq -\frac{(2-\delta)\gamma^{\text{gap}}_{\min}\ell'_{\min}}{8n}\sum_{i\in\mathcal{I}} S_i.
\end{aligned}$$

The above inequality together with (A.31) implies that

$$-2(2+\delta)\gamma^{\text{gap}}_{\max}\ell'_{\max}\cdot\frac{1}{n}\sum_{i\in\mathcal{I}}(1-\boldsymbol{s}_{i1}) \geq \langle -\nabla\mathcal{L}(\boldsymbol{p}), \boldsymbol{q}\rangle \geq -\frac{(2-\delta)\gamma^{\text{gap}}_{\min}\ell'_{\min}}{16}\cdot\frac{1}{n}\sum_{i\in\mathcal{I}}(1-\boldsymbol{s}_{i1}). \tag{A.49}$$

Here, the first inequality follows since $(1-\boldsymbol{s}_{i1}) \geq S_i$ and the last inequality is obtained from the fact that $(1-\boldsymbol{s}_{i1})/2 = (Q_i + S_i)/2 \leq S_i$ due to (A.30d) and the choice of R in (A.41).

Note that since $\sum_{i\in\mathcal{I}}(1-s_{i1}) > 0$, the right-hand side of (A.49) is positive. Hence, by setting $R = \bar{R}_\mu$, there is no stationary point within $\mathcal{C}_{\mu,\bar{R}_\mu}(\boldsymbol{p})$. Finally, we declare the constants $C = -2(2+\delta)\gamma^{\text{gap}}_{\max}\ell'_{\max} > 0$ and $c = -(1/16)(2-\delta)\gamma^{\text{gap}}_{\min}\ell'_{\min} > 0$ to obtain (A.27a).

Proof of (A.27b) **and** (A.27c):

Let $\hat{\boldsymbol{a}}_i = \boldsymbol{K}_i\hat{\boldsymbol{q}}$ for all $\hat{\boldsymbol{q}} \in \mathbb{R}^d$ with $\|\hat{\boldsymbol{q}}\| = \|\boldsymbol{p}^{\text{mm}}\|$. Note that similar to (A.29), we have

$$\max_{i\in[n],t\in[T]}\{|\hat{\boldsymbol{a}}_{it}|\} \leq \max_{i\in[n],t\in[T]}\{\|\hat{\boldsymbol{q}}\|\,\|\boldsymbol{k}_{it}\|\} \leq A. \tag{A.50}$$

Recall $\boldsymbol{s}_i = \mathbb{S}(\boldsymbol{K}_i\boldsymbol{p})$, where $\boldsymbol{p} \in \texttt{cone}_\mu(\boldsymbol{p}^{\text{mm}})$. Following similar argument as in (A.32)–(A.33),

we obtain

$$\left|\hat{\boldsymbol{a}}_i^\top \operatorname{diag}(\boldsymbol{s}_i)\gamma_i - \hat{\boldsymbol{a}}_i^\top \boldsymbol{s}_i \boldsymbol{s}_i^\top \gamma_i - \sum_{t\in\mathcal{T}_i}(\hat{\boldsymbol{a}}_{i1} - \hat{\boldsymbol{a}}_{it})\boldsymbol{s}_{it}(\gamma_{i1} - \gamma_{it})\right| \le 2\Gamma A\left((1-\boldsymbol{s}_{i1})^2 + Q_i\right). \quad \text{(A.51)}$$

Note that $\hat{\boldsymbol{a}}_{i1} - \hat{\boldsymbol{a}}_{it} = \hat{\boldsymbol{q}}^\top(\boldsymbol{x}_{i1} - \boldsymbol{x}_{it})$ for all $\hat{\boldsymbol{q}} \in \mathbb{R}^{d\times d}$ with $\|\hat{\boldsymbol{q}}\| = \|\boldsymbol{p}^{\mathrm{mm}}\|$. Hence,

$$2A\sum_{t\in\mathcal{T}_i}\boldsymbol{s}_{it}(\gamma_{i1} - \gamma_{it}) \ge \sum_{t\in\mathcal{T}_i}\left(\hat{\boldsymbol{a}}_{i1} - \hat{\boldsymbol{a}}_{it}\right)\boldsymbol{s}_{it}\left(\gamma_{i1} - \gamma_{it}\right). \quad \text{(A.52)}$$

Next, we consider bounding (A.36) for samples from $\mathcal{I}$ and $\bar{\mathcal{I}}$, separably.

Case A: $i \in \mathcal{I}$ (Samples with support tokens). From (A.22b), we get

$$2A\bar{\gamma}_i^{\mathrm{gap}}\sum_{t\in\mathcal{T}_i}\boldsymbol{s}_{it} \ge \sum_{t\in\mathcal{T}_i}(\hat{\boldsymbol{a}}_{i1} - \hat{\boldsymbol{a}}_{it})\boldsymbol{s}_{it}(\gamma_{i1} - \gamma_{it}). \quad \text{(A.53)}$$

Using (A.51) and (A.53), we get

$$2A\bar{\gamma}_i^{\mathrm{gap}}S_i + 2A\Gamma\left((1-\boldsymbol{s}_{i1})^2 + Q_i\right) \ge \hat{\boldsymbol{a}}_i^\top \operatorname{diag}(\boldsymbol{s}_i)\gamma_i - \hat{\boldsymbol{a}}_i^\top \boldsymbol{s}_i\boldsymbol{s}_i^\top\gamma_i, \quad \text{for all } \ i\in\mathcal{I}. \quad \text{(A.54)}$$

Since $\boldsymbol{s}_i = \mathbb{S}(\boldsymbol{K}_i\boldsymbol{p})$ and $\boldsymbol{p} \in \mathtt{cone}_\mu(\boldsymbol{p}^{\mathrm{mm}})$, from Lemma A.30 and (A.38), $R \ge R_1$ gives $Q_i \le S_i$ for all $i \in \mathcal{I}$. Consequently, $(1-\boldsymbol{s}_{i1})^2 = (Q_i + S_i)^2 \le 4S_i^2 \le 4S_i$. This, together with (A.22b), implies that

$$(2A\bar{\gamma}_{\max}^{\mathrm{gap}} + 10A\Gamma)\,S_i \ge \hat{\boldsymbol{a}}_i^\top \operatorname{diag}(\boldsymbol{s}_i)\gamma_i - \hat{\boldsymbol{a}}_i^\top \boldsymbol{s}_i\boldsymbol{s}_i^\top\gamma_i, \quad \text{for all } \ i\in\mathcal{I}, \quad \text{(A.55)}$$

Recall $-\ell'_{\max} > 0$ be the max values negative loss derivative admits over the ball $[-A, A]$. It follows from (A.55) that

$$(2A\bar{\gamma}_{\max}^{\mathrm{gap}} + 10A\Gamma)\cdot(-\ell'_{\max})\cdot\frac{1}{n}\sum_{i\in\mathcal{I}}S_i \ge -\frac{1}{n}\sum_{i\in\mathcal{I}}\ell'_i\cdot\hat{\boldsymbol{a}}_i^\top\mathbb{S}'(\tilde{\boldsymbol{a}}_i)\gamma_i. \quad \text{(A.56a)}$$

Case B: $i \in \bar{\mathcal{I}}$ (Samples without support tokens). Since $i \in \bar{\mathcal{I}}$, we have $S_i = 0$. Hence, from (A.51), we obtain

$$2A\Gamma Q_i \ge \hat{\boldsymbol{a}}_i^\top \operatorname{diag}(\boldsymbol{s}_i)\gamma_i - \hat{\boldsymbol{a}}_i^\top \boldsymbol{s}_i\boldsymbol{s}_i^\top\gamma_i, \quad \text{for all } \ i\in\bar{\mathcal{I}},$$

which implies that

$$2A\Gamma \cdot (-\ell'_{\max}) \cdot \frac{1}{n} \sum_{i \in \bar{\mathcal{I}}} Q_i \geq -\frac{1}{n} \sum_{i \in \bar{\mathcal{I}}} \ell'_i \cdot \hat{\boldsymbol{a}}_i^\top \mathbb{S}'(\tilde{\boldsymbol{a}}_i) \gamma_i. \tag{A.56b}$$

In the following, we show that for a sufficiently large choice of R, we have

$$\sum_{i \in \bar{\mathcal{I}}} Q_i \leq \sum_{i \in \mathcal{I}} S_i. \tag{A.57}$$

To do so, using Lemma A.30, the above inequality is guaranteed if, for δ defined in (A.21b), we have

$$|\mathcal{I}| \cdot \frac{1}{T} e^{-R\Theta(1+\frac{1}{2}\delta)} \geq |\bar{\mathcal{I}}| \cdot T e^{-R\Theta(1+\frac{3}{2}\delta)}.$$

This gives the following choice of R

$$R \geq R_4 := \frac{1}{\delta\Theta} \log\left(\frac{|\bar{\mathcal{I}}|T^2}{|\mathcal{I}|}\right). \tag{A.58}$$

Taking both *Case A* and *Case B*, and combining (A.41) and (A.47), by choosing

$$R \geq \max\left(R_1, R_4\right), \tag{A.59}$$

we obtain

$$\begin{aligned}(2A\bar{\gamma}^{\text{gap}}_{\max} + 10A\Gamma + 2A\Gamma) \cdot (-\ell'_{\max}) \cdot \frac{1}{n} \sum_{i \in \mathcal{I}} S_i &\geq -\frac{1}{n} \sum_{i=1}^{n} \ell'_i \cdot \hat{\boldsymbol{a}}_i^\top \mathbb{S}'(\tilde{\boldsymbol{a}}_i) \gamma_i \\ &= -\left\langle \nabla \mathcal{L}(\boldsymbol{p}), \boldsymbol{q} \right\rangle. \end{aligned} \tag{A.60}$$

Recall $\Theta = 1/\|\boldsymbol{p}^{\text{mm}}\|$. We declare the constant

$$\hat{C} := 2\Theta\left(\bar{\gamma}^{\text{gap}}_{\max} + 5\Gamma + \Gamma \cdot (-\ell'_{\max})\right). \tag{A.61}$$

Thus,

$$\frac{A\hat{C}}{n\Theta} \sum_{i \in \mathcal{I}} (1 - \boldsymbol{s}_{i\alpha_i}) \geq -\left\langle \nabla \mathcal{L}(\boldsymbol{p}), \hat{\boldsymbol{q}} \right\rangle. \tag{A.62}$$

To proceed, since (A.62) holds for any $\hat{\boldsymbol{q}} \in \mathbb{R}^{d \times d}$, we replace it with $-\frac{\|\boldsymbol{p}^{\text{mm}}\|}{\|\nabla \mathcal{L}(\boldsymbol{p})\|} \cdot \nabla \mathcal{L}(\boldsymbol{p})$, to

obtain

$$\left\langle \nabla \mathcal{L}(\boldsymbol{p}), \frac{\|\boldsymbol{p}^{\mathrm{mm}}\|}{\|\nabla \mathcal{L}(\boldsymbol{p})\|} \cdot \nabla \mathcal{L}(\boldsymbol{p}) \right\rangle = \|\nabla \mathcal{L}(\boldsymbol{p})\| \, \|\boldsymbol{p}^{\mathrm{mm}}\| \leq \frac{A\hat{C}}{\Theta n} \sum_{i \in \mathcal{I}} (1 - \boldsymbol{s}_{i\alpha_i}). \tag{A.63a}$$

Simplifying $\Theta = 1/\|\boldsymbol{p}^{\mathrm{mm}}\|$ on both sides gives (A.27b).

On the other hand, it follows from (A.27a) for all $\boldsymbol{q} \in \mathtt{cone}_{\mu}(\boldsymbol{p}^{\mathrm{mm}})$ with $\|\boldsymbol{q}\| = \|\boldsymbol{p}^{\mathrm{mm}}\|$ and $\boldsymbol{p} \in \mathcal{C}_{\mu,\bar{R}_\mu}(\boldsymbol{p}^{\mathrm{mm}})$ that

$$-\left\langle \nabla \mathcal{L}(\boldsymbol{p}), \boldsymbol{q} \right\rangle \geq \frac{c}{n} \sum_{i \in \mathcal{I}} (1 - \boldsymbol{s}_{i\alpha_i}) > 0. \tag{A.63b}$$

Combining (A.63a) with (A.63b), we obtain (A.27c) for all $\boldsymbol{q}, \boldsymbol{p} \in \mathtt{cone}_{\mu}(\boldsymbol{p}^{\mathrm{mm}})$.

Finally, using (A.22) and (A.21), we have $A \geq 1$ and $\Gamma/\gamma_{\min}^{\mathrm{gap}} \geq 1$. Using (A.48), (A.47), and (A.59), $\bar{R}_\mu$ in the statement of the lemma is defined as

$$\bar{R}_\mu := \frac{1}{\delta\Theta} \log \left(32AT^2 \cdot \frac{\Gamma}{\gamma_{\min}^{\mathrm{gap}}} \cdot \frac{\ell'_{\max}}{\ell'_{\min}} \cdot \frac{|\bar{\mathcal{I}}|}{|\mathcal{I}|} \right). \tag{A.64}$$

Here, we assume that $|\bar{\mathcal{I}}|/|\mathcal{I}|$ is greater than or equal to 1 to simplify the expression. Otherwise, one can replace it with 1. ∎

Lemma A.7 (Local Gradient Condition) *Suppose Assumption 2.1 on the loss function ℓ holds, and let $\boldsymbol{\alpha} = (\alpha_i)_{i=1}^n$ be locally optimal tokens according to Definition 2.2. Let $\boldsymbol{p}^{mm}$ denote the SVM solution obtained via ($\boldsymbol{p}$-SVM). Let $\mu > 0$ and $\bar{R}_\mu$ be defined as in Lemma A.6. For any $\pi > 0$, there exists R_π such that $R_\pi \geq \bar{R}_\mu$ and all $\boldsymbol{p} \in \mathcal{C}_{\mu,R_\pi}(\boldsymbol{p}^{mm})$ obeys*

$$\left\langle \nabla \mathcal{L}(\boldsymbol{p}), \frac{\boldsymbol{p}}{\|\boldsymbol{p}\|} \right\rangle \geq (1 + \pi) \left\langle \nabla \mathcal{L}(\boldsymbol{p}), \frac{\boldsymbol{p}^{mm}}{\|\boldsymbol{p}^{mm}\|} \right\rangle.$$

Proof. Let $R := \|\boldsymbol{p}\|$,

$$\begin{gathered} \bar{\boldsymbol{p}} = \|\boldsymbol{p}^{\mathrm{mm}}\| \boldsymbol{p}/\|\boldsymbol{p}\|, \quad \tilde{\boldsymbol{a}}_i = \boldsymbol{K}_i \boldsymbol{p}, \quad \bar{\boldsymbol{a}}_i = \boldsymbol{K}_i \bar{\boldsymbol{p}}, \quad \boldsymbol{a}_i = \boldsymbol{K}_i \boldsymbol{p}^{\mathrm{mm}}, \\ \gamma_i = y_i \cdot \boldsymbol{X}_i \boldsymbol{v}, \quad \ell'_i = \ell'(\gamma_i^\top \mathbb{S}(\tilde{\boldsymbol{a}}_i)), \quad \text{and} \quad \boldsymbol{s}_i = \mathbb{S}(\tilde{\boldsymbol{a}}). \end{gathered} \tag{A.65}$$

To establish the result, using (A.31), we will prove that, for any $\pi > 0$, there is sufficiently

large R_π such that for any $\boldsymbol{p} \in \mathcal{C}_{\mu,R_\pi}(\boldsymbol{p}^{\mathrm{mm}})$:

$$\begin{aligned}\langle -\nabla\mathcal{L}(\boldsymbol{p}), \bar{\boldsymbol{p}}\rangle &= -\frac{1}{n}\sum_{i=1}^{n} \ell_i' \cdot \langle \bar{\boldsymbol{a}}_i, \mathbb{S}'(\boldsymbol{K}_i\boldsymbol{p})\gamma_i\rangle \\ &\le -\frac{1+\pi}{n}\sum_{i=1}^{n} \ell_i' \cdot \langle \boldsymbol{a}_i, \mathbb{S}'(\boldsymbol{K}_i\boldsymbol{p})\gamma_i\rangle = (1+\pi)\left\langle -\nabla\mathcal{L}(\boldsymbol{p}), \boldsymbol{p}^{\mathrm{mm}}\right\rangle. \end{aligned} \tag{A.66}$$

or equivalently,

$$\sum_{i=1}^{n} -\ell_i' \cdot \left((1+\pi)\left(\boldsymbol{a}_i^\top \mathrm{diag}(\boldsymbol{s}_i)\gamma_i - \boldsymbol{a}_i^\top \boldsymbol{s}_i\boldsymbol{s}_i^\top\gamma_i\right) - \left(\bar{\boldsymbol{a}}_i^\top \mathrm{diag}(\boldsymbol{s}_i)\gamma_i - \bar{\boldsymbol{a}}_i\boldsymbol{s}_i\boldsymbol{s}_i^\top\gamma_i\right)\right) \ge 0. \tag{A.67}$$

To proceed, recall $\bar{\mathcal{T}}_i$ is the set of all non-support indices per Definition 2.2, and $\mathcal{T}_i$ is the set of support indices. It follows from (A.33) that

$$\left|\boldsymbol{a}_i^\top \mathrm{diag}(\boldsymbol{s}_i)\gamma_i - \boldsymbol{a}_i^\top \boldsymbol{s}_i\boldsymbol{s}_i^\top\gamma_i - \sum_{t\in\mathcal{T}_i}(\boldsymbol{a}_{i1}-\boldsymbol{a}_{it})\boldsymbol{s}_{it}(\gamma_{i1}-\gamma_{it})\right| \le 2\Gamma A\left((1-\boldsymbol{s}_{i1})^2 + Q_i\right). \tag{A.68}$$

Note that $\boldsymbol{p}/\|\boldsymbol{p}\| = \boldsymbol{p}/\|\boldsymbol{p}\|$. Hence, $\boldsymbol{p} \in \mathtt{cone}_\mu(\boldsymbol{p}^{\mathrm{mm}})$ and $\|\boldsymbol{p}\| = \|\boldsymbol{p}^{\mathrm{mm}}\|$ which together with (A.33) implies that

$$\left|\bar{\boldsymbol{a}}_i^\top \mathrm{diag}(\boldsymbol{s}_i)\gamma_i - \bar{\boldsymbol{a}}_i^\top \boldsymbol{s}_i\boldsymbol{s}_i^\top\gamma_i - \sum_{t\in\mathcal{T}_i}(\bar{\boldsymbol{a}}_{i1}-\bar{\boldsymbol{a}}_{it})\boldsymbol{s}_{it}(\gamma_{i1}-\gamma_{it})\right| \le 2\Gamma A\left((1-\boldsymbol{s}_{i1})^2 + Q_i\right). \tag{A.69}$$

Let us assume (without losing generality) $\alpha_i = 1$ for all $i \in [n]$. Assuming $0 < \pi < 1$ (w.l.o.g.) and using (A.68)–(A.69), (A.67) is implied by the following stronger inequality

$$\sum_{i=1}^{n} -\ell_i' \cdot \left((1+\pi)\sum_{t\in\mathcal{T}_i}(\boldsymbol{a}_{i1}-\boldsymbol{a}_{it})\boldsymbol{s}_{it}(\gamma_{i1}-\gamma_{it}) - \sum_{t\in\bar{\mathcal{T}}_i}(\bar{\boldsymbol{a}}_{i1}-\bar{\boldsymbol{a}}_{it})\boldsymbol{s}_{it}(\gamma_{i1}-\gamma_{it}) - 6\Gamma A\left((1-\boldsymbol{s}_{i1})^2+Q_i\right)\right) \ge 0.$$

Note that since $\boldsymbol{a}_{i1} - \boldsymbol{a}_{it} = 1$, using the above inequality, (A.67) is implied by the following inequality

$$\sum_{i=1}^{n}(-\ell_i') \cdot \left((1+\pi)\sum_{t\in\mathcal{T}_i}\boldsymbol{s}_{it}(\gamma_{i1}-\gamma_{it}) - \sum_{t\in\bar{\mathcal{T}}_i}(\bar{\boldsymbol{a}}_{i1}-\bar{\boldsymbol{a}}_{it})\boldsymbol{s}_{it}(\gamma_{i1}-\gamma_{it}) - 6\Gamma A\left((1-\boldsymbol{s}_{i1})^2+Q_i\right)\right) \ge 0. \tag{A.70}$$

Recall from Lemma A.6 where we define $\mathcal{I} \subseteq [n]$ be the index set that contains samples with support non-(locally) optimal tokens per Definition 2.2 and let $\bar{\mathcal{I}} = [n] - \mathcal{I}$. For $i \in \mathcal{I}$,

$\mathcal{T}_i = \emptyset$. Therefore, (A.70) implies

$$\sum_{i\in\mathcal{I}}(-\ell_i')\cdot\left((1+\pi)\sum_{t\in\mathcal{T}_i}\boldsymbol{s}_{it}(\gamma_{i1}-\gamma_{it})-\sum_{t\in\mathcal{T}_i}(\bar{\boldsymbol{a}}_{i1}-\bar{\boldsymbol{a}}_{it})\boldsymbol{s}_{it}(\gamma_{i1}-\gamma_{it})\right)-\sum_{i=1}^{n}(-\ell_i')\cdot 6\Gamma A\left((1-\boldsymbol{s}_{i1})^2+Q_i\right)\geq 0.$$

In the following, we claim that

$$\begin{aligned}0.5\pi\sum_{i\in\mathcal{I}}\sum_{t\in\mathcal{T}_i}(-\ell_i')\cdot\boldsymbol{s}_{it}(\gamma_{i1}-\gamma_{it}) &\geq \sum_{i=1}^{n}(-\ell_i')\cdot 6\Gamma A((1-\boldsymbol{s}_{i1})^2+Q_i)\\ &=\sum_{i\in\mathcal{I}}(-\ell_i')\cdot 6\Gamma A((1-\boldsymbol{s}_{i1})^2+Q_i)+\sum_{i\in\bar{\mathcal{I}}}(-\ell_i')\cdot 6\Gamma A(Q_i^2+Q_i).\end{aligned}$$

We will prove the claim by showing

$$0.25\pi\sum_{i\in\mathcal{I}}\sum_{t\in\mathcal{T}_i}(-\ell_i')\cdot\boldsymbol{s}_{it}(\gamma_{i1}-\gamma_{it})\geq\sum_{i\in\mathcal{I}}(-\ell_i')\cdot 6\Gamma A((1-\boldsymbol{s}_{i1})^2+Q_i),\qquad\text{and}\qquad\text{(A.71a)}$$

$$0.25\pi\sum_{i\in\mathcal{I}}\sum_{t\in\mathcal{T}_i}(-\ell_i')\cdot\boldsymbol{s}_{it}(\gamma_{i1}-\gamma_{it})\geq\sum_{i\in\bar{\mathcal{I}}}(-\ell_i')\cdot 6\Gamma A\left(Q_i^2+Q_i\right).\qquad\text{(A.71b)}$$

Proof of (A.71a) directly follows the earlier argument, namely, following (A.37), (A.40), and (A.41) which leads to the choice

$$R\geq\max(R_1,\hat{R}_2),\quad\text{where}\quad\hat{R}_2:=\frac{2}{3\delta\Theta}\log\left(\frac{48\,(2-\delta)\,T\Gamma A}{\pi\gamma_{\min}^{\text{gap}}}\right),\qquad\text{(A.72a)}$$

where R_1 is defined in (A.38).

Additionally, (A.71b) follows (A.37), (A.44), (A.46), and (A.47), which leads to the choice

$$R\geq\hat{R}_3:=\frac{1}{\delta\Theta}\log\left(\frac{96|\bar{\mathcal{I}}|T^2\Gamma A\ell'_{\max}}{(2-\delta)|\mathcal{I}|\pi\gamma_{\min}^{\text{gap}}\ell'_{\min}}\right).\qquad\text{(A.72b)}$$

To conclude with the result, what remains is proving the comparison

$$\sum_{i\in\mathcal{I}}-\ell_i'\cdot\left(\sum_{t\in\mathcal{T}_i}(1+0.5\pi)\boldsymbol{s}_{it}(\gamma_{i1}-\gamma_{it})-(\bar{\boldsymbol{a}}_{i1}-\bar{\boldsymbol{a}}_{it})\boldsymbol{s}_{it}(\gamma_{i1}-\gamma_{it})\right)\geq 0.\qquad\text{(A.73)}$$

To proceed, we split the problem into two scenarios.

Scenario 1: $\|\bar{\boldsymbol{p}}-\boldsymbol{p}^{\text{mm}}\|\leq\epsilon=\frac{\pi}{4A\Theta}$ for some $\epsilon>0$. In this scenario, for the first token (as an optimal token) and any $t\in\mathcal{T}_i$ and $i\in[n]$, we have

$$|\bar{\boldsymbol{a}}_{it}-\boldsymbol{a}_{it}|=|\boldsymbol{k}_{it}^{\top}(\bar{\boldsymbol{p}}-\boldsymbol{p}^{\text{mm}})|\leq A\Theta\epsilon=\frac{\pi}{4}.$$

Note that $\boldsymbol{a}_{i1} - \boldsymbol{a}_{it} = 1$ where $\boldsymbol{a}_i = \boldsymbol{K}_i \boldsymbol{p}^{\mathrm{mm}}$. Consequently, we obtain

$$\begin{aligned}\bar{\boldsymbol{a}}_{i1} - \bar{\boldsymbol{a}}_{it} &= \boldsymbol{a}_{i1} - \boldsymbol{a}_{it} \\ &\quad + \bar{\boldsymbol{a}}_{i1} - \boldsymbol{a}_{i1} - \bar{\boldsymbol{a}}_{it} + \boldsymbol{a}_{it} \\ &\leq \boldsymbol{a}_{i1} - \boldsymbol{a}_{it} + 2A\Theta\epsilon = 1 + 0.5\pi.\end{aligned}$$

Similarly, $\bar{\boldsymbol{a}}_{i1} - \bar{\boldsymbol{a}}_{it} \geq 1 - 0.5\pi \geq 0.5$. Since all terms $\bar{\boldsymbol{a}}_{i1} - \bar{\boldsymbol{a}}_{it}, \boldsymbol{s}_{it}, \gamma_{i1} - \gamma_{it}$ in (A.73) are nonnegative and $(\bar{\boldsymbol{a}}_{i1} - \bar{\boldsymbol{a}}_{it})\boldsymbol{s}_{it}(\gamma_{i1} - \gamma_{it}) \leq (1 + 0.5\pi)\boldsymbol{s}_{it}(\gamma_{i1} - \gamma_{it})$, above implies the desired result in (A.73).

Scenario 2: $\|\bar{\boldsymbol{p}} - \boldsymbol{p}^{\mathrm{mm}}\| \geq \epsilon = \frac{\pi}{4A\Theta}$. Since $\boldsymbol{p}$ is not (locally) max-margin, in this scenario, for some $i \in [n]$, $\nu = \nu(\epsilon) > 0$, and $\tau \in \mathcal{T}_i$, we have that

$$\bar{\boldsymbol{a}}_{i1} - \bar{\boldsymbol{a}}_{i\tau} \leq 1 - 2\nu.$$

Here $\tau = \arg\max_{\tau \in \mathcal{T}_i} \boldsymbol{p}^\top \boldsymbol{k}_{i\tau}$ denotes the nearest point to $\bar{\boldsymbol{a}}_{i1}$ (along the $\boldsymbol{p}$ direction). Note that a non-neighbor $t \in \bar{\mathcal{T}}_i$ cannot be nearest because $\boldsymbol{p} \in \mathtt{cone}_\mu(\boldsymbol{p}^{\mathrm{mm}})$ and (A.23) holds. Recall that $\boldsymbol{s}_i = \mathbb{S}(\bar{R}\bar{\boldsymbol{a}}_i)$ where $\bar{R} = \|\boldsymbol{p}\|\Theta$. To proceed, let $\underline{\bar{\boldsymbol{a}}}_i := \min_{t \in \mathcal{T}_i} \bar{\boldsymbol{a}}_{i1} - \bar{\boldsymbol{a}}_{it}$,

$$\mathcal{I}_1 := \left\{ i \in \mathcal{I} : \underline{\bar{\boldsymbol{a}}}_i \leq 1 - 2\nu \right\}, \qquad \mathcal{I} - \mathcal{I}_1 := \left\{ i \in \mathcal{I} : 1 - 2\nu < \underline{\bar{\boldsymbol{a}}}_i \right\}.$$

For all $\mathcal{I} - \mathcal{I}_1$,

$$\begin{aligned}&\sum_{t \in \mathcal{T}_i} (\bar{\boldsymbol{a}}_{i1} - \bar{\boldsymbol{a}}_{it})\boldsymbol{s}_{it}(\gamma_{i1} - \gamma_{it}) - (1 + 0.5\pi) \sum_{t \in \mathcal{T}_i} \boldsymbol{s}_{it}(\gamma_{i1} - \gamma_{it}) \\ &\qquad \leq \sum_{t \in \mathcal{T}_i,\ \bar{\boldsymbol{a}}_{i1} - \bar{\boldsymbol{a}}_{it} \geq 1 + \frac{\pi}{2}} (\bar{\boldsymbol{a}}_{i1} - \bar{\boldsymbol{a}}_{it})\boldsymbol{s}_{it}(\gamma_{i1} - \gamma_{it}) \\ &\qquad \leq 2A\Gamma \sum_{t \in \mathcal{T}_i,\ \bar{\boldsymbol{a}}_{i1} - \bar{\boldsymbol{a}}_{it} \geq 1 + \frac{\pi}{2}} \boldsymbol{s}_{it} \\ &\qquad \leq 2A\Gamma T e^{-\bar{R}(1 + \frac{\pi}{2})}.\end{aligned} \tag{A.74}$$

Here, the first inequality follows since $\boldsymbol{s}_{it} > 0$ and $\gamma_{i1} - \gamma_{it}$ are non-negative; second inequality is due to the definition of global scalar $\Gamma = \sup_{i \in [n], t, \tau \in [T]} |\gamma_{it} - \gamma_{i\tau}|$, (A.21c), and since $\boldsymbol{p} \in \mathtt{cone}_\mu(\boldsymbol{p}^{\mathrm{mm}})$ with $\|\boldsymbol{p}\| = \|\boldsymbol{p}^{\mathrm{mm}}\|$ and $\max_{t \in [T]} |\bar{\boldsymbol{a}}_t| = \max_{t \in [T]} |(\boldsymbol{K}\boldsymbol{p})_t| \leq A$.

For all $i \in \mathcal{I}_1$, split the tokens into two groups: Let $\mathcal{N}_i$ be the group of tokens obeying $\bar{\boldsymbol{a}}_{i1} - \bar{\boldsymbol{a}}_{it} \leq 1 - \nu$ and $\mathcal{T}_i - \mathcal{N}_i$ be the rest of the neighbors. Observe that

$$\frac{\sum_{t \in \mathcal{T}_i - \mathcal{N}_i} \boldsymbol{s}_{it}}{\sum_{t \in \mathcal{T}_i} \boldsymbol{s}_{it}} \leq T \frac{e^{\nu \bar{R}}}{e^{2\nu \bar{R}}} = T e^{-\bar{R}\nu}.$$

Using $|\bar{\boldsymbol{a}}_{i1} - \bar{\boldsymbol{a}}_{it}| \leq 2A$ and $\gamma_{\min}^{\mathrm{gap}} = \min_{i\in[n]} \gamma_i^{\mathrm{gap}} = \min_{i\in[n]}(\gamma_{i1} - \max_{t\in\mathcal{T}_i} \gamma_{it})$, observe that

$$\sum_{t\in\mathcal{T}_i-\mathcal{N}_i} (\bar{\boldsymbol{a}}_{i1} - \bar{\boldsymbol{a}}_{it})\boldsymbol{s}_{it}(\gamma_{i1} - \gamma_{it}) \leq \frac{2\Gamma ATe^{-\bar{R}\nu}}{\gamma_{\min}^{\mathrm{gap}}} \sum_{t\in\mathcal{T}_i} \boldsymbol{s}_{it}(\gamma_{i1} - \gamma_{it}).$$

Thus,

$$\begin{aligned}
\sum_{t\in\mathcal{T}_i} (\bar{\boldsymbol{a}}_{i1} - \bar{\boldsymbol{a}}_{it})\boldsymbol{s}_{it}(\gamma_{i1} - \gamma_{it}) &= \sum_{t\in\mathcal{N}_i} (\bar{\boldsymbol{a}}_{i1} - \bar{\boldsymbol{a}}_{it})\boldsymbol{s}_{it}(\gamma_{i1} - \gamma_{it}) + \sum_{t\in\mathcal{T}_i-\mathcal{N}_i} (\bar{\boldsymbol{a}}_{i1} - \bar{\boldsymbol{a}}_{it})\boldsymbol{s}_{it}(\gamma_{i1} - \gamma_{it}) \\
&\leq \sum_{t\in\mathcal{N}_i} (1-\nu)\boldsymbol{s}_{it}(\gamma_{i1} - \gamma_{it}) + \frac{2\Gamma ATe^{-\bar{R}\nu}}{\gamma_{\min}^{\mathrm{gap}}} \sum_{t\in\mathcal{T}_i} \boldsymbol{s}_{it}(\gamma_{i1} - \gamma_{it}) \\
&\leq \left(1 - \nu + \frac{2\Gamma ATe^{-\bar{R}\nu}}{\gamma_{\min}^{\mathrm{gap}}}\right) \sum_{t\in\mathcal{T}_i} \boldsymbol{s}_{it}(\gamma_{i1} - \gamma_{it}) \\
&\leq \left(1 + \frac{2\Gamma ATe^{-\bar{R}\nu}}{\gamma_{\min}^{\mathrm{gap}}}\right) \sum_{t\in\mathcal{T}_i} \boldsymbol{s}_{it}(\gamma_{i1} - \gamma_{it}).
\end{aligned}$$

Hence, choosing

$$R \geq \hat{R}_4 := \frac{1}{\nu\Theta} \log\left(\frac{8\Gamma AT}{\gamma_{\min}^{\mathrm{gap}}\pi}\right) \tag{A.75}$$

results in that

$$\begin{aligned}
&\sum_{t\in\mathcal{T}_i} (\bar{\boldsymbol{a}}_{i1} - \bar{\boldsymbol{a}}_{it})\boldsymbol{s}_{it}(\gamma_{i1} - \gamma_{it}) - \left(1 + \frac{\pi}{2}\right) \sum_{t\in\mathcal{T}_i} \boldsymbol{s}_{it}(\gamma_{i1} - \gamma_{it}) \\
&\leq \left(\frac{2\Gamma ATe^{-\bar{R}\nu}}{\gamma_{\min}^{\mathrm{gap}}} - \frac{\pi}{2}\right) \sum_{t\in\mathcal{T}_i} \boldsymbol{s}_{it}(\gamma_{i1} - \gamma_{it}) \\
&\leq -\frac{\pi}{4} \sum_{t\in\mathcal{T}_i} \boldsymbol{s}_{it}(\gamma_{i1} - \gamma_{it}) \\
&\leq -\frac{\pi}{4T} \gamma_{\min}^{\mathrm{gap}} e^{-\bar{R}(1-2\nu)}.
\end{aligned} \tag{A.76}$$

Here, the last inequality follows from the fact that for all $i \in \mathcal{I}_1$,

$$\sum_{t\in\mathcal{T}_i} \boldsymbol{s}_{it} \geq \max_{t\in\mathcal{T}_i} s_{it} \geq \frac{e^{-\bar{R}(1-2\nu)}}{\sum_{t=1}^{T} e^{-\bar{R}(\bar{\boldsymbol{a}}_{i1} - \bar{\boldsymbol{a}}_{it})}} \geq \frac{e^{-\bar{R}(1-2\nu)}}{T},$$

where $\boldsymbol{s}_i = \mathbb{S}(\bar{R}\bar{\boldsymbol{a}}_i)$ and $\bar{R} = \|\boldsymbol{p}\|\Theta$.

From Assumption 2.1, let $-\ell'_{\min}$ and $-\ell'_{\max}$ be the min and max values negative loss

derivative admits over the ball $[-A, A]$, respectively. It follows from (A.74) and (A.76) that

$$\begin{aligned}
-\sum_{i\in\mathcal{I}} \ell_i' \cdot \left(\sum_{t\in\mathcal{T}_i} (\bar{\boldsymbol{a}}_{i1} - \bar{\boldsymbol{a}}_{it}) \boldsymbol{s}_{it} (\gamma_{i1} - \gamma_{it}) - \sum_{t\in\mathcal{T}_i} (1 + 0.5\pi) \boldsymbol{s}_{it} (\gamma_{i1} - \gamma_{it}) \right) \\
\leq (-\ell'_{\max}) |\mathcal{I}| 2A\Gamma T \Gamma e^{-\bar{R}(1+\frac{\pi}{2})} - \frac{(-\ell'_{\min})}{T} \cdot \frac{\pi \gamma_{\min}^{\text{gap}}}{4} e^{-\bar{R}(1-2\nu)} \\
\leq 0.
\end{aligned}$$

Here, the last inequality is obtained by using

$$R \geq \hat{R}_5 := \frac{1}{(2\nu + \pi/2)\Theta} \log \left(\frac{8|\mathcal{I}|\Gamma A T^2 \ell'_{\max}}{\ell'_{\min} \gamma_{\min}^{\text{gap}} \pi} \right), \tag{A.77}$$

Combing with (A.75), (A.73) is guaranteed by choosing

$$R \geq \max \left\{ \hat{R}_4, \hat{R}_5 \right\},$$

where $\nu = \nu(\frac{\pi}{4A\Theta})$ depends only on π and global problem variables.

Combining this with the prior R choices in (A.72), (A.75), and (A.77), i.e., $\max\{R_1, \hat{R}_2, \ldots, \hat{R}_5\}$, gives

$$R_\pi := \frac{1}{\min(\delta, \nu, \pi)\Theta} \log \left(\frac{C_0 A T^2}{\pi} \cdot \frac{\Gamma}{\gamma_{\min}^{\text{gap}}} \cdot \frac{\ell'_{\max}}{\ell'_{\min}} \cdot \frac{|\bar{\mathcal{I}}|}{|\mathcal{I}|} \right) \quad \text{with} \quad C_0 \geq 96. \tag{A.78}$$

Here, similar to (A.64), we assume that $|\bar{\mathcal{I}}|/|\mathcal{I}|$ is greater than or equal to 1 to simplify the expression. Otherwise, one can replace $|\bar{\mathcal{I}}|/|\mathcal{I}|$ with 1 in (A.78). ■

A.2.3 Proof of Theorem 2.1

Proof. This proof is a direct corollary of Lemma A.14 which itself is a special case of the nonlinear head Theorem 2.7. Let us verify that $f(\boldsymbol{X}) = \boldsymbol{v}^\top \boldsymbol{X}^\top \mathbb{S}(\boldsymbol{X}\boldsymbol{p})$ satisfies the assumptions of Lemma A.14 where we replace the nonlinear head with linear $\boldsymbol{v}$. To see this, set the optimal sets to be the singletons $\mathcal{R}_i = \{\text{opt}_i\}$. Given $(\boldsymbol{X}_i, y_i)$, let $\boldsymbol{s}_i = \mathbb{S}(\boldsymbol{K}_i \boldsymbol{p})$ and $q_i = q_i^{\boldsymbol{p}} = \sum_{t\neq \text{opt}_i} \boldsymbol{s}_{it}$. Recalling score definition $\gamma_i = y_i \cdot \boldsymbol{X}_i \boldsymbol{v}$ and setting $\nu_i := \gamma_{i\text{opt}_i}$ and $Z_i := \sum_{t\neq \text{opt}_i} \gamma_{it} \boldsymbol{s}_{it}$, a particular prediction can be written as

$$\begin{aligned}
y_i \cdot \boldsymbol{v}^\top \boldsymbol{X}_i^\top \mathbb{S}(\boldsymbol{X}_i \boldsymbol{p}) = \gamma_i^\top \boldsymbol{s}_i = \gamma_{i\text{opt}_i}(1 - q_i) + \sum_{t\neq \text{opt}_i} \gamma_{it} \boldsymbol{s}_{it} \\
= \nu_i (1 - q_i) + Z_i.
\end{aligned}$$

To proceed, we demonstrate the choices for $C, \epsilon > 0$. Let $C := -\min_{i\in[n],t\in[T]} \gamma_{it} \wedge 0$ and $q_{\max} = \max_{i\in[n]} q_i$. Note that $Z_i \geq \sum_{t\neq \mathrm{opt}_i} \gamma_{it} \boldsymbol{s}_{it} \geq q_i \gamma_{\min} \geq -C q_{\max}$. Now, using strict score optimality of opt_i's for all $i \in [n]$, we set

$$\epsilon := 1 - \sup_{i\in[n]} \frac{\sum_{t\neq \mathrm{opt}_i} \gamma_{it} \boldsymbol{s}_{it}}{\nu_i q_i} \geq 1 - \sup_{i\in[n]} \frac{\sup_{t\neq \mathrm{opt}_i} \gamma_{it}}{\gamma_{i\mathrm{opt}_i}} > 0.$$

We conclude by observing $Z_i \leq \nu_i q_i \frac{\sum_{t\neq \mathrm{opt}_i} \gamma_{it} \boldsymbol{s}_{it}}{\nu_i q_i} \leq \nu_i q_i \epsilon$ as desired. ∎

A.2.4 Proof of Theorem 2.2

Proof. We first show that $\lim_{t\to\infty} \|\boldsymbol{p}(t)\| = \infty$. From Lemma A.2, we have

$$\langle \nabla \mathcal{L}(\boldsymbol{p}), \boldsymbol{p}^{\mathrm{mm}\star} \rangle = \frac{1}{n} \sum_{i=1}^{n} \ell'(y_i \cdot \boldsymbol{v}^\top \boldsymbol{X}_i^\top \mathbb{S}(\boldsymbol{K}_i \boldsymbol{p})) \cdot \langle \boldsymbol{K}_i \boldsymbol{p}^{\mathrm{mm}\star}, \mathbb{S}'(\boldsymbol{a}_i) \gamma_i \rangle ,$$

where $\gamma_i = y_i \cdot \boldsymbol{X}_i \boldsymbol{v}$ and $\boldsymbol{a}_i = \boldsymbol{K}_i \boldsymbol{p}$.

It follows from Lemma A.2 that $\langle \nabla \mathcal{L}(\boldsymbol{p}), \boldsymbol{p}^{\mathrm{mm}\star} \rangle < 0$ for all $\boldsymbol{p} \in \mathbb{R}^d$. Hence, for any finite $\boldsymbol{p}$, $\langle \nabla \mathcal{L}(\boldsymbol{p}), \boldsymbol{p}^{\mathrm{mm}\star} \rangle$ cannot be equal to zero, as a sum of negative terms. Therefore, there are no finite critical points $\boldsymbol{p}$, for which $\nabla \mathcal{L}(\boldsymbol{p}) = 0$. On the other hand, Lemma A.4 states $\nabla \mathcal{L}(\boldsymbol{p}(t)) \to 0$ which implies that $\|\boldsymbol{p}(t)\| \to \infty$.

Next, we provide the directional convergence for the setting $n = 1$. Let us consider an arbitrary value of $\epsilon \in (0, 1)$ and set $\pi = \epsilon/(1-\epsilon)$. As $\lim_{t\to\infty} \|\boldsymbol{p}(t)\| = \infty$, we can select a specific t_ϵ such that for all $t \geq t_\epsilon$, it holds that $\|\boldsymbol{p}(t)\| \geq R_\epsilon \vee 1/2$ for any choice of R_ϵ. To proceed, we choose R_ϵ based on Lemma A.3 so that for any $t \geq t_\epsilon$, we have that

$$\left\langle -\nabla \mathcal{L}(\boldsymbol{p}(t)), \frac{\boldsymbol{p}^{\mathrm{mm}\star}}{\|\boldsymbol{p}^{\mathrm{mm}\star}\|} \right\rangle \geq (1-\epsilon) \left\langle -\nabla \mathcal{L}(\boldsymbol{p}(t)), \frac{\boldsymbol{p}(t)}{\|\boldsymbol{p}(t)\|} \right\rangle .$$

Multiplying both sides by the stepsize η and using the gradient descent update, we get

$$\begin{aligned}
&\left\langle \boldsymbol{p}(t+1)-\boldsymbol{p}(t), \frac{\boldsymbol{p}^{\mathrm{mm}\star}}{\|\boldsymbol{p}^{\mathrm{mm}\star}\|} \right\rangle \\
&\geq (1-\epsilon)\left\langle \boldsymbol{p}(t+1)-\boldsymbol{p}(t), \frac{\boldsymbol{p}(t)}{\|\boldsymbol{p}(t)\|} \right\rangle \\
&= \frac{(1-\epsilon)}{2\|\boldsymbol{p}(t)\|}\left(\|\boldsymbol{p}(t+1)\|^2-\|\boldsymbol{p}(t)\|^2-\|\boldsymbol{p}(t+1)-\boldsymbol{p}(t)\|^2\right) \\
&\geq (1-\epsilon)\left(\frac{1}{2\|\boldsymbol{p}(t)\|}\left(\|\boldsymbol{p}(t+1)\|^2-\|\boldsymbol{p}(t)\|^2\right)-\|\boldsymbol{p}(t+1)-\boldsymbol{p}(t)\|^2\right) \\
&\geq (1-\epsilon)\left(\|\boldsymbol{p}(t+1)\|-\|\boldsymbol{p}(t)\|-\|\boldsymbol{p}(t+1)-\boldsymbol{p}(t)\|^2\right) \\
&\geq (1-\epsilon)\Big(\|\boldsymbol{p}(t+1)\|-\|\boldsymbol{p}(t)\|-2\eta\left(\mathcal{L}(\boldsymbol{p}(t))-\mathcal{L}(\boldsymbol{p}(t+1))\right)\Big).
\end{aligned} \tag{A.79}$$

Here, the second inequality is obtained from $\|\boldsymbol{p}(t)\| \geq 1/2$; the third inequality follows since for any $a, b > 0$, we have $(a^2-b^2)/(2b)-(a-b) \geq 0$; and the last inequality uses Lemma A.4.

Summing the above inequality over $t \geq t_\epsilon$ gives

$$\left\langle \frac{\boldsymbol{p}(t)}{\|\boldsymbol{p}(t)\|}, \frac{\boldsymbol{p}^{\mathrm{mm}\star}}{\|\boldsymbol{p}^{\mathrm{mm}\star}\|} \right\rangle \geq 1-\epsilon+\frac{C(\epsilon,\eta)}{\|\boldsymbol{p}(t)\|},$$

for some finite constant $C(\epsilon, \eta)$ defined as

$$C(\epsilon,\eta) := \left\langle \boldsymbol{p}(t_\epsilon), \frac{\boldsymbol{p}^{\mathrm{mm}\star}}{\|\boldsymbol{p}^{\mathrm{mm}\star}\|} \right\rangle - (1-\epsilon)\|\boldsymbol{p}(t_\epsilon)\| - 2\eta(1-\epsilon)\left(\mathcal{L}(\boldsymbol{p}(t_\epsilon))-\mathcal{L}^*\right), \tag{A.80}$$

where $\mathcal{L}^*$ denotes the minimum objective.

Since $\|\boldsymbol{p}(t)\| \to \infty$, we get

$$\liminf_{t\to\infty} \left\langle \frac{\boldsymbol{p}(t)}{\|\boldsymbol{p}(t)\|}, \frac{\boldsymbol{p}^{\mathrm{mm}\star}}{\|\boldsymbol{p}^{\mathrm{mm}\star}\|} \right\rangle \geq 1-\epsilon.$$

Given that we can choose any value of $\epsilon \in (0,1)$, we have $\boldsymbol{p}(t)/\|\boldsymbol{p}(t)\| \to \boldsymbol{p}^{\mathrm{mm}\star}/\|\boldsymbol{p}^{\mathrm{mm}\star}\|$. ■

A.2.5 Proof of Theorem 2.3

Proof. Following the proof of Lemma A.6, let $(\mathcal{T}_i)_{i=1}^n$ denote the sets of SVM-neighbors as defined in Definition 2.2. We define $\bar{\mathcal{T}}_i = [T] - \mathcal{T}_i - \{\alpha_i\}$ as the tokens that are non-SVM neighbors. Additionally, let μ be defined as in (A.21). Let us denote the initialization lower bound as $R_\mu^0 := R$, where R is given in the Theorem 2.3's statement. Consider an arbitrary value of $\epsilon \in (0, \mu/2)$ and let $1/(1+\pi) = 1-\epsilon$. $\pi \leq$ We additionally denote $R_\epsilon \leftarrow R_\pi \vee 1/2$

where R_π was defined in (A.78). At initialization $\boldsymbol{p}(0)$, we set $\epsilon = \mu/2$ to obtain $R_\mu^0 = R_{\mu/2}$ and provide the proof in four steps:

Step 1: There are no stationary points within $\mathcal{C}_{\mu,R_\mu^0}(\boldsymbol{p}^{\mathrm{mm}})$. We begin by proving that there are no stationary points within $\mathcal{C}_{\mu,R_\mu^0}(\boldsymbol{p}^{\mathrm{mm}})$. Then, since $R_\mu^0 \geq \bar{R}_\mu$ per Lemma A.6, we can apply (A.27a) to find that: For all $\boldsymbol{q}, \boldsymbol{p} \in \mathtt{cone}_\mu(\boldsymbol{p}^{\mathrm{mm}})$ with $\boldsymbol{q} \neq 0$ and $\|\boldsymbol{p}\| \geq R_\mu^0$, we have that $-\boldsymbol{q}^\top \nabla \mathcal{L}(\boldsymbol{p})$ is strictly positive.

Step 2: It follows from Lemma A.7 that, for all $\epsilon \in (0, \mu/2)$, all $\boldsymbol{p} \in \mathcal{C}_{\mu,R_\epsilon}(\boldsymbol{p}^{\mathrm{mm}})$ satisfy

$$\left\langle -\nabla \mathcal{L}(\boldsymbol{p}), \frac{\boldsymbol{p}^{\mathrm{mm}}}{\|\boldsymbol{p}^{\mathrm{mm}}\|} \right\rangle \geq (1-\epsilon) \left\langle -\nabla \mathcal{L}(\boldsymbol{p}), \frac{\boldsymbol{p}}{\|\boldsymbol{p}\|} \right\rangle. \tag{A.81}$$

The argument above applies to a general $\epsilon \in (0, \mu/2)$. However, at initialization $\boldsymbol{p}(0)$, we set $\epsilon = \mu/2$ to obtain our earlier R_μ^0 choice. To proceed, for any $\epsilon \in (0, \mu/2)$, we will show that after gradient descent enters the conic set $\mathcal{C}_{\mu,R_\epsilon}(\boldsymbol{p}^{\mathrm{mm}})$ for the first time, it will never leave the set. Let t_ϵ be the first time gradient descent enters $\mathcal{C}_{\mu,R_\epsilon}(\boldsymbol{p}^{\mathrm{mm}})$. In **Step 4**, we will prove that such t_ϵ is guaranteed to exist. Additionally, for $\epsilon \leftarrow \mu/2$, note that $t_\epsilon = 0$ i.e. the point of initialization.

Step 3: Updates remain inside the cone $\mathcal{C}_{\mu,R_\epsilon}(\boldsymbol{p}^{\mathrm{mm}})$. By leveraging the results from **Step 1** and **Step 2**, we demonstrate that the gradient iterates, with an appropriate constant step size, starting from $\boldsymbol{p}(t_\epsilon) \in \mathcal{C}_{\mu,R_\epsilon}(\boldsymbol{p}^{\mathrm{mm}})$, remain within this cone.

We proceed by induction. Suppose that the claim holds up to iteration $t \geq t_\epsilon$. This implies that $\boldsymbol{p}(t) \in \mathcal{C}_{\mu,R_\epsilon}(\boldsymbol{p}^{\mathrm{mm}})$. Hence, recalling cone definition, for $\mu > 0$ and R_ϵ, we have $\left\langle \frac{\boldsymbol{p}(t)}{\|\boldsymbol{p}(t)\|}, \frac{\boldsymbol{p}^{\mathrm{mm}}}{\|\boldsymbol{p}^{\mathrm{mm}}\|} \right\rangle \geq 1-\mu$ and $\|\boldsymbol{p}(t)\| \geq R_\epsilon$. Let

$$\rho(t) := -\frac{1}{1-\epsilon} \left\langle \nabla \mathcal{L}(\boldsymbol{p}(t)), \frac{\boldsymbol{p}^{\mathrm{mm}}}{\|\boldsymbol{p}^{\mathrm{mm}}\|} \right\rangle.$$

Note that $\rho(t) > 0$ due to **Step 1**. This together with the gradient descent update rule gives

$$\begin{aligned} \left\langle \frac{\boldsymbol{p}(t+1)}{\|\boldsymbol{p}(t)\|}, \frac{\boldsymbol{p}^{\mathrm{mm}}}{\|\boldsymbol{p}^{\mathrm{mm}}\|} \right\rangle &= \left\langle \frac{\boldsymbol{p}(t)}{\|\boldsymbol{p}(t)\|} - \frac{\eta}{\|\boldsymbol{p}(t)\|} \nabla \mathcal{L}(\boldsymbol{p}(t)), \frac{\boldsymbol{p}^{\mathrm{mm}}}{\|\boldsymbol{p}^{\mathrm{mm}}\|} \right\rangle \\ &\geq 1 - \mu - \frac{\eta}{\|\boldsymbol{p}(t)\|} \left\langle \nabla \mathcal{L}(\boldsymbol{p}(t)), \frac{\boldsymbol{p}^{\mathrm{mm}}}{\|\boldsymbol{p}^{\mathrm{mm}}\|} \right\rangle \\ &= 1 - \mu + \frac{\eta \rho(t)(1-\epsilon)}{\|\boldsymbol{p}(t)\|}. \end{aligned} \tag{A.82a}$$

Note that from Lemma A.6, we have $\langle \nabla \mathcal{L}(\boldsymbol{p}(t)), \boldsymbol{p}(t) \rangle < 0$ which implies that $\|\boldsymbol{p}(t+1)\| \geq \|\boldsymbol{p}(t)\|$. This together with R_ϵ definition and $\|\boldsymbol{p}(t)\| \geq 1/2$ implies that

$$\begin{aligned}
\|\boldsymbol{p}(t+1)\| &\leq \frac{1}{2\|\boldsymbol{p}(t)\|} \left(\|\boldsymbol{p}(t+1)\|^2 + \|\boldsymbol{p}(t)\|^2 \right) \\
&= \frac{1}{2\|\boldsymbol{p}(t)\|} \left(2\|\boldsymbol{p}(t)\|^2 - 2\eta \langle \nabla \mathcal{L}(\boldsymbol{p}(t)), \boldsymbol{p}(t) \rangle + \eta^2 \|\nabla \mathcal{L}(\boldsymbol{p}(t))\|^2 \right) \\
&\leq \|\boldsymbol{p}(t)\| - \frac{\eta}{\|\boldsymbol{p}(t)\|} \langle \nabla \mathcal{L}(\boldsymbol{p}(t)), \boldsymbol{p}(t) \rangle + \eta^2 \|\nabla \mathcal{L}(\boldsymbol{p}(t))\|^2.
\end{aligned} \tag{A.82b}$$

Hence, using (A.81)

$$\begin{aligned}
\frac{\|\boldsymbol{p}(t+1)\|}{\|\boldsymbol{p}(t)\|} &\leq 1 - \frac{\eta}{\|\boldsymbol{p}(t)\|} \left\langle \nabla \mathcal{L}(\boldsymbol{p}(t)), \frac{\boldsymbol{p}(t)}{\|\boldsymbol{p}(t)\|} \right\rangle + \eta^2 \frac{\|\nabla \mathcal{L}(\boldsymbol{p}(t))\|^2}{\|\boldsymbol{p}(t)\|} \\
&\leq 1 - \frac{\eta}{(1-\epsilon)\|\boldsymbol{p}(t)\|} \left\langle \nabla \mathcal{L}(\boldsymbol{p}(t)), \frac{\boldsymbol{p}^{\mathrm{mm}}}{\|\boldsymbol{p}^{\mathrm{mm}}\|} \right\rangle + \eta^2 \frac{\|\nabla \mathcal{L}(\boldsymbol{p}(t))\|^2}{\|\boldsymbol{p}(t)\|} \\
&= 1 + \frac{\eta \rho(t)}{\|\boldsymbol{p}(t)\|} + \frac{\eta^2 \|\nabla \mathcal{L}(\boldsymbol{p}(t))\|^2}{\|\boldsymbol{p}(t)\|} =: C_1(\rho(t), \eta).
\end{aligned} \tag{A.82c}$$

Here, the second inequality follows from (A.81).

Now, it follows from (A.82a) and (A.82c) that

$$\begin{aligned}
&\left\langle \frac{\boldsymbol{p}(t+1)}{\|\boldsymbol{p}(t+1)\|}, \frac{\boldsymbol{p}^{\mathrm{mm}}}{\|\boldsymbol{p}^{\mathrm{mm}}\|} \right\rangle \\
&\geq \frac{1}{C_1(\rho(t), \eta)} \left(1 - \mu + \frac{\eta \rho(t)(1-\epsilon)}{\|\boldsymbol{p}(t)\|} \right) \\
&= 1 - \mu + \frac{1}{C_1(\rho(t), \eta)} \left((1-\mu)(1 - C_1(\rho(t), \eta)) + \frac{\eta \rho(t)(1-\epsilon)}{\|\boldsymbol{p}(t)\|} \right) \\
&= 1 - \mu + \frac{\eta}{C_1(\rho(t), \eta)} \left((\mu - 1)(\frac{\rho(t)}{\|\boldsymbol{p}(t)\|} + \frac{\eta \|\nabla \mathcal{L}(\boldsymbol{p}(t))\|^2}{\|\boldsymbol{p}(t)\|}) + \frac{\rho(t)(1-\epsilon)}{\|\boldsymbol{p}(t)\|} \right) \\
&= 1 - \mu + \frac{\eta}{C_1(\rho(t), \eta)} \left(\frac{\rho(t)(\mu - \epsilon)}{\|\boldsymbol{p}(t)\|} - \eta(1-\mu) \frac{\|\nabla \mathcal{L}(\boldsymbol{p}(t))\|^2}{\|\boldsymbol{p}(t)\|} \right) \\
&\geq 1 - \mu,
\end{aligned} \tag{A.83}$$

where the last inequality uses our choice of stepsize $\eta \leq 1/L_p$ in Theorem 2.3's statement. Specifically, we need η to be small to ensure the last inequality. We will guarantee this by choosing a proper R_ϵ in Lemma A.7. Specifically, Lemma A.7 leaves the choice of C_0 in R_ϵ lower bound of (A.78) open (it can always be chosen larger). Here, by choosing C_0 sufficiently large, we ensure $\eta \leq 1/L_p$ works well.

To proceed, we have that

$$\begin{aligned}\frac{(\mu-\epsilon)}{1-\mu}\frac{\rho(t)}{\|\nabla\mathcal{L}(\boldsymbol{p}(t))\|^2} &\geq \frac{\mu-\epsilon}{1-\mu}\frac{1}{1-\epsilon}\frac{c}{\hat{C}}\frac{\Theta}{A}\frac{1}{A\hat{C}T}e^{R_\mu^0\Theta/2} \\ &\geq \frac{\mu}{2(1-\mu)(1-\frac{\mu}{2})}\frac{c}{\hat{C}}\frac{\Theta}{A}\frac{1}{ACT}e^{R_\mu^0\Theta/2} \geq \eta.\end{aligned} \tag{A.84}$$

Here, the second inequality uses our choice of $\epsilon \in (0, \mu/2)$ (see **Step 2**), and the first inequality is obtained from Lemma A.6 since

$$\begin{aligned}&\frac{\rho(t)}{\|\nabla\mathcal{L}(\boldsymbol{p}(t))\|} = -\frac{1}{1-\epsilon}\left\langle\frac{\nabla\mathcal{L}(\boldsymbol{p}(t))}{\|\nabla\mathcal{L}(\boldsymbol{p}(t))\|}, \frac{\boldsymbol{p}^{\mathrm{mm}}}{\|\boldsymbol{p}^{\mathrm{mm}}\|}\right\rangle \geq \frac{1}{1-\epsilon}\frac{c}{\hat{C}}\frac{\Theta}{A}, \\ &\frac{1}{\|\nabla\mathcal{L}(\boldsymbol{p}(t))\|} \geq \frac{1}{A\hat{C}\frac{1}{n}\sum_{i\in\mathcal{I}}(1-\boldsymbol{s}_{i\alpha_i})} \geq \frac{1}{A\hat{C}Te^{-R_\mu^0\Theta/2}}\end{aligned}$$

for some dataset-dependent positive constants C, $\hat{C}$, and c in (A.27); and A, Θ defined in (A.21).

Next, we will demonstrate that the choice of η in (A.84) does indeed meet our step size condition as stated in the theorem, i.e., $\eta \leq 1/L_p$. Recall that $1/(1+\pi) = 1-\epsilon$, which implies that $\pi = \epsilon/(1-\epsilon)$. Combining this with (A.72b), we obtain:

$$\begin{aligned}&R_\pi \geq \frac{\max(2,\delta^{-1})}{\Theta}\log\left(\frac{C_0T\Gamma A}{\pi\gamma_{\min}^{\mathrm{gap}}}\right), \quad \text{where} \quad C_0 \geq 64\pi, \\ &\Rightarrow R_\epsilon \geq \frac{\max(2,\delta^{-1})}{\Theta}\log\left(\frac{(1-\epsilon)C_0T\Gamma A}{\epsilon\gamma_{\min}^{\mathrm{gap}}}\right), \quad \text{where} \quad C_0 \geq 64\frac{\epsilon}{1-\epsilon}.\end{aligned}$$

On the other hand, at the initialization, we have $\epsilon = \mu/2$ which implies that

$$R_\mu^0 \geq \frac{\max(2,\delta^{-1})}{\Theta}\log\left(\frac{(2-\mu)C_0T\Gamma A}{\mu\gamma_{\min}^{\mathrm{gap}}}\right), \quad \text{where} \quad C_0 \geq 64\frac{\mu}{(2-\mu)}. \tag{A.85}$$

In the following, we will determine a lower bound on C_0 such that our step size condition in Theorem 2.3's statement, i.e., $\eta \leq 1/L_p$, is satisfied. Note that for the choice of η in (A.84) to meet the condition $\eta \leq 1/L_p$, the following condition must hold:

$$\frac{1}{L_p} \leq \frac{\mu}{(2-\mu)}\frac{1}{C_2T}e^{R_\mu^0\Theta/2} \Rightarrow R_\mu^0 \geq \frac{2}{\Theta}\log\left(\frac{1}{L_p}\frac{(2-\mu)}{\mu}C_2T\right), \tag{A.86}$$

where $C_2 = (1-\mu)\frac{\bar{A}^2C^2}{\Theta c}$.

This together with (A.85) implies that for sufficiently large

$$R_\mu^0 \geq \frac{\max(2,\delta^{-1})}{\Theta}\log\left(\frac{(2-\mu)C_3T}{\mu}\right), \quad \text{where} \quad C_3 = \frac{C_0\Gamma A}{\gamma_{\min}^{\text{gap}}} \vee \frac{C_2}{L_p},$$

the step size bound in (A.84) ensures that $\eta \leq 1/L_p$ guarantees (A.83). Hence, $\boldsymbol{p}(t+1)$ remains within the cone, i.e., $\boldsymbol{p}(t+1) \in \mathcal{C}_{\mu,R_\epsilon}(\boldsymbol{p}^{\text{mm}})$.

Step 4: The correlation of $\boldsymbol{p}(t)$ and $\boldsymbol{p}^{\text{mm}}$ increases over t. The remainder is similar to the proof of Theorem 2.2. From Step 3, we have that all iterates remain within the initial conic set i.e. $\boldsymbol{p}(t) \in \mathcal{C}_{\mu,R_\mu^0}(\boldsymbol{p}^{\text{mm}})$ for all $t \geq 0$. Note that it follows from Lemma A.6 that $\langle \nabla\mathcal{L}(\boldsymbol{p}), \boldsymbol{p}^{\text{mm}}/\|\boldsymbol{p}^{\text{mm}}\|\rangle < 0$, for any finite $\boldsymbol{p} \in \mathcal{C}_{\mu,R_\mu^0}(\boldsymbol{p}^{\text{mm}})$. Hence, there are no finite critical points $\boldsymbol{p} \in \mathcal{C}_{\mu,R_\mu^0}(\boldsymbol{p}^{\text{mm}})$, for which $\nabla\mathcal{L}(\boldsymbol{p}) = 0$. Now, based on Lemma A.4, which guarantees that $\nabla\mathcal{L}(\boldsymbol{p}(t)) \to 0$, this implies that $\|\boldsymbol{p}(t)\| \to \infty$. Consequently, for any choice of $\epsilon \in (0, \mu/2)$ there is a time t_ϵ such that, for all $t \geq t_\epsilon$, $\boldsymbol{p}(t) \in \mathcal{C}_{\mu,R_\epsilon}(\boldsymbol{p}^{\text{mm}})$. Once within $\mathcal{C}_{\mu,R_\epsilon}(\boldsymbol{p}^{\text{mm}})$, following similar steps in (A.79) and (A.80), for any $t \geq t_\epsilon$,

$$\left\langle \frac{\boldsymbol{p}(t)}{\|\boldsymbol{p}(t)\|}, \frac{\boldsymbol{p}^{\text{mm}}}{\|\boldsymbol{p}^{\text{mm}}\|} \right\rangle \geq 1 - \epsilon + \frac{C_2(\epsilon,\eta)}{\|\boldsymbol{p}(t)\|}, \quad \boldsymbol{p}(t) \in \mathcal{C}_{\mu,R_\epsilon}(\boldsymbol{p}^{\text{mm}}),$$

for some finite constant $C_2(\epsilon,\eta)$. Consequently,

$$\liminf_{t\to\infty} \left\langle \frac{\boldsymbol{p}(t)}{\|\boldsymbol{p}(t)\|}, \frac{\boldsymbol{p}^{\text{mm}}}{\|\boldsymbol{p}^{\text{mm}}\|} \right\rangle \geq 1 - \epsilon, \quad \text{where} \quad \boldsymbol{p}(t) \in \mathcal{C}_{\mu,R_\epsilon}(\boldsymbol{p}^{\text{mm}}).$$

Since the choice of $\epsilon \in (0, \mu/2)$ is arbitrary, we obtain $\boldsymbol{p}(t)/\|\boldsymbol{p}(t)\| \to \boldsymbol{p}^{\text{mm}}/\|\boldsymbol{p}^{\text{mm}}\|$. ∎

A.2.6 Proof of Theorem 2.4

A.2.6.1 Supporting Lemma

We present a lemma that will aid in simplifying our analysis. We begin with a definition.

Definition A.1 *Let $\boldsymbol{q} \in \mathbb{R}^d - \{\mathbf{0}\}$ and $(y_i, \boldsymbol{K}_i, \boldsymbol{X}_i)_{i=1}^n$ be our dataset. We define the (possibly non-unique) selected-tokens of $\boldsymbol{q}$ as follows:*[2]

$$\alpha_i \in \arg\max_{t\in[T]} \boldsymbol{k}_{it}^\top \boldsymbol{q}. \tag{A.87}$$

Next, we define the margin and directional-neighbors for $\boldsymbol{q}$ as the minimum margin tokens

[2] If α_i is unique for all $i \in [n]$, let us call it, unique selected tokens.

to the selected-tokens, i.e.,

$$\Gamma_{\boldsymbol{q}} = \min_{i\in[n],t\neq\alpha_i} (\boldsymbol{k}_{i\alpha_i} - \boldsymbol{k}_{it})^\top \boldsymbol{q}, \tag{A.88}$$

$$\mathcal{M}_{\boldsymbol{q}} = \left\{(i,t) \;\middle|\; (\boldsymbol{k}_{i\alpha_i} - \boldsymbol{k}_{it})^\top \boldsymbol{q} = \Gamma_{\boldsymbol{q}}\right\}. \tag{A.89}$$

Finally, we say that $\boldsymbol{q}$ is neighbor-optimal if the scores of its directional-neighbors are strictly less than the corresponding selected-token. Concretely, for all $(i,t) \in \mathcal{M}_{\boldsymbol{q}}$, we require that

$$\gamma_{it} = y_i \cdot \boldsymbol{x}_{it}^\top \boldsymbol{v} < \gamma_{i\alpha_i} = y_i \cdot \boldsymbol{x}_{i\alpha_i}^\top \boldsymbol{v}.$$

Lemma A.8 (When does one direction dominate another?) *Suppose $\boldsymbol{q}, \boldsymbol{p} \in \mathbb{R}^d$ be two unit Euclidean norm vectors with identical selected tokens. Specifically, for each $i \in [n]$, there exists unique $\alpha_i \in [T]$ such that $\alpha_i = \arg\max_{t\in[T]} \boldsymbol{k}_{it}^\top \boldsymbol{q} = \arg\max_{t\in[T]} \boldsymbol{k}_{it}^\top \boldsymbol{p}$. Suppose directional margins obey $\Gamma_{\boldsymbol{q}} < \Gamma_{\boldsymbol{p}}$ and set $\delta_\Gamma = \Gamma_{\boldsymbol{p}} - \Gamma_{\boldsymbol{q}}$.*

- *Suppose $\boldsymbol{q}$ and $\boldsymbol{p}$ are both neighbor-optimal. Then, for some $R(\delta_\Gamma)$ and all $R > R(\delta_\Gamma)$, we have that $\mathcal{L}(R \cdot \boldsymbol{p}) < \mathcal{L}(R \cdot \boldsymbol{q})$.*
- *Suppose $\boldsymbol{q}$ has a unique directional-neighbor and is not neighbor-optimal (i.e. this neighbor has higher score). Let $\delta_{\boldsymbol{q}}$ be the margin difference between unique directional-neighbor and the second-most minimum-margin neighbor (i.e. the one after the unique one, see (A.93)) of $\boldsymbol{q}$. Then, for some $R(\delta_\Gamma \wedge \delta_{\boldsymbol{q}})$ and all $R > R(\delta_\Gamma \wedge \delta_{\boldsymbol{q}})$, we have that $\mathcal{L}(R \cdot \boldsymbol{q}) < \mathcal{L}(R \cdot \boldsymbol{p})$.*

Proof. We prove these two statements in order. First define the directional risk baseline induced by letting $R \to \infty$ and purely selecting the tokens $\boldsymbol{\alpha} = (\alpha_i)_{i=1}^n$. This is given by

$$\mathcal{L}_\star := \frac{1}{n}\sum_{i=1}^n \ell\left(y_i \cdot \boldsymbol{v}^\top \boldsymbol{x}_{i\alpha_i}\right).$$

We evaluate $\boldsymbol{q}, \boldsymbol{p}$ with respect to $\mathcal{L}_\star$. To proceed, let $\boldsymbol{s}_i = \mathbb{S}(R\boldsymbol{K}_i\boldsymbol{q})$. Define $\Gamma_{\boldsymbol{q}}^{it} = \boldsymbol{k}_{i\alpha_i}^\top \boldsymbol{q} - \boldsymbol{k}_{it}^\top \boldsymbol{q}$. Note that, the smallest value for $t \neq \alpha_i$ is achieved for $\Gamma_{\boldsymbol{q}}$. For sufficiently large $R \gtrsim \mathcal{O}(\log(T)/\Gamma_{\boldsymbol{q}})$, observe that, for $t \neq \alpha_i$

$$e^{-R\Gamma_{\boldsymbol{q}}^{it}} \geq \boldsymbol{s}_{it} = \frac{e^{R\boldsymbol{k}_{it}^\top \boldsymbol{q}}}{\sum_{t\in[T]} e^{R\boldsymbol{k}_{it}^\top \boldsymbol{q}}} \geq 0.5 e^{-R\Gamma_{\boldsymbol{q}}^{it}}. \tag{A.90}$$

Recalling the score definition and let M_+, M_- be the upper and lower bounds on $-\ell'$ over its bounded domain that scores fall on, respectively. Note that, for some intermediate

$M_+ \geq M_i \geq M_-$ values, we have

$$\begin{aligned}\mathcal{L}(R\boldsymbol{q}) - \mathcal{L}_\star &= \frac{1}{n}\sum_{i=1}^{n} \ell(\sum_{t\in[T]} \boldsymbol{s}_{it}\gamma_{it}) - \ell(\gamma_{i\alpha_i}) \\ &= \frac{1}{n}\sum_{i=1}^{n} M_i \sum_{t\neq\alpha_i} \boldsymbol{s}_{it}(\gamma_{i\alpha_i} - \gamma_{it}).\end{aligned}$$

Now, using (A.90) for a refreshed $M_+ \geq M_{it} \geq 0.5M_-$ values, we can write

$$\mathcal{L}(R\boldsymbol{q}) - \mathcal{L}_\star = \frac{1}{n}\sum_{i=1}^{n}\sum_{t\neq\alpha_i} M_{it} e^{-R\Gamma_{\boldsymbol{q}}^{it}}(\gamma_{i\alpha_i} - \gamma_{it}). \tag{A.91}$$

The same bound also applies to $\boldsymbol{p}$ with some M'_{it}, multipliers

$$\mathcal{L}(R\boldsymbol{p}) - \mathcal{L}_\star = \frac{1}{n}\sum_{i=1}^{n}\sum_{t\neq\alpha_i} M'_{it} e^{-R\Gamma_{\boldsymbol{p}}^{it}}(\gamma_{i\alpha_i} - \gamma_{it}). \tag{A.92}$$

We can now proceed with the proof.

Case 1: $\boldsymbol{q}$ and $\boldsymbol{p}$ are both neighbor-optimal. This means that $\gamma_{i\alpha_i} - \gamma_{it} > 0$ for all $i \in [n], t \neq \alpha_i$. Let $K_+ > K_- > 0$ be upper and lower bounds on $\gamma_{i\alpha_i} - \gamma_{it}$ values. We can now upper bound the right hand side of (A.91) via

$$M_+K_+Te^{-R\Gamma_{\boldsymbol{q}}} \geq \mathcal{L}(R\boldsymbol{q}) - \mathcal{L}_\star \geq \frac{1}{2n}M_-K_-e^{-R\Gamma_{\boldsymbol{q}}}.$$

Consequently, $\mathcal{L}(R\boldsymbol{q}) > \mathcal{L}(R\boldsymbol{p})$ as soon as $\frac{1}{2n}M_-K_-e^{-R\Gamma_{\boldsymbol{q}}} > M_+K_+Te^{-R\Gamma_{\boldsymbol{p}}}$. Since M_+, K_+, n, T are global constants, this happens under the stated condition on the margin gap $\Gamma_{\boldsymbol{p}} - \Gamma_{\boldsymbol{q}}$.

Case 2: $\boldsymbol{q}$ has a unique directional-neighbor and is not neighbor-optimal. In this scenario, $\mathcal{L}(R\boldsymbol{q}) - \mathcal{L}_\star$ is actually negative for large R. To proceed, define the maximum score difference $K_+ = \sup_{i,t\neq\alpha_i} |\gamma_{i\alpha_i} - \gamma_{it}|$. Also let (j, β) be the unique directional neighbor achieving the minimum margin $\Gamma_{\boldsymbol{q}}$. Then, $\delta_{\boldsymbol{q}}$ – the margin difference between unique directional-neighbor and the second minimum-margin neighbor (i.e. the one after the unique one) of $\boldsymbol{q}$ – is defined as

$$\delta_{\boldsymbol{q}} = \min_{i\in[n],\ t\neq\alpha_i,\ (i,t)\neq(j,\beta)} \Gamma_{\boldsymbol{q}}^{it} - \Gamma_{\boldsymbol{q}}. \tag{A.93}$$

To proceed, we can write

$$\mathcal{L}(R\boldsymbol{p}) - \mathcal{L}_\star \geq -M_+K_+Te^{-R\Gamma_{\boldsymbol{p}}}.$$

On the other hand, setting $\kappa = \gamma_{j\beta} - \gamma_{j\alpha_j} > 0$, we can bound

$$\begin{aligned}\mathcal{L}(R\boldsymbol{q}) - \mathcal{L}_\star &= -\frac{1}{n} M_{j\beta} e^{-R\Gamma_{\boldsymbol{q}}}\kappa + \frac{1}{n}\sum_{i\in[n],\ t\neq\alpha_i,\ (i,t)\neq(j,\beta)} M_{it} e^{-R\Gamma_{\boldsymbol{q}}^{it}}(\gamma_{i\alpha_i} - \gamma_{it}) \\ &\leq -\frac{1}{n} M_- e^{-R\Gamma_{\boldsymbol{q}}}\kappa + M_+ K_+ T e^{-R(\Gamma_{\boldsymbol{q}}+\delta_{\boldsymbol{q}})}.\end{aligned}$$

Consequently, we have found that $\mathcal{L}(R\boldsymbol{p}) > \mathcal{L}(R\boldsymbol{q})$ as soon as

$$\frac{1}{n} M_- e^{-R\Gamma_{\boldsymbol{q}}}\kappa \geq M_+ K_+ T(e^{-R(\Gamma_{\boldsymbol{q}}+\delta_{\boldsymbol{q}})} + e^{-R\Gamma_{\boldsymbol{p}}})$$

This happens when $R \gtrsim \frac{1}{\delta_{\boldsymbol{q}} \wedge (\Gamma_{\boldsymbol{p}} - \Gamma_{\boldsymbol{q}})}$ (up to logarithmic terms) establishing the desired statement. ∎

A.2.6.2 Proof of Theorem 2.4

Define the locally-optimal unit directions

$$\mathcal{P}^{\mathrm{mm}} = \left\{ \frac{\boldsymbol{p}^{\mathrm{mm}}(\boldsymbol{\alpha})}{\|\boldsymbol{p}^{\mathrm{mm}}(\boldsymbol{\alpha})\|} \;\middle|\; \boldsymbol{\alpha} \text{ is a locally-optimal set of indices} \right\}.$$

The theorem below shows that cone-restricted regularization paths can only directionally converge to an element of this set.

Theorem A.1 (Non-LOMM Regularization Paths Fail) *Fix a unit Euclidean norm vector $\boldsymbol{q} \in \mathbb{R}^d$ such that $\boldsymbol{q} \notin \mathcal{P}^{mm}$. Assume that the token scores are distinct (i.e., $\gamma_{it} \neq \gamma_{i\tau}$ for $t \neq \tau$) and the key embeddings $\boldsymbol{k}_{it}$ are in general position. Specifically, we require the following conditions to hold* [3]*:*

- *When $m = d$, all matrices $\bar{\boldsymbol{K}} \in \mathbb{R}^{m\times d}$ where each row of $\bar{\boldsymbol{K}}$ has the form $\boldsymbol{k}_{it} - \boldsymbol{k}_{i\alpha_i}$ for a unique $(i, \alpha_i, t \neq \alpha_i)$ tuple, are full-rank.*
- *When $m = d+1$, the vector of all ones is not in the range space of any such $\bar{\boldsymbol{K}}$ matrix.*

Fix arbitrary $\epsilon > 0, R_0 > 0$. Define the local regularization path of $\boldsymbol{q}$ as its (ϵ, R_0)-conic neighborhood:

$$\bar{\boldsymbol{p}}(R) = \operatorname*{arg\,min}_{\boldsymbol{p}\in\mathcal{C}_{\epsilon,R_0}(\boldsymbol{q}),\|\boldsymbol{p}\|\leq R} \mathcal{L}(\boldsymbol{p}), \quad \textit{where} \quad \mathcal{C}_{\epsilon,R_0}(\boldsymbol{q}) = \texttt{cone}_\epsilon(\boldsymbol{q}) \cap \left\{\boldsymbol{p}\in\mathbb{R}^d \middle|\ \|\boldsymbol{p}\| \geq R_0\right\}. \quad \text{(A.94)}$$

[3] This requirement holds for general data because it is guaranteed by adding arbitrarily small independent gaussian perturbations to keys $\boldsymbol{k}_{it}$.

Then, either $\lim_{R\to\infty} \|\bar{\boldsymbol{p}}(R)\| < \infty$ *or* $\lim_{R\to\infty} \bar{\boldsymbol{p}}(R)/\|\bar{\boldsymbol{p}}(R)\| \neq \boldsymbol{q}$. *In both scenarios* $\lim_{R\to\infty} \bar{\boldsymbol{p}}(R)/R \neq \boldsymbol{q}$.

Proof. We will prove the result by dividing the problem into distinct cases. In each case, we will construct an alternative direction that achieves a strictly better objective than some $\delta = \delta(\epsilon) > 0$ neighborhood of $\boldsymbol{q}$, thereby demonstrating the suboptimality of the $\boldsymbol{q}$ direction. Let's define the δ neighborhood as follows:

$$\mathcal{N}_\delta = \left\{ \boldsymbol{p} \mid \left\| \frac{\boldsymbol{p}}{\|\boldsymbol{p}\|} - \boldsymbol{q} \right\| \leq \delta \quad \text{and} \quad \|\boldsymbol{p}\| \geq R_0 \right\}. \tag{A.95}$$

Now, let's recall a few more definitions based on Definition A.1. First, the tokens selected by $\boldsymbol{q}$ are given by (A.87). To proceed, let's initially consider the scenario where α_i is unique for all $i \in [n]$, meaning that as we let $c \to \infty$, $c \cdot \boldsymbol{q}$ will choose a single token per input. Later, we will revisit the setting when $\arg\max$ is not a singleton, and $\boldsymbol{q}$ is allowed to select multiple tokens.

Additionally, it's important to note that $\|\bar{\boldsymbol{p}}(R)\|$ is non-decreasing by definition. Suppose it has a finite upper bound $\|\bar{\boldsymbol{p}}(R)\| \leq M$ for all $R < \infty$. In that scenario, we have $\lim_{R\to\infty} \frac{\bar{\boldsymbol{p}}(R)}{R} = 0 \neq \boldsymbol{q}$.

• **(A) $\boldsymbol{q}$ selects a single token per input:** Given that the indices $\boldsymbol{\alpha} = (\alpha_i)_{i=1}^n$ defined in (A.87) are uniquely determined, we can conclude that the $\boldsymbol{q}$ direction eventually selects tokens $\boldsymbol{k}_{i\alpha_i}$. Recall the definition of the margin $\Gamma_{\boldsymbol{q}}$ from (A.88) and the set of directional neighbors, which is defined as the indices that achieve $\Gamma_{\boldsymbol{q}}$, as shown in (A.89). Let us refer to $\boldsymbol{q}$ as *neighbor-optimal* if $\gamma_{it} < \gamma_{i\alpha_i}$ for all $(i,t) \in \mathcal{M}_{\boldsymbol{q}}$.

We will consider two cases for this scenario: when $\boldsymbol{q}$ is neighbor-optimal and when $\boldsymbol{q}$ is not neighbor-optimal.

◇ **(A1) $\boldsymbol{q}$ is neighbor-optimal.** In this case, we will argue that max-margin direction $\bar{\boldsymbol{p}}^{\mathrm{mm}} := \boldsymbol{p}^{\mathrm{mm}}(\boldsymbol{\alpha})/\|\boldsymbol{p}^{\mathrm{mm}}(\boldsymbol{\alpha})\|$ can be used to construct a strictly better objective than $\boldsymbol{q}$. Note that $\boldsymbol{p}^{\mathrm{mm}}(\boldsymbol{\alpha})$ exists because $\boldsymbol{q}$ is already a viable separating direction for tokens $\boldsymbol{\alpha}$. Specifically, consider the direction $\boldsymbol{q}' = \frac{\boldsymbol{q}+\epsilon\bar{\boldsymbol{p}}^{\mathrm{mm}}}{\|\boldsymbol{q}+\epsilon\bar{\boldsymbol{p}}^{\mathrm{mm}}\|}$. Observe that, $\boldsymbol{q}'$ lies within $\mathtt{cone}_{2\epsilon}(\boldsymbol{q})$,[4]

$$\boldsymbol{q}^\top \boldsymbol{q}' \geq \frac{1-\epsilon}{1+\epsilon} \geq 1 - 2\epsilon.$$

We now argue that, there exists $\delta = \delta_\epsilon > 0$ such that for all $R > R_\epsilon$

$$\min_{R_\epsilon \leq r \leq R} \mathcal{L}(r \cdot \boldsymbol{q}') < \min_{\boldsymbol{p}\in\mathcal{N}_\delta, R_\epsilon \leq \|\boldsymbol{p}\| \leq R} \mathcal{L}(\boldsymbol{p}). \tag{A.96}$$

[4]As a result, let us prove the result for $\epsilon \leftarrow 2\epsilon$ without losing generality.

To prove this, we study the margin $\Gamma_{\boldsymbol{q}'}$ induced by $\boldsymbol{q}'$ and the maximum margin Γ_δ induced within $\boldsymbol{p} \in \mathcal{N}_\delta$. Concretely, we will show that $\Gamma_{\boldsymbol{q}'} > \Gamma_\delta$ and directly apply the first statement of Lemma A.8 to conclude with (A.96).

Let $\Gamma = 1/\|\boldsymbol{p}^{\mathrm{mm}}(\boldsymbol{\alpha})\|$ be the margin induced by $\bar{\boldsymbol{p}}^{\mathrm{mm}}$. Note that $\Gamma > \Gamma_{\boldsymbol{q}}$ by the optimality of $\boldsymbol{p}^{\mathrm{mm}}(\boldsymbol{\alpha})$ and the fact that $\boldsymbol{q} \neq \boldsymbol{p}^{\mathrm{mm}}(\boldsymbol{\alpha})$. Consequently, we can lower and upper bound the margins via

$$\begin{aligned}
\Gamma_{\boldsymbol{q}'} &= \min_{i\in[n]} \min_{t\neq\alpha_i} (\boldsymbol{k}_{i\alpha_i} - \boldsymbol{k}_{it})^\top \boldsymbol{q}' \geq \frac{\Gamma_{\boldsymbol{q}} + \epsilon\Gamma}{1+\epsilon} \geq \Gamma_{\boldsymbol{q}} + \frac{\epsilon}{2}(\Gamma - \Gamma_{\boldsymbol{q}}), \\
\Gamma_\delta &= \max_{\boldsymbol{p}\in\mathcal{N}_\delta} \min_{i\in[n]} \min_{t\neq\alpha_i} (\boldsymbol{k}_{i\alpha_i} - \boldsymbol{k}_{it})^\top \boldsymbol{p}/\|\boldsymbol{p}\| \\
&\leq \max_{\|\boldsymbol{r}\|\leq 1} \min_{i\in[n]} \min_{t\neq\alpha_i} (\boldsymbol{k}_{i\alpha_i} - \boldsymbol{k}_{it})^\top (\boldsymbol{q} + \delta\boldsymbol{r}) \\
&\leq \Gamma_{\boldsymbol{q}} + M\delta,
\end{aligned}$$

where $M = \max_{i,t,\tau} \|\boldsymbol{k}_{it} - \boldsymbol{k}_{i\tau}\|$.

Consequently, setting $\delta = \frac{\epsilon}{4M}(\Gamma - \Gamma_{\boldsymbol{q}})$, we find that

$$\Gamma_{\boldsymbol{q}'} \geq \Gamma_\delta + \frac{\epsilon}{4}(\Gamma - \Gamma_{\boldsymbol{q}}).$$

Equipped with this inequality, we apply the first statement of Lemma A.8 which concludes that[5] for some $R_\epsilon = R(\frac{\epsilon}{4}(\Gamma - \Gamma_{\boldsymbol{q}}))$ and all $R > R_\epsilon$, (A.96) holds. This in turn implies that, within $\mathcal{C}_\epsilon$, the optimal solution is

- either upper bounded by R_ϵ in ℓ_2 norm (i.e. $\lim_{R\to\infty} \|\bar{\boldsymbol{p}}(R)\| < \infty$) or
- at least $\delta = \delta(\epsilon) > 0$ away from $\boldsymbol{q}$ after ℓ_2-normalization i.e. $\|\frac{\bar{\boldsymbol{p}}(R)}{\|\bar{\boldsymbol{p}}(R)\|} - \boldsymbol{q}\| \geq \delta$.

In either scenario, we have proven that $\lim_{R\to\infty} \frac{\bar{\boldsymbol{p}}(R)}{R} \neq \boldsymbol{q}$.

$\diamond$ **(A2) $\boldsymbol{q}$ is not neighbor-optimal.** In this scenario, we will prove that $\bar{\boldsymbol{p}}(R)$ is finite to obtain $\lim_{R\to\infty} \bar{\boldsymbol{p}}(R)/R = 0 \neq \boldsymbol{q}$. To start, assume that conic neighborhood ϵ of $\boldsymbol{q}$ is small enough so that selected-tokens $\boldsymbol{\alpha}$ remain unchanged within $\mathcal{C}_\epsilon$. This is without generality because if directional convergence fails in a small neighborhood of $\boldsymbol{q}$, it will fail in the larger neighborhood as well. Secondly, if $\lim_{R\to\infty} \|\bar{\boldsymbol{p}}(R)\| \to \infty$ and $\bar{\boldsymbol{p}}(R) \in \mathcal{C}_\epsilon$, since softmax will eventually perfectly select $\boldsymbol{\alpha}$ (i.e. assigning probability 1 on token indices (i, α_i)), we would have

$$\lim_{R\to\infty} \mathcal{L}(\bar{\boldsymbol{p}}(R)) = \mathcal{L}_\star = \frac{1}{n}\sum_{i=1}^n \ell(\gamma_{i\alpha_i}).$$

[5]Here, we apply this lemma to compare $\boldsymbol{q}'$ against all $\boldsymbol{p} \in \mathcal{N}_\delta$. We can do this uniform comparison because the R requirement in Lemma A.8 only depends on the margin difference and global problem variables and not the particular choice of $\boldsymbol{p} \in \mathcal{N}_\delta$.

Note that, this is simply by selection of $\boldsymbol{\alpha}$ and regardless of $\bar{\boldsymbol{p}}(R)$ directionally converges to $\boldsymbol{q}$. This means that, if there exists a finite $\boldsymbol{p} \in \mathcal{C}_\epsilon$ such that $\mathcal{L}(\boldsymbol{p}) < \mathcal{L}_\star$ (i.e. outperforming the training loss of $\|\bar{\boldsymbol{p}}(R)\| \to \infty$), then $\|\bar{\boldsymbol{p}}(R)\| < \infty$. This would conclude the proof.

Thus, we will simply find such a $\boldsymbol{p}$ obeying $\mathcal{L}(\boldsymbol{p}) < \mathcal{L}_\star$. To this aim, we first prove the following lemma.

Lemma A.9 *Given a fixed unit Euclidean norm vector $\boldsymbol{p}$, if all directional neighbors of $\boldsymbol{p}$ consistently have higher scores for their associated selected tokens, i.e., $\gamma_{i\alpha_i} < \gamma_{i\beta}$ for all (i, α_i) and directional neighbor (i, β), then there exists $\bar{R}$ such that for all $R > \bar{R}$,*

$$\mathcal{L}(R \cdot \boldsymbol{p}) < \mathcal{L}_\star = \lim_{R \to \infty} \mathcal{L}(R \cdot \boldsymbol{p}). \tag{A.97}$$

Proof. Define the maximum score difference $K_+ = \sup_{i \in [n], t \neq \alpha_i} |\gamma_{i\alpha_i} - \gamma_{it}|$. Also let $\mathcal{M}_{\boldsymbol{p}}$ be the set of directional neighbors achieving the minimum margin $\Gamma_{\boldsymbol{p}}$; see (A.89). Define $\Gamma^{it} = \boldsymbol{k}_{i\alpha_i}^\top \boldsymbol{p} - \boldsymbol{k}_{it}^\top \boldsymbol{p}$. Define $\delta_{\boldsymbol{p}}$ to be the margin difference between the directional-neighbors and the second-most minimum-margin neighbors defined as

$$\delta_{\boldsymbol{p}} = \min_{i \in [n], t \neq \alpha_i, (i,t) \notin \mathcal{M}_{\boldsymbol{p}}} \Gamma_{\boldsymbol{p}}^{it} - \Gamma_{\boldsymbol{p}}. \tag{A.98}$$

To proceed, setting $\kappa = \min_{(j,\beta) \in \mathcal{M}_{\boldsymbol{p}}} \gamma_{j\beta} - \gamma_{j\alpha_j} > 0$ and using (A.91), we can bound

$$\begin{aligned}
\mathcal{L}(R\boldsymbol{p}) - \mathcal{L}_\star &\leq -\frac{1}{n} \sum_{(j,\beta) \in \mathcal{M}_{\boldsymbol{p}}} M_{j\beta}\, e^{-R\Gamma_{\boldsymbol{p}}\kappa} + \frac{1}{n} \sum_{i \in [n], t \neq \alpha_i, (i,t) \notin \mathcal{M}_{\boldsymbol{p}}} M_{it}\, e^{-R\Gamma_{\boldsymbol{p}}^{it}} (\gamma_{i\alpha_i} - \gamma_{it}) \\
&\leq -\frac{1}{n} M_- e^{-R\Gamma_{\boldsymbol{p}}\kappa} + M_+ K_+ T e^{-R(\Gamma_{\boldsymbol{p}} + \delta_{\boldsymbol{p}})}.
\end{aligned}$$

Consequently, we have found that $\mathcal{L}(R\boldsymbol{p}) < \mathcal{L}_\star$ as soon as

$$\frac{1}{n} M_- e^{-R\Gamma_{\boldsymbol{q}}\kappa} \geq M_+ K_+ T e^{-R(\Gamma_{\boldsymbol{q}} + \delta_{\boldsymbol{q}})}.$$

This happens when $R \gtrsim 1/\delta_{\boldsymbol{q}}$ (up to logarithmic terms) establishing the desired statement. ■

Based on this lemma, what we need is constructing a perturbation to modify $\boldsymbol{q}$'s directional neighbors and make sure all of them have strictly better scores than their associated selected-tokens. Note that, once we construct a new candidate (say $\boldsymbol{q}_0 = \boldsymbol{q} + \text{perturbation}$), all sufficiently large scalings of $\boldsymbol{q}_0$ will achieve $\mathcal{L}(R \cdot \boldsymbol{q}_0) < \mathcal{L}_\star$. Thus, we can find a strictly better solution than $\mathcal{L}_\star$ for any norm lower bound R_0 – which is enforced within the definition of $\mathcal{C}_\epsilon$.

Lemma A.10 *There are at most d directional neighbors i.e. $|\mathcal{M}_{\boldsymbol{q}}| \leq d$.*

Proof. Directional neighbors are indices (i,t) obeying the inequality

$$(\boldsymbol{k}_{i\alpha_i} - \boldsymbol{k}_{it})^\top \boldsymbol{q} = \Gamma_{\boldsymbol{q}}.$$

Declare $\boldsymbol{D}$ to be the matrix with rows obtained by these key differences $\boldsymbol{k}_{it} - \boldsymbol{k}_{i\alpha_i}$. $\boldsymbol{D} \in \mathbb{R}^{M\times d}$ where $M = |\mathcal{M}_{\boldsymbol{q}}|$. We then obtain $\boldsymbol{D}\boldsymbol{q} = -\Gamma_{\boldsymbol{q}}\mathbf{1}_M$ where $\mathbf{1}_M$ is the all ones vector. If $M > d$, the equality $\boldsymbol{D}\boldsymbol{q} = -\Gamma_{\boldsymbol{q}}\mathbf{1}_M$ cannot be satisfied because $\mathbf{1}_M$ is not in the range space of $\boldsymbol{D}$ by our assumption of general key embedding positions. ■

To proceed, $|\mathcal{M}_{\boldsymbol{q}}| \leq d$ and let $\boldsymbol{D}$ be as defined in the lemma above. $\boldsymbol{D}$ is also full-rank by our assumption of general key positions. We use $\boldsymbol{D}$ to construct a perturbation as follows. Let $\mathcal{M}_{\boldsymbol{q}}^+ \subset \mathcal{M}_{\boldsymbol{q}}$ be the set of directional neighbor with strictly higher scores than their associated selected-tokens. In other words, all $(j,\beta) \in \mathcal{M}_{\boldsymbol{q}}^+$ obeys

$$\gamma_{j\beta} > \gamma_{j\alpha_j}.$$

Define the score difference $\kappa = \min_{(j,\beta)\in\mathcal{M}_{\boldsymbol{q}}^+} \gamma_{j\beta} - \gamma_{j\alpha_j} > 0$. We know $\kappa > 0$ because $\boldsymbol{\alpha}$ is not neighbor-optimal. Finally, define the indicator vector of $\mathbf{1}_+$ with same dimension as cardinality $|\mathcal{M}_{\boldsymbol{q}}|$. $\mathbf{1}_+$ is 1 for the rows of $\boldsymbol{D}$ corresponding to $\mathcal{M}_{\boldsymbol{q}}^+$ and is 0 otherwise. Finally, set the perturbation as

$$\boldsymbol{q}^\perp = \boldsymbol{D}^\dagger \mathbf{1}_+.$$

where we used the full-rankness of $\boldsymbol{D}$ during pseudo-inversion. To proceed, for a small $\epsilon_0 > 0$, consider the candidate direction $\boldsymbol{q}_0 = \boldsymbol{q} + \epsilon_0\boldsymbol{q}^\perp$. We pick $\epsilon_0 = \mathcal{O}(\epsilon)$ sufficiently small to ensure $\boldsymbol{q}_0 \in \mathcal{C}_\epsilon$. To finalize, let us consider the margins of the tokens within $\boldsymbol{q}_0$. Similar to Lemma A.9, set $\delta_{\boldsymbol{q}} = \min_{i\in[n],t\neq\alpha_i,(i,t)\notin\mathcal{M}_{\boldsymbol{q}}} \Gamma_{\boldsymbol{q}}^{it} - \Gamma_{\boldsymbol{q}} > 0$. Let $\bar{\epsilon}_0 = \|\boldsymbol{q}_0\| - 1 = \|\boldsymbol{q} + \epsilon_0\boldsymbol{q}^\perp\| - 1$. Using definition of $\boldsymbol{q}^\perp$, we have that

- For $(i,t) \in \mathcal{M}_{\boldsymbol{q}}^+$, we achieve a margin of

$$(\boldsymbol{k}_{i\alpha_i} - \boldsymbol{k}_{it})^\top(\boldsymbol{q} + \epsilon_0\boldsymbol{q}^\perp)/(1+\bar{\epsilon}_0) = \frac{\Gamma_{\boldsymbol{q}} - \epsilon_0}{1+\bar{\epsilon}_0}.$$

- For $(i,t) \in \mathcal{M}_{\boldsymbol{q}}^+$, we achieve a margin of $\frac{\Gamma_{\boldsymbol{q}}}{1+\bar{\epsilon}_0}$.
- For $(i,t) \notin \mathcal{M}_{\boldsymbol{q}}$, setting $K = \|\boldsymbol{q}^\perp\| \cdot \sup_{i,t,\tau} \|\boldsymbol{k}_{i\alpha_i} - \boldsymbol{k}_{it}\|$, we achieve a margin of at most

$$(\boldsymbol{k}_{i\alpha_i} - \boldsymbol{k}_{it})^\top(\boldsymbol{q} + \epsilon_0\boldsymbol{q}^\perp)/(1+\bar{\epsilon}_0) \geq \frac{\Gamma_{\boldsymbol{q}} + \delta_{\boldsymbol{q}} - \epsilon_0 K}{1+\bar{\epsilon}_0}.$$

In short, since $\bar{\epsilon}_0 = \mathcal{O}(\epsilon_0)$, setting ϵ_0 sufficiently small guarantees that $\mathcal{M}_{\boldsymbol{q}}^+$ is the set of directional neighbors of $\boldsymbol{q}_0$. Since $\mathcal{M}_{\boldsymbol{q}}^+$ has strictly higher scores than their associated selected-tokens, applying Lemma A.9 on $\boldsymbol{q}_0$ shows that, $\mathcal{L}(R{\cdot}\boldsymbol{q}_0) < \mathcal{L}_\star$ for sufficiently large R implying $\|\bar{\boldsymbol{p}}(R)\| < \infty$.

• **(B) $\boldsymbol{q}$ selects multiple tokens for some inputs** $i \in [n]$**:** In this setting, we will again construct a perturbation to create a scenario where $\boldsymbol{q}_0 = \boldsymbol{q} + \epsilon \boldsymbol{q}^\perp$ selects a single token for each input $i \in [n]$. We will then employ margin analysis (first statement of Lemma A.8) to conclude that $\boldsymbol{q}_0$ outperforms a $\delta \ll \epsilon$ neighborhood of $\boldsymbol{q}$.

Let $\mathcal{I} \subset [n]$ be the set of inputs for which $\boldsymbol{q}$ selects multiple tokens. Specifically, for each $i \in \mathcal{I}$, there is $\mathcal{T}_i \subset [T]$ such that $|\mathcal{T}_i| \geq 2$ and for any $i \in \mathcal{I}$ and $\theta \in \mathcal{T}_i$,

$$\boldsymbol{k}_{i\theta}^\top \boldsymbol{q} = \arg\max_{t \in [T]} \boldsymbol{k}_{it}^\top \boldsymbol{q}.$$

From these multiply-selected token indices let us select the highest score one, namely, $\beta_i = \arg\max_{\theta \in \mathcal{T}_i} \gamma_{i\theta}$ for $i \in \mathcal{I}$. Now, define the unique optimal tokens for each input as $\boldsymbol{\alpha} \in \mathbb{R}^n$ where $\alpha_i := \beta_i$ for $i \in \mathcal{I}$ and $\alpha_i = \arg\max_{t \in [T]} \boldsymbol{k}_{it}^\top \boldsymbol{q}$ for $i \notin \mathcal{I}$. Define $\mathcal{L}_\star = \frac{1}{n}\sum_{i=1}^n \ell(\gamma_{i\alpha_i})$ as earlier.

Secondly, we construct a perturbation $\boldsymbol{q}^\perp$ to show that $\boldsymbol{q}_0 = \boldsymbol{q} + \epsilon_0 \boldsymbol{q}^\perp$ can select tokens $\boldsymbol{\alpha}$ asymptotically. To see this, define the matrix $\boldsymbol{D}$ where each (unique) row is given by $\boldsymbol{k}_{i\alpha_i} - \boldsymbol{k}_{i\theta}$ where $\theta \in \mathcal{T}_i, \theta \neq \alpha_i$, $i \in \mathcal{I}$. Now note that, $(\boldsymbol{k}_{i\alpha_i} - \boldsymbol{k}_{i\theta})^\top \boldsymbol{q} = 0$ for all $\theta \in \mathcal{T}_i, \theta \neq \alpha_i$, $i \in \mathcal{I}$. Since keys are in general positions, this implies that $\boldsymbol{D}$ has at most $d-1$ rows and, thus, its rows are linearly independent. Consequently, choose $\boldsymbol{q}^\perp = \boldsymbol{D}^\dagger \boldsymbol{1}$ where $\dagger$ denotes pseudo-inverse and $\boldsymbol{1}$ is the all ones vector. Also let $\Gamma_{\boldsymbol{q}}$ be the margin of directional margin of $\boldsymbol{q}$ that is

$$\Gamma_{\boldsymbol{q}} = \min_{i \in [n], t \notin \mathcal{T}_i} (\boldsymbol{k}_{i\alpha_i} - \boldsymbol{k}_{it})^\top \boldsymbol{q}.$$

With this choice and setting $K = \sup_{i,t,\tau} \|\boldsymbol{k}_{i\alpha_i} - \boldsymbol{k}_{it}\|$, we have that

- For $\theta \in \mathcal{T}_i, \theta \neq \alpha_i$: $(\boldsymbol{k}_{i\alpha_i} - \boldsymbol{k}_{i\theta})^\top \boldsymbol{q}_0 = (\boldsymbol{k}_{i\alpha_i} - \boldsymbol{k}_{i\theta})^\top \boldsymbol{q}^\perp = \epsilon_0$.
- For all other (i,t) with $t \neq \alpha_i$: $(\boldsymbol{k}_{i\alpha_i} - \boldsymbol{k}_{it})^\top \boldsymbol{q}_0 \geq \Gamma_{\boldsymbol{q}} - K\|\boldsymbol{q}^\perp\|\epsilon_0$.

Choosing $\epsilon_0 < \Gamma_{\boldsymbol{q}}/(1 + K\|\boldsymbol{q}^\perp\|)$, together, these imply that,

- $\boldsymbol{\alpha} = (\alpha_i)_{i=1}^n$ is the selected-tokens of $\boldsymbol{q}_0$,
- $\boldsymbol{q}_0$ achieves a directional margin of

$$\Gamma_{\boldsymbol{q}_0} = \frac{\epsilon_0}{\|\boldsymbol{q}_0\|} \geq \frac{\epsilon_0}{1 + \epsilon_0\|\boldsymbol{q}^\perp\|},$$

- $\boldsymbol{q}_0$ is neighbor-optimal because directional neighbors are $\theta \in \mathcal{T}_i, \theta \neq \alpha_i$ and $\gamma_{i\theta} < \gamma_{i\alpha_i}$.

Note that, these conditions lay the groundwork for applying the first statement of Lemma A.8 with $\boldsymbol{p} \leftarrow \boldsymbol{q}_0$. We next explore the optimal directions within $\mathcal{N}_\delta$ and show that $\boldsymbol{q}_0$ strictly outperform them in terms of training loss.

To proceed, given small $\delta \ll \epsilon_0$, let us study $\mathcal{L}_R = \min_{\boldsymbol{p} \in \mathcal{N}_\delta, R \leq \|\boldsymbol{p}\| < \infty} \mathcal{L}(\boldsymbol{p})$. Here, recall that $\boldsymbol{p}$ has a δ-small directional perturbation around $\boldsymbol{q}$ which can modify the token selections by breaking the ties between the multiply-selected token indices by $\boldsymbol{q}$. However, thanks to the distinct token score assumption, for large $R \leq \|\boldsymbol{p}\|$, the optimal $\boldsymbol{p} \in \mathcal{N}_\delta$ is guaranteed to (uniquely) select indices $\alpha_i \in \mathcal{T}_i$. Because all other $\boldsymbol{p}$ directions – which, asymptotically, either select other tokens $\theta \in \mathcal{T}_i, \theta \neq \alpha_i$ or split the probabilities equally across a subset of $\mathcal{T}_i$ – achieve a larger loss. For instance, $\boldsymbol{q}$ will split the probabilities equally across $\mathcal{T}_i$ to achieve an asymptotic loss of

$$\mathcal{L}_\star^{\boldsymbol{q}} := \lim_{R \to \infty} \mathcal{L}(R \cdot \boldsymbol{q}) = \frac{1}{n} \sum_{i \notin \mathcal{I}} \ell(\gamma_{i\alpha_i}) + \frac{1}{n} \sum_{i \in \mathcal{I}} \ell\left(\frac{1}{|\mathcal{T}_i|} \sum_{\theta \in \mathcal{T}_i} \gamma_{i\theta} \right)$$

Thus, $\mathcal{L}_\star^{\boldsymbol{q}} > \mathcal{L}_\star$ because $\ell\left(\frac{1}{|\mathcal{T}_i|} \sum_{\theta \in \mathcal{T}_i} \gamma_{i\theta}\right) > \ell\left(\gamma_{i\alpha_i}\right) = \ell(\gamma_{i\alpha_i})$ where α_i has the highest score i.e. $\gamma_{i\alpha_i} > \frac{1}{|\mathcal{T}_i|} \sum_{\theta \in \mathcal{T}_i} \gamma_{i\theta}$. Set $\tilde{\boldsymbol{p}}(R) = \arg\min_{\boldsymbol{p} \in \mathcal{N}_\delta, \|\boldsymbol{p}\| \leq R} \mathcal{L}(\boldsymbol{p})$. Consequently, there are two scenarios are:

- $\lim_{R \to \infty} \|\tilde{\boldsymbol{p}}(R)\|$ is finite. This already proves the statement of the theorem as $\tilde{\boldsymbol{p}}(R)/R \to 0$ within $\delta < \epsilon$ neighborhood of $\boldsymbol{q}$.
- For sufficiently large R, the selected-tokens of $\tilde{\boldsymbol{p}}(R)$ are $\boldsymbol{\alpha} = (\alpha_i)_{i=1}^n$.

Proceeding with the second (remaining scenario), we study the directional margin of $\tilde{\boldsymbol{p}}(R)$. More broadly, for any $\boldsymbol{p} \in \mathcal{N}_\delta$ and $\bar{\boldsymbol{p}} = \boldsymbol{p}/\|\boldsymbol{p}\|$ with selected-tokens $\boldsymbol{\alpha}$, using the fact that $\|\bar{\boldsymbol{p}} - \boldsymbol{q}\| \leq \delta \ll \epsilon_0$, we can bound the directional margin as

- For $\theta \in \mathcal{T}_i, \theta \neq \alpha_i$: $(\boldsymbol{k}_{i\alpha_i} - \boldsymbol{k}_{i\theta})^\top \bar{\boldsymbol{p}} = (\boldsymbol{k}_{i\alpha_i} - \boldsymbol{k}_{i\theta})^\top (\bar{\boldsymbol{p}} - \boldsymbol{q}) \leq K\delta$.
- For all other (i, t) with $t \neq \alpha_i$: $(\boldsymbol{k}_{i\alpha_i} - \boldsymbol{k}_{it})^\top \bar{\boldsymbol{p}} \geq \Gamma_{\boldsymbol{q}} - K\delta$.

This means that, any such $\bar{\boldsymbol{p}} \in \mathcal{N}_\delta$ achieves a directional margin of at most

$$\Gamma_{\bar{\boldsymbol{p}}} \leq K\delta.$$

Applying Lemma A.8 and setting $\delta = \mathcal{O}(\epsilon_0)$, this implies that for

$$R \gtrsim \frac{1}{\Gamma_{\boldsymbol{q}_0} - \Gamma_{\bar{\boldsymbol{p}}}} = \frac{1}{\frac{\epsilon_0}{1 + \epsilon_0 \|\boldsymbol{q}^\perp\|} - K\delta} = \mathcal{O}(\frac{1}{\epsilon_0}),$$

we have that $\mathcal{L}(R\cdot \boldsymbol{q}_0) < \min_{\|\boldsymbol{p}\|=R, \boldsymbol{p}\in\mathcal{N}_\delta} \mathcal{L}(\boldsymbol{p})$. Since this holds for all R, (A.96) holds (similar to Case **(A1)**) and we conclude that whenever $\|\bar{\boldsymbol{p}}(R)\| \to \infty$, it doesn't directionally converge within $\mathcal{N}_\delta$ (i.e. $\delta > 0$ neighborhood of $\boldsymbol{q}$) proving the advertised result. ■

A.2.7 Proof of Lemma 2.2

We prove a slightly general restatement where we require $\boldsymbol{v} \in \text{range}(\boldsymbol{W}^\top)$ – instead of full-rank $\boldsymbol{W}$.

Lemma A.11 *Suppose for all $i \in [n]$ and $t \neq opt_i$, $y_i = 1$ and $\gamma_{it} < \gamma_{iopt_i}$. Also suppose $\boldsymbol{v} \in \text{range}(\boldsymbol{W}^\top)$. Then, $\boldsymbol{p}^{mm\star}$ exists – i.e. ($\boldsymbol{p}$-SVM) is feasible for optimal indices $\alpha_i \leftarrow opt_i$.*

Proof. To establish the existence of $\boldsymbol{p}^{\text{mm}\star}$, we only need to find a direction that demonstrates the feasibility of ($\boldsymbol{p}$-SVM), i.e. we need to find $\boldsymbol{p}$ that satisfies the margin constraints. To begin, let's define the minimum score difference:

$$\underline{\gamma} = \min_{i\in[n], t\neq \text{opt}_i} \gamma_{i\text{opt}_i} - \gamma_{it}.$$

We then set $\boldsymbol{p} = \underline{\gamma}^{-1}(\boldsymbol{W}^\top)^\dagger \boldsymbol{v}$ where † denotes pseudo-inverse. By assumption $\boldsymbol{W}^\top \boldsymbol{p} = \underline{\gamma}^{-1}\boldsymbol{v}$. To conclude, observe that $\boldsymbol{p}$ is a feasible solution since $\boldsymbol{k}_{it} = \boldsymbol{W}\boldsymbol{x}_{it}$ and for all $i \in [n]$ and $t \neq \text{opt}_i$, we have that

$$\begin{aligned}(\boldsymbol{k}_{i\text{opt}_i} - \boldsymbol{k}_{it})^\top \boldsymbol{p} = (\boldsymbol{x}_{i\text{opt}_i} - \boldsymbol{x}_{it})^\top \boldsymbol{W}^\top \boldsymbol{p} &= \underline{\gamma}^{-1}(\boldsymbol{x}_{i\text{opt}_i} - \boldsymbol{x}_{it})^\top \boldsymbol{W}^\top (\boldsymbol{W}^\top)^\dagger \boldsymbol{v} \\ &= \underline{\gamma}^{-1}(\boldsymbol{x}_{i\text{opt}_i} - \boldsymbol{x}_{it})^\top \boldsymbol{v} \geq 1,\end{aligned}$$

which together with the constraints in ($\boldsymbol{p}$-SVM) completes the proof. ■

A.3 Proofs for Section 2.4

A.3.1 Proof of Theorem 2.5

Proof. Suppose the claim is incorrect and either $\boldsymbol{p}_R/R$ or $\boldsymbol{v}_r/r$ fails to converge as R, r grows. Set $\Xi = 1/\|\boldsymbol{p}^{\text{mm}}\|$, $\Gamma = 1/\|\boldsymbol{v}^{\text{mm}}\|$, $\tilde{\boldsymbol{p}}^{\text{mm}} = R\Xi\boldsymbol{p}^{\text{mm}}$ and $\tilde{\boldsymbol{v}}^{\text{mm}} = r\Gamma\boldsymbol{v}^{\text{mm}}$. The proof strategy is obtaining a contradiction by proving that $(\tilde{\boldsymbol{v}}^{\text{mm}}, \tilde{\boldsymbol{p}}^{\text{mm}})$ is a strictly better solution compared to $(\boldsymbol{v}_r, \boldsymbol{p}_R)$ for large R, r. Without losing generality, we will set $\alpha_i = 1$ for all $i \in [n]$ as the problem is invariant to tokens' permutation. Define $q_i^{\boldsymbol{p}} = 1 - \boldsymbol{s}_{i1}^{\boldsymbol{p}}$ to be the

amount of non-optimality (cumulative probability of non-first tokens) where $\boldsymbol{s}_i^{\boldsymbol{p}} = \mathbb{S}(\boldsymbol{K}_i\boldsymbol{p})$ is the softmax probabilities.

• **Case 1: $\boldsymbol{p}_R/R$ does not converge.** Under this scenario there exists $\delta, \gamma = \gamma(\delta) > 0$ such that we can find arbitrarily large R with $\|\boldsymbol{p}_R/R - \tilde{\boldsymbol{p}}^{\mathrm{mm}}/R\| \geq \delta$ and margin induced by $\boldsymbol{p}_R/R$ is at most $\Xi(1-\gamma)$ (from strong convexity of ($\boldsymbol{p}$-SVM)). Following $q_i^{\boldsymbol{p}}$ definition above, set $\hat{q}_{\max} = \sup_{i\in[n]} q_i^{\boldsymbol{p}_R}$ to be worst non-optimality in $\boldsymbol{p}_R$ and $q^\star_{\max} = \sup_{i\in[n]} q_i^{\tilde{\boldsymbol{p}}^{\mathrm{mm}}}$ to be the same for $\tilde{\boldsymbol{p}}^{\mathrm{mm}}$. Repeating the identical argument in Theorem 2.7 (specifically (A.111)), we can bound the non-optimality amount $q_i^{\tilde{\boldsymbol{p}}^{\mathrm{mm}}}$ of $\tilde{\boldsymbol{p}}^{\mathrm{mm}}$ as

$$q_i^{\tilde{\boldsymbol{p}}^{\mathrm{mm}}} = \frac{\sum_{t\neq\alpha_i}\exp(\boldsymbol{k}_{it}^\top\tilde{\boldsymbol{p}}^{\mathrm{mm}})}{\sum_{t\in[T]}\exp(\boldsymbol{k}_{it}^\top\tilde{\boldsymbol{p}}^{\mathrm{mm}})} \leq \frac{\sum_{t\neq\alpha_i}\exp(\boldsymbol{k}_{it}^\top\tilde{\boldsymbol{p}}^{\mathrm{mm}})}{\exp(\boldsymbol{k}_{i\alpha_i}^\top\tilde{\boldsymbol{p}}^{\mathrm{mm}})} \leq T\exp(-R\Xi). \tag{A.99}$$

Thus, $q^\star_{\max} = \max_{i\in[n]} q_i^{\tilde{\boldsymbol{p}}^{\mathrm{mm}}} \leq T\exp(-R\Xi)$. Next without losing generality, assume first margin constraint is γ-violated by $\boldsymbol{p}_R$ and $\min_{t\neq\alpha_1}(\boldsymbol{k}_{1\alpha_1} - \boldsymbol{k}_{1t})^\top\boldsymbol{p}_R \leq \Xi R(1-\gamma)$. Denoting the amount of non-optimality of the first input as $q_1^{\boldsymbol{p}_R}$, we find

$$q_1^{\boldsymbol{p}_R} = \frac{\sum_{t\neq\alpha_1}\exp(\boldsymbol{k}_{1t}^\top\boldsymbol{p}_R)}{\sum_{t\in[T]}\exp(\boldsymbol{k}_{1t}^\top\boldsymbol{p}_R)} \geq \frac{1}{T}\frac{\sum_{t\neq\alpha_1}\exp(\boldsymbol{k}_{1t}^\top\boldsymbol{p}_R)}{\exp(\boldsymbol{k}_{1\alpha_1}^\top\boldsymbol{p}_R)} \geq T^{-1}\exp(-(1-\gamma)R\Xi). \tag{A.100}$$

We similarly have $q^\star_{\max} \geq T^{-1}\exp(-R\Xi)$ to find that

$$\begin{aligned} \log(\hat{q}_{\max}) &\geq -(1-\gamma)\Xi R - \log T, \\ -\Xi R - \log T \leq \log(q^\star_{\max}) &\leq -\Xi R + \log T. \end{aligned} \tag{A.101}$$

In words, $\tilde{\boldsymbol{p}}^{\mathrm{mm}}$ contains exponentially less non-optimality compared to $\boldsymbol{p}_R$ as R grows. The remainder of the proof differs from Theorem 2.7 as we need to upper/lower bound the logistic loss of $(\tilde{\boldsymbol{v}}^{\mathrm{mm}}, \tilde{\boldsymbol{p}}^{\mathrm{mm}})$ and $(\boldsymbol{v}_r, \boldsymbol{p}_R)$ respectively to conclude with the contradiction.

First, let us upper bound the logistic loss of $(\tilde{\boldsymbol{v}}^{\mathrm{mm}}, \tilde{\boldsymbol{p}}^{\mathrm{mm}})$. Set $\boldsymbol{r}_i = \boldsymbol{X}_i^\top\mathbb{S}(\boldsymbol{K}_i\tilde{\boldsymbol{p}}^{\mathrm{mm}})$. Observe that if $\|\boldsymbol{r}_i - \boldsymbol{x}_{i1}\| \leq \epsilon_i$, we have that $\boldsymbol{v}^{\mathrm{mm}}$ satisfies the SVM constraints on $\boldsymbol{r}_i$ with $y_i\cdot\boldsymbol{r}_i^\top\boldsymbol{v}^{\mathrm{mm}} \geq 1 - \epsilon_i/\Gamma$. Consequently, setting $\epsilon_{\max} = \sup_{i\in[n]}\epsilon_i$, $\boldsymbol{v}^{\mathrm{mm}}$ achieves a label-margin of $\Gamma - \epsilon_{\max}$ on the dataset $(y_i, \boldsymbol{r}_i)_{i\in[n]}$. With this, we upper bound the logistic loss of $(\tilde{\boldsymbol{v}}^{\mathrm{mm}}, \tilde{\boldsymbol{p}}^{\mathrm{mm}})$ as follows. Let $M = \sup_{i\in[n],t,\tau\in[T]}\|\boldsymbol{x}_{it} - \boldsymbol{x}_{i\tau}\|$. Let us recall the fact (A.101) that worst-case perturbation is

$$\epsilon_{\max} \leq M\exp(-\Xi R + \log T) = MT\exp(-\Xi R).$$

This implies that

$$\begin{aligned}\mathcal{L}(\tilde{\boldsymbol{v}}^{\mathrm{mm}}, \tilde{\boldsymbol{p}}^{\mathrm{mm}}) &\le \max_{i\in[n]} \log(1+\exp(-y_i \boldsymbol{r}_i^\top \tilde{\boldsymbol{v}}^{\mathrm{mm}})). \\ &\le \max_{i\in[n]} \exp(-y_i \boldsymbol{r}_i^\top \tilde{\boldsymbol{v}}^{\mathrm{mm}}) \\ &\le \exp(-r\Gamma + r\epsilon_{\max}) \\ &\le e^{rMT\exp(-\Xi R)} e^{-r\Gamma}. \end{aligned} \tag{A.102}$$

Conversely, we obtain a lower bound for $(\boldsymbol{v}_r, \boldsymbol{p}_R)$. Set $\boldsymbol{r}_i = \boldsymbol{X}_i^\top \mathbb{S}(\boldsymbol{K}_i \boldsymbol{p}_R)$. Using Assumption 2.3, we find that solving ($\boldsymbol{v}$-SVM) on $(y_i, \boldsymbol{r}_i)_{i\in[n]}$ achieves at most $\Gamma - \nu e^{-(1-\gamma)\Xi R}/T$ margin. Consequently, we have

$$\begin{aligned}\mathcal{L}(\boldsymbol{v}_r, \boldsymbol{p}_R) &\ge \frac{1}{n} \max_{i\in[n]} \log(1+\exp(-y_i \boldsymbol{r}_i^\top \boldsymbol{v}_r)) \\ &\ge \frac{1}{2n} \max_{i\in[n]} \exp(-y_i \boldsymbol{r}_i^\top \boldsymbol{v}_r) \wedge \log 2 \\ &\ge \frac{1}{2n} \exp(-r(\Gamma - \nu e^{-(1-\gamma)\Xi R}/T)) \wedge \log 2 \\ &\ge \frac{1}{2n} e^{r(\nu/T)\exp(-(1-\gamma)\Xi R)} e^{-r\Gamma} \wedge \log 2. \end{aligned} \tag{A.103}$$

Observe that this lower bound dominates the upper bound from (A.102) when R is large, specifically when:

$$\begin{aligned} \frac{1}{2n} \exp\left(r(\nu/T)\exp\left(-(1-\gamma)\Xi R\right)\right) &\ge \exp\left(rMT\exp(-\Xi R)\right), \\ -\log(2n) + r(\nu/T)\exp\left(-(1-\gamma)\Xi R\right) &\ge rMT\exp(-\Xi R), \\ r\exp\left(-(1-\gamma)\Xi R\right)\left(\frac{\nu}{T} - MT\exp(-\gamma\Xi R)\right) &\ge \log(2n). \end{aligned}$$

As R gets larger,

$$r \gtrsim \frac{T\log(2n)}{\nu} \exp((1-\gamma)\Xi R) = \exp(\mathcal{O}(R)).$$

Thus, we indeed obtain the desired contradiction since such large R is guaranteed to exist when $\boldsymbol{p}_R/R \not\to \boldsymbol{p}^{\mathrm{mm}}$.

• **Case 2: $\boldsymbol{v}_r/r$ does not converge.** This is the simpler scenario: There exists $\delta > 0$ such that we can find arbitrarily large r obeying $\|\boldsymbol{v}_r/r - \boldsymbol{v}^{\mathrm{mm}}/\|\boldsymbol{v}^{\mathrm{mm}}\|\| \ge \delta$. If $\|\boldsymbol{p}_R/R - \Xi\boldsymbol{p}^{\mathrm{mm}}\| \not\to 0$, then "Case 1" applies. Otherwise, we have $\|\boldsymbol{p}_R/R - \Xi\boldsymbol{p}^{\mathrm{mm}}\| \to 0$, thus we can assume $\|\boldsymbol{p}_R/R - \Xi\boldsymbol{p}^{\mathrm{mm}}\| \le \epsilon$ for arbitrary choice of $\epsilon > 0$.

On the other hand, due to the strong convexity of ($\boldsymbol{v}$-SVM), for some $\gamma := \gamma(\delta) > 0$,

$\boldsymbol{v}_r$ achieves a margin of at most $(1-\gamma)\Gamma r$ on the dataset $(y_i, \boldsymbol{x}_{i1})_{i\in[n]}$. Additionally, since $\|\boldsymbol{p}_R/R - \Xi \boldsymbol{p}^{\mathrm{mm}}\| \le \epsilon$, $\boldsymbol{p}_R$ strictly separates all optimal tokens (for small enough $\epsilon > 0$) and $\hat{q}_{\max} := f(\epsilon) \to 0$ as $R \to \infty$. Consequently, setting $\boldsymbol{r}_i = \boldsymbol{X}_i^\top \mathbb{S}(\boldsymbol{K}_i \boldsymbol{p}_R)$, for sufficiently large $R > 0$ setting $M = \sup_{i\in[n],t\in[T]} \|\boldsymbol{x}_{it}\|$, we have that

$$\begin{aligned}\min_{i\in[n]} y_i \boldsymbol{v}_r^\top \boldsymbol{r}_i &\le \min_{i\in[n]} y_i \boldsymbol{v}_r^\top \boldsymbol{x}_{i1} + \sup_{i\in[n]} |\boldsymbol{v}_r^\top (\boldsymbol{r}_i - \boldsymbol{x}_{i1})| \\ &\le (1-\gamma)\Gamma r + M f(\epsilon) r \\ &\le (1-\gamma/2)\Gamma r. \end{aligned} \tag{A.104}$$

This in turn implies that logistic loss is lower bounded by (following (A.103)),

$$\mathcal{L}(\boldsymbol{v}_r, \boldsymbol{p}_R) \ge \frac{1}{2n} e^{\gamma\Gamma r/2} e^{-\Gamma r} \wedge \log 2.$$

Going back to (A.102), this exponentially dominates the upper bound of $(\tilde{\boldsymbol{p}}^{\mathrm{mm}}, \tilde{\boldsymbol{v}}^{\mathrm{mm}})$ whenever $rMT\exp(-\Xi R) < r\gamma\Gamma/2$, (that is, whenever R, r are sufficiently large), again concluding the proof. ∎

A.3.2 Proof of Theorem 2.6

We will prove this result in two steps. Our first claim restricts the optimization to the particular quadrant induced by $\min_{t\neq\alpha_i}(\boldsymbol{k}_{i\alpha_i} - \boldsymbol{k}_{it})^\top \boldsymbol{p}_R \ge 0$ under the theorem's condition $\mathbb{S}(\boldsymbol{K}_i \boldsymbol{p}_R)_{\alpha_i} \to 1$.

Lemma A.12 *Suppose $\mathbb{S}(\boldsymbol{K}_i \boldsymbol{p}_R)_{\alpha_i} \to 1$. Then, there exists R_0 such that for all $R \ge R_0$, we have that,*

$$\min_{t\neq\alpha_i} (\boldsymbol{k}_{i\alpha_i} - \boldsymbol{k}_{it})^\top \boldsymbol{p}_R \ge 0, \quad \text{for all} \quad i \in [n]. \tag{A.105}$$

Proof. Suppose the claim does not hold. Set $\boldsymbol{s}_i^R = \mathbb{S}(\boldsymbol{K}_i \boldsymbol{p}_R)$. Fix R_0 such that $\boldsymbol{s}_{i\alpha_i}^R \ge 0.9$ for all $R \ge R_0$. On the other hand, there exists arbitrarily large R for which $(\boldsymbol{k}_{i\alpha_i} - \boldsymbol{k}_{it})^\top \boldsymbol{p}_R < 0$ for some $t \neq \alpha_i \in [T]$ and $i \in [n]$. At this (R, i, t) choices, we have that $\boldsymbol{s}_{it}^R \ge \boldsymbol{s}_{i\alpha_i}^R$. Since $\boldsymbol{s}_{it}^R + \boldsymbol{s}_{i\alpha_i}^R \le 1$, we find $\boldsymbol{s}_{i\alpha_i}^R < 0.5$ which contradicts with $\boldsymbol{s}_{i\alpha_i}^R \ge 0.9$. ∎

Let $\mathcal{Q}$ be the set of $\boldsymbol{p}$ satisfying the quadrant constraint (A.105) – i.e. indices $(\alpha_i)_{i=1}^n$ are selected. Let $\boldsymbol{h}_R$ be the solution of regularization path of $(\boldsymbol{v}, \boldsymbol{p})$ subject to the constraint $\boldsymbol{p} \in \mathcal{Q}$. From Lemma A.12, we know that, for some R_0 and all $R \ge R_0$, $\boldsymbol{h}_R = \boldsymbol{p}_R$. Thus, if the limit exists, we have that $\lim_{R\to\infty} \boldsymbol{h}_R/R = \lim_{R\to\infty} \boldsymbol{p}_R/R$.

To proceed, we will prove that $\lim_{R\to\infty} \boldsymbol{h}_R/R$ exists and is equal to $\boldsymbol{p}^{\text{relax}}/\|\boldsymbol{p}^{\text{relax}}\|$ and simultaneously establish $\boldsymbol{v}_r/r \to \boldsymbol{v}^{\text{mm}}/\|\boldsymbol{v}^{\text{mm}}\|$.

Lemma A.13 $\lim_{R\to\infty} \boldsymbol{h}_R/R = \boldsymbol{p}^{relax}/\|\boldsymbol{p}^{relax}\|$ *and* $\lim_{r\to\infty} \boldsymbol{v}_r/r = \boldsymbol{v}^{mm}/\|\boldsymbol{v}^{mm}\|$.

Proof. The proof will be similar to that of Theorem 2.5. As usual, we aim to show that SVM-solutions constitute the most competitive direction. Set $\Xi = 1/\|\boldsymbol{p}^{\text{relax}}\|$.

• **Case 1: $\boldsymbol{h}_R/R$ does not converge.** Under this scenario there exists $\delta, \gamma = \gamma(\delta) > 0$ such that we can find arbitrarily large R with $\|\boldsymbol{h}_R/R - \Xi\boldsymbol{p}^{\text{relax}}\| \geq \delta$. This implies that margin induced by $\boldsymbol{h}_R/R$ is at most $\Xi(1-\gamma)$ over the support vectors $\mathcal{S}$ (from strong convexity of (2.11)). The reason is that, $\boldsymbol{h}_R$ satisfies $\boldsymbol{h}_R^\top(\boldsymbol{k}_{i\alpha_i} - \boldsymbol{k}_{it}) \geq 0$ for all $t \neq \alpha_i$ by construction as $\boldsymbol{h}_R \in \mathcal{Q}$. Thus, a constraint over the support vectors have to be violated (when normalized to the same ℓ_2 norm as $\|\boldsymbol{p}^{\text{relax}}\| = 1/\Xi$).

As usual, we will construct a solution strictly superior to $\boldsymbol{h}_R$ and contradicts with its optimality.

Construction of competitor: Rather than using $\boldsymbol{p}^{\text{relax}}$ direction, we will choose a slightly deviating direction that ensures the selection of the correct tokens over non-supports $\bar{\mathcal{S}}$. Specifically, consider the solution of (2.11) where we tighten the non-support constraints by arbitrarily small $\epsilon > 0$.

$$\boldsymbol{p}^{\epsilon\text{-rlx}} = \arg\min_{\boldsymbol{p}} \|\boldsymbol{p}\| \quad \text{such that} \quad \boldsymbol{p}^\top(\boldsymbol{k}_{i\alpha_i} - \boldsymbol{k}_{it}) \geq \begin{cases} 1 & \text{for all} \quad t \neq \alpha_i,\ i \in \mathcal{S} \\ \epsilon & \text{for all} \quad t \neq \alpha_i,\ i \in \bar{\mathcal{S}} \end{cases} . \quad \text{(A.106)}$$

Let $\boldsymbol{p}^{\text{mm}}$ be the solution of ($\boldsymbol{p}$-SVM) with $\boldsymbol{\alpha} = (\alpha_i)_{i=1}^n$ (which was assumed to be separable). Observe that $\boldsymbol{p}_\epsilon^{\text{mm}} = \epsilon\boldsymbol{p}^{\text{mm}} + (1-\epsilon)\boldsymbol{p}^{\text{relax}}$ satisfies the constraints of (A.106). Additionally, $\boldsymbol{p}_\epsilon^{\text{mm}}$ would achieve a margin of $\frac{1}{(1-\epsilon)/\Xi + \epsilon/\Delta} = \frac{\Delta\Xi}{\Delta + \epsilon(\Xi - \Delta)}$ where $\Delta = 1/\|\boldsymbol{p}^{\text{mm}}\|$. Using optimality of $\boldsymbol{p}^{\epsilon\text{-rlx}}$, this implies that the reduced margin $\Xi_\epsilon = 1/\|\boldsymbol{p}^{\epsilon\text{-rlx}}\|$ (by enforcing ϵ over non-support) over the support vectors is a Lipschitz function of ϵ. That is $\Xi_\epsilon \geq \Xi - \epsilon M$ for some $M \geq 0$. To proceed, choose an $\epsilon > 0$ such that, it is strictly superior to margin induced by $\boldsymbol{h}_R$, that is,

$$\Xi_\epsilon \geq \Xi\left(1 - \frac{\gamma}{2}\right).$$

To proceed, set $\tilde{\boldsymbol{p}}^{\epsilon\text{-rlx}} = R\Xi_\epsilon\boldsymbol{p}^{\epsilon\text{-rlx}}$. Let us recall the following notation from the proof of Theorem 2.5: $\boldsymbol{s}_i^{\boldsymbol{p}} = \mathbb{S}(\boldsymbol{K}_i\boldsymbol{p})$ and $q_i^{\boldsymbol{p}} = 1 - \boldsymbol{s}_{i\alpha_i}$. Set $\hat{q}_{\max} = \max_{i\in\mathcal{S}} q_i^{\boldsymbol{h}_R}$ to be worst non-optimality of $\boldsymbol{h}_R$ over **support set**. Similarly, define $q^\star_{\max} = \max_{i\in\mathcal{S}} q_i^{\tilde{\boldsymbol{p}}^{\epsilon\text{-rlx}}}$ to be the same for $\tilde{\boldsymbol{p}}^{\epsilon\text{-rlx}}$. Repeating the identical arguments to (A.99), (A.100), (A.101), and using the fact

that $\boldsymbol{p}^{\epsilon\text{-rlx}}$ achieves a margin $\Xi(1-\frac{\gamma}{2}) \leq \Xi_\epsilon \leq \Xi$, we end up with the lines

$$\log(\hat{q}_{\max}) \geq -(1-\gamma)\Xi R - \log T, \tag{A.107a}$$

$$-\Xi R - \log T \leq \log(q^\star_{\max}) \leq -\Xi(1-0.5\gamma)R + \log T. \tag{A.107b}$$

In what follows, we will prove that $\tilde{\boldsymbol{p}}^{\epsilon\text{-rlx}}$ achieves a strictly smaller logistic loss contradicting with the optimality of $\boldsymbol{p}_R$ (whenever $\|\boldsymbol{h}_R/R - \Xi \boldsymbol{p}^{\text{relax}}\| \geq \delta$).

Upper bounding logistic loss. Let us now upper bound the logistic loss of $(\tilde{\boldsymbol{v}}^{\text{mm}}, \tilde{\boldsymbol{p}}^{\epsilon\text{-rlx}})$ where $\tilde{\boldsymbol{v}}^{\text{mm}} = r\Gamma \boldsymbol{v}^{\text{mm}}$ with $\boldsymbol{v}^{\text{mm}}$ being the solution of ($\boldsymbol{v}$-SVM) with $\boldsymbol{r}_i \leftarrow \boldsymbol{x}_{i\alpha_i}$ and $\Gamma = 1/\|\boldsymbol{v}^{\text{mm}}\|$. Set $\boldsymbol{r}_i = \boldsymbol{X}_i^\top \mathbb{S}(\boldsymbol{K}_i \tilde{\boldsymbol{p}}^{\epsilon\text{-rlx}})$. Set $\upsilon = \min_{i\in\bar{\mathcal{S}}} y_i \cdot \boldsymbol{x}_{i\alpha_i}^\top \boldsymbol{v}^{\text{mm}} - 1$ to be the additional margin buffer that non-support vectors have access to. Also set $M = \sup_{i\in[n],t,\tau\in[T]} \|\boldsymbol{x}_{it} - \boldsymbol{x}_{i\tau}\|$. Observe that we can write

$$\boldsymbol{x}_{i\alpha_i} - \boldsymbol{r}_i = \sum_{t\neq\alpha_i} \boldsymbol{s}_{it}(\boldsymbol{x}_{i\alpha_i} - \boldsymbol{x}_{it}) \implies \|\boldsymbol{x}_{i\alpha_i} - \boldsymbol{r}_i\| \leq q_i M.$$

Non-supports achieve strong label-margin: Using above and (A.106) for all $i \in \bar{\mathcal{S}}$ and $t \neq \alpha_i$, we have that $\boldsymbol{s}_{it} \leq e^{-\epsilon\Xi_\epsilon R}\boldsymbol{s}_{i\alpha_i} \leq e^{-\epsilon\Xi(1-\gamma/2)R}\boldsymbol{s}_{i\alpha_i}$. Consequently, whenever $R \geq \bar{R}_0 := (\epsilon\Xi(1-\gamma/2))^{-1}\log(\frac{TM}{\Gamma\upsilon})$,

$$q_i^{\tilde{\boldsymbol{p}}^{\epsilon\text{-rlx}}} \leq \frac{\sum_{t\neq\alpha_i}\boldsymbol{s}_{it}}{\boldsymbol{s}_{i\alpha_i}} \leq Te^{-\epsilon\Xi(1-\gamma/2)R} \leq \frac{\Gamma\upsilon}{M}.$$

This implies that, on $i \in \bar{\mathcal{S}}$

$$y_i \cdot \boldsymbol{r}_i^\top \boldsymbol{v}^{\text{mm}} \geq 1 + \upsilon + y_i \cdot (\boldsymbol{r}_i - \boldsymbol{x}_{i\alpha_i})^\top \boldsymbol{v}^{\text{mm}} \geq 1 + \upsilon - q_i M \|\boldsymbol{v}^{\text{mm}}\| \geq 1. \tag{A.108}$$

In words: Above a fixed $\bar{R}_0$ that only depends on $\gamma = \gamma(\delta)$, features $\boldsymbol{r}_i$ induced by all non-support indices $i \in \bar{\mathcal{S}}$ achieve margin at least 1. What remains is analyzing the margin shrinkage over the support vectors as in Theorem 2.5.

Controlling support margin and combining bounds: Over $\mathcal{S}$, suppose $\boldsymbol{v}^{\text{mm}}$ satisfies the SVM constraints on $\boldsymbol{r}_i$ with $y_i \cdot \boldsymbol{r}_i^\top \boldsymbol{v}^{\text{mm}} \geq 1 - \epsilon_i/\Gamma$. Consequently, setting $\epsilon_{\max} = \sup_{i\in[n]} \epsilon_i$, $\boldsymbol{v}^{\text{mm}}$ achieves a label-margin of $\Gamma - \epsilon_{\max}$ on the dataset $(y_i, \boldsymbol{r}_i)_{i\in[n]}$. Next, we recall the fact (A.107b) that worst-case perturbation is $\epsilon_{\max} \leq M\exp(-\Xi(1-0.5\gamma)R + \log T) = MT\exp(-\Xi(1-0.5\gamma)R)$. With this and (A.108), we upper bound the logistic loss of

$(\tilde{\boldsymbol{v}}^{\mathrm{mm}}, \tilde{\boldsymbol{p}}^{\epsilon\text{-rlx}})$ as follows.

$$\begin{aligned}\mathcal{L}(\tilde{\boldsymbol{v}}^{\mathrm{mm}}, \tilde{\boldsymbol{p}}^{\mathrm{mm}}) &\leq \max_{i\in[n]} \log(1+\exp(-y_i \boldsymbol{r}_i^\top \tilde{\boldsymbol{v}}^{\mathrm{mm}})). \\ &\leq \max_{i\in[n]} \exp(-y_i \boldsymbol{r}_i^\top \tilde{\boldsymbol{v}}^{\mathrm{mm}}) \\ &\leq \exp(-r\Gamma + r\epsilon_{\max}) \\ &\leq e^{rMT\exp(-\Xi(1-0.5\gamma)R)} e^{-r\Gamma}. \end{aligned} \tag{A.109}$$

Conversely, we obtain a lower bound for $(\boldsymbol{v}_r, \boldsymbol{h}_R)$. Set $\boldsymbol{r}_i = \boldsymbol{X}_i^\top \mathbb{S}(\boldsymbol{K}_i \boldsymbol{h}_R)$. Recall the lower bound (A.107a) over the support vector set $\mathcal{S}$. Combining this with our Assumption 2.3 over the support vectors of ($\boldsymbol{v}$-SVM) implies that, solving ($\boldsymbol{v}$-SVM) on $(y_i, r_i)_{i\in[n]}$ achieves at most $\Gamma - \nu e^{-(1-\gamma)\Xi R}/T$ margin. Consequently, we have

$$\begin{aligned}\mathcal{L}(\boldsymbol{v}_r, \boldsymbol{h}_R) &\geq \frac{1}{n} \max_{i\in[n]} \log(1+\exp(-y_i \boldsymbol{r}_i^\top \boldsymbol{v}_r)) \\ &\geq \frac{1}{2n} \max_{i\in[n]} \exp(-y_i \boldsymbol{r}_i^\top \boldsymbol{v}_r) \wedge \log 2 \\ &\geq \frac{1}{2n} \exp(-r(\Gamma - \nu e^{-(1-\gamma)\Xi R}/T)) \wedge \log 2 \\ &\geq \frac{1}{2n} e^{r(\nu/T)\exp(-(1-\gamma)\Xi R)} e^{-r\Gamma} \wedge \log 2. \end{aligned} \tag{A.110}$$

Observe that, this lower bound dominates the previous upper bound when R is large, namely, when (ignoring the multiplier $1/2n$ for brevity)

$$(\nu/T) e^{-(1-\gamma)\Xi R} \geq MT e^{-\Xi(1-0.5\gamma)R} \iff R \geq R_0 := \frac{2}{\gamma\Xi} \log\left(\frac{MT^2}{\nu}\right).$$

Thus, we obtain the desired contradiction since $\tilde{\boldsymbol{p}}^{\epsilon\text{-rlx}}$ is a strictly better solution compared to $\boldsymbol{p}_R = \boldsymbol{h}_R$ (once R is sufficiently large).

• **Case 2: $\boldsymbol{v}_r/r$ does not converge.** This is the simpler scenario: There exists $\delta > 0$ such that we can find arbitrarily large r obeying $\|\boldsymbol{v}_r/r - \boldsymbol{v}^{\mathrm{mm}}/\|\boldsymbol{v}^{\mathrm{mm}}\|\| \geq \delta$. First, note that, due to the strong convexity of ($\boldsymbol{v}$-SVM), for some $\gamma := \gamma(\delta) > 0$, $\boldsymbol{v}_r$ achieves a margin of at most $(\Gamma - \gamma)r$ on the dataset $(y_i, \boldsymbol{x}_{i1})_{i\in[n]}$. By theorem's condition, we are provided that $\mathbb{S}(\boldsymbol{K}_i \boldsymbol{p}_R)_{\alpha_i} \to 1$. This immediately implies that, for any choice of $\epsilon = \gamma/3 > 0$, above some sufficiently large (r_0, R_0), we have that $\|\boldsymbol{x}_i^{\boldsymbol{p}_R} - \boldsymbol{r}_i\| \leq \epsilon$. Following (A.109), this implies that, choosing $\tilde{\boldsymbol{v}}^{\mathrm{mm}} = r\boldsymbol{v}^{\mathrm{mm}}/\|\boldsymbol{v}^{\mathrm{mm}}\|$ achieves a logistic loss of at most $e^{r\gamma/3} e^{-r\Gamma}$. Again using

$\|\boldsymbol{x}_i^{\boldsymbol{p}_R} - \boldsymbol{r}_i\| \le \epsilon$, for sufficiently large (r, R) we have that

$$\begin{aligned}\min_{i\in[n]} y_i \boldsymbol{v}_r^\top \boldsymbol{r}_i &\le \min_{i\in[n]} y_i \boldsymbol{v}_r^\top \boldsymbol{x}_{i1} + \sup_{i\in[n]} |\boldsymbol{v}_r^\top (\boldsymbol{r}_i - \boldsymbol{x}_{i1})| \\ &\le (\Gamma - \gamma) r + \epsilon r \\ &\le (\Gamma - 2\gamma/3) r.\end{aligned}$$

This in turn implies that logistic loss is lower bounded by (following (A.110)),

$$\mathcal{L}(\boldsymbol{v}_r, \boldsymbol{p}_R) \ge \frac{1}{2n} e^{2\gamma r/3} e^{-r\Gamma} \wedge \log 2.$$

This dominates the above upper bound $e^{r\gamma/3}e^{-r\Gamma}$ of $\tilde{\boldsymbol{v}}^{\mathrm{mm}}$ whenever $\frac{1}{2n}e^{\gamma r/3} > 1 \iff r > \frac{3}{\gamma}\log(2n)$, (that is, when r is sufficiently large), again concluding the proof. ∎

A.4 Regularization Path of Attention with Nonlinear Head

A.4.1 Proof of Theorem 2.7

Proof. The key idea is showing that, thanks to the exponential tail of softmax-attention, (harmful) contribution of the non-optimal token with the minimum margin can dominate the contribution of all other tokens as $R \to \infty$. This high-level approach is similar to earlier works on implicit bias of gradient descent with logistic loss [89, 177].

Pick $\boldsymbol{p}^{\mathrm{mm}} \in \mathcal{G}^{\mathrm{mm}}$ and set $\boldsymbol{p}_R^\star = R\frac{\boldsymbol{p}^{\mathrm{mm}}}{\|\boldsymbol{p}^{\mathrm{mm}}\|}$. This will be the baseline model that $\boldsymbol{p}_R$ has to compete against. Also let $\bar{\boldsymbol{p}}_R = \frac{\boldsymbol{p}_R}{\Xi R}$. Now suppose $\texttt{dist}\,(\bar{\boldsymbol{p}}_R, \mathcal{G}^{\mathrm{mm}}) \not\to 0$ as $R \to \infty$. Then, there exists $\delta > 0$ such that, we can always find arbitrarily large R obeying $\texttt{dist}\,(\bar{\boldsymbol{p}}_R, \mathcal{G}^{\mathrm{mm}}) \ge \delta$.

Since $\bar{\boldsymbol{p}}_R$ is $\delta > 0$ bounded away from $\mathcal{G}^{\mathrm{mm}}$, and $\|\bar{\boldsymbol{p}}_R\| = \|\boldsymbol{p}^{\mathrm{mm}}\|$, $\bar{\boldsymbol{p}}_R$ strictly violates at least one of the inequality constraints in ($\boldsymbol{p}$-SVM'). Otherwise, we would have $\bar{\boldsymbol{p}}_R \in \mathcal{G}^{\mathrm{mm}}$. Without losing generality, suppose $\bar{\boldsymbol{p}}_R$ violates the first margin constraint, that is, for some $\gamma := \gamma(\delta) > 0$, $\max_{\alpha\in\mathcal{R}_1} \min_{\beta\in\bar{\mathcal{R}}_1} \bar{\boldsymbol{p}}_R^\top(\boldsymbol{k}_{1\alpha} - \boldsymbol{k}_{1\beta}) \le 1 - \gamma$. Now, we will argue that this will lead to a contradiction as $R \to \infty$ since we will show that $\mathcal{L}(\boldsymbol{p}_R^\star) < \mathcal{L}(\boldsymbol{p}_R)$ for sufficiently large R.

First, let us control $\mathcal{L}(\boldsymbol{p}_R^\star)$. We study $\boldsymbol{s}_i^\star = \mathbb{S}(\boldsymbol{K}_i \boldsymbol{p}_R^\star)$ and let $\alpha_i \in \mathcal{R}_i$ be the index α in ($\boldsymbol{p}$-SVM') for which $\mathrm{margin}_i = \max_{\alpha\in\mathcal{R}_i} \min_{\beta\in\bar{\mathcal{R}}_i} (\boldsymbol{k}_{i\alpha} - \boldsymbol{k}_{i\beta})^\top \boldsymbol{p}^{\mathrm{mm}} \ge 1$ is attained. Then, we

bound the non-optimality amount $q_i^\star$ of $\boldsymbol{p}_R^\star$ as

$$q_i^\star = \frac{\sum_{t\in\bar{\mathcal{R}}_i}\exp(\boldsymbol{k}_{it}^\top \boldsymbol{p}_R^\star)}{\sum_{t\in[T]}\exp(\boldsymbol{k}_{it}^\top \boldsymbol{p}_R^\star)} \leq \frac{\sum_{t\in\bar{\mathcal{R}}_i}\exp(\boldsymbol{k}_{it}^\top \boldsymbol{p}_R^\star)}{\exp(\boldsymbol{k}_{i\alpha_i}^\top \boldsymbol{p}_R^\star)} \leq T\exp(-\Xi R).$$

Thus, $q_{\max}^\star = \max_{i\in[n]} q_i^\star \leq T\exp(-\Xi R)$. Secondly, we wish to control $\mathcal{L}(\boldsymbol{p}_R)$ by lower bounding the non-optimality in $\boldsymbol{p}_R$. Focusing on the first margin constraint, let $\alpha \in \mathcal{R}_1$ be the index in ($\boldsymbol{p}$-SVM') for which $\text{margin}_1 \leq 1-\gamma$ is attained. Denoting the amount of non-optimality of the first input as $\hat{q}_1$, we find[6]

$$\hat{q}_1 = \frac{\sum_{t\in\bar{\mathcal{R}}_1}\exp(\boldsymbol{k}_{1t}^\top \boldsymbol{p}_R)}{\sum_{t\in[T]}\exp(\boldsymbol{k}_{1t}^\top \boldsymbol{p}_R)} \geq \frac{1}{T}\frac{\sum_{t\in\bar{\mathcal{R}}_1}\exp(\boldsymbol{k}_{1t}^\top \boldsymbol{p}_R)}{\exp(\boldsymbol{k}_{1\alpha}^\top \boldsymbol{p}_R)} \geq T^{-1}\exp(-\Xi R(1-\gamma)).$$

We similarly have $q_{\max}^\star \geq T^{-1}\exp(-\Xi R)$. In conclusion, for $\boldsymbol{p}_R, \boldsymbol{p}_R^\star$, denoting maximum non-optimality by $\hat{q}_{\max} \geq \hat{q}_1$ and $q_{\max}^\star$, we respectively obtained

$$\begin{aligned} \log(\hat{q}_{\max}) &\geq -(1-\gamma)(\Xi R) - \log T, \\ -(\Xi R) - \log T \leq \log(q_{\max}^\star) &\leq -(\Xi R) + \log T. \end{aligned} \tag{A.111}$$

The above inequalities satisfy Assumption 2.4 as follows where $\boldsymbol{p} \leftarrow \boldsymbol{p}_R^\star$ and $\boldsymbol{p}' \leftarrow \boldsymbol{p}_R$: Set $R_0 = 3\gamma^{-1}\Xi^{-1}\log T$ so that $\log T = \frac{\gamma\Xi R_0}{3}$. Secondly, set $\rho_0 = -\Xi R_0 - \log T$. This way, $\rho_0 \geq \log(q_{\max}^\star)$ implies $R \geq R_0$ and $\log T \leq \frac{\gamma\Xi R}{3}$. Using the latter inequality, we bound the $\log T$ terms to obtain

- $\log(\hat{q}_{\max}) \geq -(1-2\gamma/3)(\Xi R)$, and
- $\log(q_{\max}^\star) \leq -(1-\gamma/3)(\Xi R)$.

To proceed, we pick $1+\Delta = \frac{1-\gamma/3}{1-2\gamma/3}$ implying $\Delta := \frac{\gamma}{3-2\gamma}$. Finally, for this Δ, there exists $\rho(\Delta)$ which we need to ensure $\log(\hat{q}_{\max}) \leq \rho(\Delta)$. This can be guaranteed by picking sufficiently large R that ensures $\log(q_{\max}^\star) \leq -(1-\gamma/3)(\Xi R) \leq \rho(\Delta)$ to satisfy all conditions of Assumption 2.4. Since such large R exists by initial assumption $\texttt{dist}\,(\bar{\boldsymbol{p}}_R, \mathcal{G}^{\text{mm}}) \not\to 0$, Assumption 2.4 in turn implies that $\mathcal{L}(\boldsymbol{p}_R^\star) < \mathcal{L}(\boldsymbol{p}_R)$ contradicting with the optimality of $\boldsymbol{p}_R$ in (2.12). ∎

[6]Here, we assumed margin is non-negative i.e. $\boldsymbol{k}_{1\alpha}^\top \boldsymbol{p}_R \geq \sup_{t\in\bar{\mathcal{R}}_1} \boldsymbol{k}_{1t}^\top \boldsymbol{p}_R$. Otherwise, $\sup_{t\in[T]} \boldsymbol{k}_{1t}^\top \boldsymbol{p}_R$ is attained in $\bar{\mathcal{R}}_1$ which implies $\hat{q}_1 \geq T^{-1}$. Thus, we can still use the identical inequality (A.111) with the choice $\gamma = 1$.

A.4.2 Application to Linearly-mixed Labels

The following example shows that if non-optimal tokens result in reduced score (in terms of the alignment of prediction and label), Assumption 2.4 holds. The high-level idea behind this lemma is that, if the optimal risk is achieved by setting $q^{\boldsymbol{p}}_{\max} = 0$, then, Assumption 2.4 will hold.

Lemma A.14 (Linear label mixing) *Recall $q^{\boldsymbol{p}}_i = \sum_{t\in\bar{\mathcal{R}}_t} s^{\boldsymbol{p}}_{it}$ from Assumption 2.4. Suppose $y_i \in \{-1, 1\}$ and*

$$y_i \cdot \psi(\boldsymbol{X}_i^\top \boldsymbol{s}^{\boldsymbol{p}}_i) = \nu_i(1 - q^{\boldsymbol{p}}_i) + Z_i,$$

for some $(\nu_i)_{i=1}^n > 0$. Here $Z_i = Z_i(\boldsymbol{p})$ is the contribution of non-optimal tokens to prediction. For some $C, \epsilon > 0$ and for all $\boldsymbol{p} \in \mathbb{R}^d$, assume

$$-Cq^{\boldsymbol{p}}_{\max} \le Z_i \le (1-\epsilon)\nu_i q^{\boldsymbol{p}}_i. \tag{A.112}$$

Then, Assumption 2.4 holds for $\mathcal{L}(\boldsymbol{p}) = \frac{1}{n}\sum_{i=1}^n \ell(y_i \cdot \psi(\boldsymbol{X}_i^\top \boldsymbol{s}^{\boldsymbol{p}}_i))$ when $\ell(\cdot)$ is a strictly decreasing loss function with continuous derivative.

Proof. Recall the assumption $y_i \cdot \psi(\boldsymbol{X}_i^\top \boldsymbol{s}^{\boldsymbol{p}}_i) = \nu_i(1 - q^{\boldsymbol{p}}_i) + Z_i$ with Z_i obeying (A.112). Let us also write the loss function

$$\mathcal{L}(\boldsymbol{p}) = \frac{1}{n}\sum_{i=1}^n \ell(\nu_i(1 - s^{\boldsymbol{p}}_i) + Z_i).$$

Define $q^{\boldsymbol{p}}_{\max} = \sup_{i\le[n]} q^{\boldsymbol{p}}_i$. Let M be the maximum absolute value of score over tokens. Let

$$B = \max_{|x|\le M} -\ell'(x) \ge A = \min_{|x|\le M} -\ell'(x) > 0.$$

Through Taylor's Theorem (integral remainder), we have that

$$B(q^{\boldsymbol{p}}_i\nu_i - Z_i) \ge \ell(\nu_i(1 - q^{\boldsymbol{p}}_i) + Z_i) - \ell(\nu_i) \ge A(q^{\boldsymbol{p}}_i\nu_i - Z_i) \ge \epsilon A\nu_i q^{\boldsymbol{p}}_i.$$

Set $\mathcal{L}_\star = \frac{1}{n}\sum_{i=1}^n \ell(\nu_i)$. Set $C_+ = B(C + \max_{i\in[n]}\nu_i)$ and $C_- = n^{-1}A\epsilon\min_{i\in[n]}\nu_i$. This also implies

$$\begin{aligned} C_+ q^{\boldsymbol{p}}_{\max} \ge \frac{1}{n}\sum_{i\in[n]} B(q^{\boldsymbol{p}}_i\nu_i - Z_i) \ge \mathcal{L}(\boldsymbol{p}) - \mathcal{L}_\star &\ge \frac{1}{n}\sum_{i\in[n]} A(q^{\boldsymbol{p}}_i\nu_i - Z_i) \\ &\ge \frac{1}{n}\sum_{i\in[n]} \epsilon A\nu_i q^{\boldsymbol{p}}_i \ge C_- q^{\boldsymbol{p}}_{\max}. \end{aligned}$$

Thus, to prove $\mathcal{L}(\boldsymbol{p}') > \mathcal{L}(\boldsymbol{p})$, we simply need to establish the stronger statement $C_- q_{\max}^{\boldsymbol{p}'} > C_+ q_{\max}^{\boldsymbol{p}}$.

Going back to the condition of Assumption 2.4, any $\log(q_{\max}^{\boldsymbol{p}}) \le (1+\Delta)\log(q_{\max}^{\boldsymbol{p}'})$ obeys $q_{\max}^{\boldsymbol{p}} \le (q_{\max}^{\boldsymbol{p}'})^{1+\Delta}$ i.e. $q_{\max}^{\boldsymbol{p}'} \ge (q_{\max}^{\boldsymbol{p}})^{(1+\Delta)^{-1}}$. Following above, we wish to ensure $q_{\max}^{\boldsymbol{p}'} > \Theta q_{\max}^{\boldsymbol{p}}$ for such $(\boldsymbol{p}, \boldsymbol{p}')$ pairs where $\Theta = C_+/C_- > 1$. This is guaranteed by

$$(q_{\max}^{\boldsymbol{p}})^{(1+\Delta)^{-1}-1} > \Theta \iff \frac{\Delta}{1+\Delta}\log(q_{\max}^{\boldsymbol{p}}) < -\log(\Theta).$$

The above is satisfied by choosing a $\rho(\Delta) := -2(1+\Delta^{-1})\log(\Theta)$ in Assumption 2.4. Thus, all $\boldsymbol{p}, \boldsymbol{p}'$ with $\log(q_{\max}^{\boldsymbol{p}}) \le \rho = \rho(\Delta)$ satisfies the condition of Assumption 2.4 finishing the proof. ■

A.5 Experimental Details

In this section, we provide additional implementation details for the experiments.

• We build one attention layer using `PyTorch`. During training, we use `SGD` optimizer with learning rate 0.1 and train the model for 1000 iterations. To better visualize the convergence path, we normalize the gradient of $\boldsymbol{p}$ (and $\boldsymbol{v}$) at each iteration.

• Next, given the gradient solution $\boldsymbol{p}$, we determine locally-optimal indices to be those with the highest softmax scores. Using these optimal indices, we utilize python package `cvxopt` to build and solve ($\boldsymbol{p}$-SVM), and then get solution $\boldsymbol{p}^{\mathrm{mm}}$. After obtaining $\boldsymbol{p}^{\mathrm{mm}}$, we also verify that these indices satisfy our local-optimal definition. The examples we use in the paper are all trivial to verify (by construction).

• In Figures 2.4a and 2.4b, $\boldsymbol{v}^{\mathrm{mm}}$ (blue dashed) is solved using python package `sklearn.svm` via ($\boldsymbol{v}$-SVM) based on the given label information, and red dashed line represents $\boldsymbol{p}^{\mathrm{relax}}$ direction instead, which is the solution of (2.11). Note that in both figures, $\boldsymbol{v}^{\mathrm{mm}}/\|\boldsymbol{v}^{\mathrm{mm}}\| = [0, 1]$. Therefore, in Figure 2.4a all optimal tokens are support vectors and $\boldsymbol{p}^{\mathrm{relax}} = \boldsymbol{p}^{\mathrm{mm}}$. Whereas in Figure 2.4b, yellow $\star$ is not a support vector and only needs to satisfy positive correlation with $\boldsymbol{p}$. Gray dashed line displays the $\boldsymbol{p}^{\mathrm{mm}}$ direction.

Failure of gradient descent's global convergence when $n \ge 2$ (refer to Theorem 2.2). Figure A.1 provides a counter-example demonstrating that the $n = 1$ restriction is indeed necessary and tight to guarantee global convergence of the gradient descent iterates $\boldsymbol{p}(t+1) = \boldsymbol{p}(t) - \eta\nabla\mathcal{L}(\boldsymbol{p}(t))$ on (ERM). For this example, we use logistic loss in (ERM). We set $n = T = d = 2$, implying that there is only one non-optimal token, thus Assumption 2.2 is satisfied. The red and blue lines represent `GMM` and `LMM` solutions, respectively. We note that the green star and teal square indicate the locally-optimal tokens. Specifically, referring

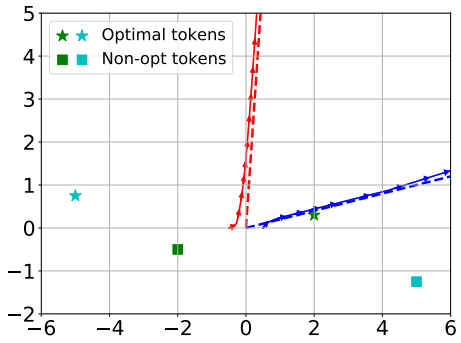


Figure A.1: The convergence behavior of the gradient descent on the attention weights $\boldsymbol{p}$ using the logistic loss in (ERM) with $n = T = d = 2$.

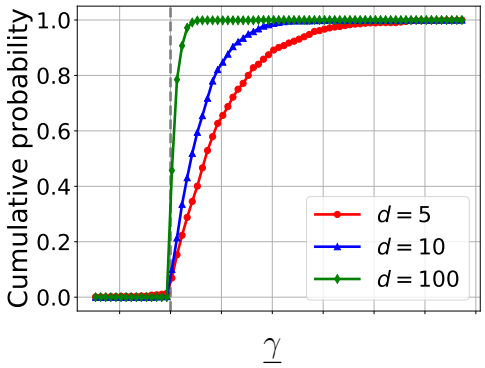


Figure A.2: Cumulative prob. of the gap $\underline{\gamma} := \min_{i\in[n],t\in\mathcal{T}_i} \gamma_{i\alpha_i} - \gamma_{it}$ in Figure 2.3b.

to the local optimality definition (Definition 2.2), for LMM solution ($\boldsymbol{p}^{\mathrm{mm}}$) represented by the blue line, the square teal token does not have any SVM-neighbors. The arrows indicate the two trajectories originating from different initializations. This demonstrates that the gradient descent iterates $\boldsymbol{p}(t+1) = \boldsymbol{p}(t) - \eta\nabla\mathcal{L}(\boldsymbol{p}(t))$ on (ERM) with two different initializations converge to two different SVM solutions (GMM and LMM). Results validate the necessity of $n = 1$ in Theorem 2.2 to provide the gradient descent convergence to $\boldsymbol{p}^{\mathrm{mm}\star}$ from any initialization.

• In Figure 2.3, we again applied normalized gradient descent with a learning rate equal to 1. Each trial involved randomly generated data and training for 1000 iterations as discussed in Theorem 2.4. The tokens selected by $\boldsymbol{p}$ were denoted as $(\alpha_i)_{i=1}^n$, where $\alpha_i = \arg\max_{t\in[T]} \mathbb{S}(\boldsymbol{X}_i\boldsymbol{p})_t$. The averaged softmax probabilities were calculated as $\bar{s} := \frac{1}{n}\sum_{i=1}^n \mathbb{S}(\boldsymbol{X}_i\boldsymbol{p})_{\alpha_i}$ (same as Figure 2.4c). The red bars in Figure 2.3b represent the values of $\mathbb{P}(\bar{s} \le 1-10^{-5})$ for each choice of d. Figure A.2 displays the cumulative probability distribution of $\underline{\gamma}$ from Figure 2.3b, with the gray dashed line indicating $\underline{\gamma} = 0$. From this figure, we observe that the minimal score gap exhibit a sharp transition at zero ($<$1% of the instances have $\underline{\gamma} < 0$), demonstrating that, in most random problem instances with $\|\boldsymbol{p}\| \to \infty$ ($\bar{s} \to 1$), problem directionally converges to an LMM i.e. $\boldsymbol{p}(t)/\|\boldsymbol{p}(t)\| \to \boldsymbol{p}^{\mathrm{mm}}/\|\boldsymbol{p}^{\mathrm{mm}}\|$. We believe the rare occurrence of a negative score gap is due to small score differences (so that optimal token is not clearly distinguished) and finite number of gradient iterations we run. Interestingly, even in the negative score gap scenarios, gradient descent is aligned with $\boldsymbol{p}^{\mathrm{mm}}(\boldsymbol{\alpha})$ (even if $\boldsymbol{p}^{\mathrm{mm}}(\boldsymbol{\alpha})$ is not LMM) which can be predicted from our Section A.4 which handles the scenario where there are multiple optimal tokens per input.

APPENDIX B

Appendix for Chapter 3

B.1 Proof of Theorem 3.1: Separability Under Mild Over-Parameterization

Proof. We denote Kronecker product of two matrices via $\otimes$. Additionally, given $\boldsymbol{W} \in \mathbb{R}^{d\times d}$, let us denote its vectorization $\boldsymbol{w} = \mathrm{vec}(\boldsymbol{W}) \in \mathbb{R}^{d^2}$. We first note that separation is implied by the linear independence of the constraints. Specifically, we are interested in guaranteeing

$$\langle \boldsymbol{w}, \boldsymbol{f}_{it}\rangle \geq 1 \quad \text{for all} \quad i \in [n], \quad t \neq \boldsymbol{\alpha}_i, \quad \text{where} \quad \boldsymbol{f}_{it} := (\boldsymbol{x}_{i\boldsymbol{\alpha}_i} - \boldsymbol{x}_{it}) \otimes \boldsymbol{z}_i.$$

Note that, the inequality constraints above are feasible as soon as $\boldsymbol{f}_{it}$'s are linearly independent. Thus, we will instead prove linear independence of the vectors $(\boldsymbol{f}_{it})_{i\in[n],t\neq\boldsymbol{\alpha}_i}$. Also note that, since there are finitely many $\boldsymbol{\alpha}$ choices, if we show almost sure separation for a fixed but arbitrary $\boldsymbol{\alpha}$ choice, through union bound, we recover the result for all $\boldsymbol{\alpha}$. Thus, we prove the result for a fixed $\boldsymbol{\alpha}$ choice.

We will prove this result inductively. Let $\boldsymbol{M}_{n-1} \in \mathbb{R}^{(n-1)(T-1)\times d^2}$ denote the matrix whose rows are given by the features $(\boldsymbol{f}_{it})_{i\in[n-1],t\neq\boldsymbol{\alpha}_i}$. Suppose the result is correct for $n-1$, thus, $\boldsymbol{M}_{n-1}$ is full row-rank almost surely (post random Gaussian perturbation). Now, fix $\boldsymbol{M}_{n-1}$ and, conditioned on $\boldsymbol{M}_{n-1}$ being full row-rank, let us show that $\boldsymbol{M}_n$ is also full row-rank almost surely. To prove this, consider the n'th example $(\boldsymbol{X}_n, \boldsymbol{z}_n)$. Let $(\boldsymbol{g}_t)_{t=1}^T, \boldsymbol{h} \in \mathbb{R}^d$ be random vectors with i.i.d. $\mathcal{N}(0,\sigma^2)$ entries. Consider the perturbed input $\boldsymbol{X}'_n \in \mathbb{R}^{T\times d}$ with tokens $\boldsymbol{x}'_{nt} = \boldsymbol{x}_{nt} + \boldsymbol{g}_t$ and $\boldsymbol{z}'_n = \boldsymbol{z}_n + \boldsymbol{h}$. Note that for self-attention, we set $\boldsymbol{z}_n = \boldsymbol{x}_{n1}$ and $\boldsymbol{h} = \boldsymbol{g}_1$. From these, create the matrix $\tilde{\boldsymbol{M}}_n \in \mathbb{R}^{(T-1)\times d^2}$ with rows $(\boldsymbol{f}'_{nt})_{t\neq\boldsymbol{\alpha}_n}$ where $\boldsymbol{f}'_{nt} = (\boldsymbol{x}'_{n\boldsymbol{\alpha}_n} - \boldsymbol{x}'_{nt}) \otimes \boldsymbol{z}'_n$. Observe that $\boldsymbol{M}_n = \begin{bmatrix} \tilde{\boldsymbol{M}}_n \\ \boldsymbol{M}_{n-1} \end{bmatrix}$. To conclude with the result, we will apply Lemma B.1. To apply this lemma, we have two claims.

Claim 1: Let $\bar{\boldsymbol{z}}_n$ be the projection of $\boldsymbol{z}'_n$ on the orthogonal complement of $(\boldsymbol{z}_i)_{i=1}^{n-1}$. Consider the matrix $\bar{\boldsymbol{M}}_n$ with rows $\bar{\boldsymbol{f}}_{nt} = (\boldsymbol{x}'_{n\boldsymbol{\alpha}_n} - \boldsymbol{x}'_{nt}) \otimes \bar{\boldsymbol{z}}_n$ for $t \neq \boldsymbol{\alpha}_n$. $\bar{\boldsymbol{M}}_n$ is rank $T-1$ almost surely whenever $d \geq \max(T-1, n)$.

To see this claim, first denote the orthogonal complement of the span of the vectors $(\boldsymbol{z}_i)_{i=1}^{n-1}$ by Z_{n-1}. The span of the vectors $(\boldsymbol{z}_i)_{i=1}^{n-1}$ is at most $n-1$ dimensional and, since $d \geq n$, $\dim(Z_{n-1}) \geq 1$. Consequently, $\bar{\boldsymbol{z}}_n \neq 0$ almost surely because the Gaussian variable $\boldsymbol{z}_n + \boldsymbol{h}$ will have nonzero projection on Z_{n-1} almost surely. Secondly, let $\bar{\boldsymbol{X}} \in \mathbb{R}^{(T-1)\times d}$ be the matrix whose rows are equal to $\boldsymbol{x}'_{n\boldsymbol{\alpha}_n} - \boldsymbol{x}'_{nt}$ for $t \neq \boldsymbol{\alpha}_n$. $\bar{\boldsymbol{X}}$ is full row-rank almost surely, this is because conditioned on $\boldsymbol{g}_{\boldsymbol{\alpha}_n}$, the matrix $\bar{\boldsymbol{X}}$ is written as $\bar{\boldsymbol{X}} = \tilde{\boldsymbol{X}} + \boldsymbol{G}$ where $\tilde{\boldsymbol{X}}$ is deterministic and $\boldsymbol{G}$ is i.i.d. Gaussian. The latter perturbation ensures full row-rank almost surely whenever $T-1 \leq d$. Finally, note that $\bar{\boldsymbol{M}}_n = \bar{\boldsymbol{X}} \otimes \bar{\boldsymbol{z}}_n$. Since the rank of the Kronecker product is multiplicative, we conclude with the claim.

Claim 2: Let $S_{n-1} \subset \mathbb{R}^{d^2}$ be the null space of $\boldsymbol{M}_{n-1}$. There exists a subspace $P \subseteq S_{n-1}$ such that rows of $\bar{\boldsymbol{M}}_n$ are projections of the rows of $\boldsymbol{M}_n$ on P, that is, $\bar{\boldsymbol{f}}_{nt} = \Pi_P(\boldsymbol{f}'_{nt})$ where Π denotes set projection.

To show this claim, let us consider the matrix forms of the vectorized features i.e. let us work with $\mathbb{R}^{d\times d}$ rather than $\mathbb{R}^{d^2}$. Denote the notation change as $\boldsymbol{F}_{it} = (\boldsymbol{x}_{i\boldsymbol{\alpha}_i} - \boldsymbol{x}_{it})\boldsymbol{z}_i^\top \leftrightarrow \boldsymbol{f}_{it} = (\boldsymbol{x}_{i\boldsymbol{\alpha}_i} - \boldsymbol{x}_{it}) \otimes \boldsymbol{z}_i$. Recall that Z_{n-1} denotes the orthogonal complement of $(\boldsymbol{z}_i)_{i=1}^{n-1}$. Define Q to be the set of matrices in $\mathbb{R}^{d\times d}$ whose column space lies in Z_{n-1} and P to be the vectorization of Q. We first show that P is a subset of the null space of S_{n-1}. To see this, fix any matrix $\boldsymbol{A} \in P$ and a row $\boldsymbol{f}_{it}$ from $\boldsymbol{M}_{n-1}$. Matricized $\boldsymbol{f}_{it}$ can be written as $\boldsymbol{F}_{it} = \boldsymbol{a}\boldsymbol{z}_i^\top$ for $\boldsymbol{z}_i \in Z_{n-1}^\perp$. Since $\boldsymbol{A} \in Q$, this implies $\langle \boldsymbol{F}_{it}, \boldsymbol{A}\rangle = \boldsymbol{a}^\top \boldsymbol{A}\boldsymbol{z}_i = 0$ as $\boldsymbol{A}\boldsymbol{z}_i = 0$. This holds for all $\boldsymbol{F}_{it}$, thus, $\texttt{vectorized}(\boldsymbol{A}) \in \texttt{null}(S_{n-1})$.

Next, we need to show that $\bar{\boldsymbol{f}}_{nt}$ is the projection of $\boldsymbol{f}'_{nt}$ on P. To see this, we will show that $\bar{\boldsymbol{f}}_{nt} \in P$ whereas $\boldsymbol{f}'_{nt} - \bar{\boldsymbol{f}}_{nt} \in P^\perp$ for all t. Write $\boldsymbol{F}'_{nt} = \texttt{matricized}(\boldsymbol{f}'_{nt}) = \boldsymbol{a}\boldsymbol{z}'^\top_n$. We have that $\bar{\boldsymbol{F}}_{nt} = \texttt{matricized}(\bar{\boldsymbol{f}}_{nt}) = \boldsymbol{a}\bar{\boldsymbol{z}}_n^\top$ where $\bar{\boldsymbol{z}}_n = \Pi_{Z_{n-1}}(\boldsymbol{z}'_n)$. This implies $\bar{\boldsymbol{F}}_{nt} \in Q$ and $\bar{\boldsymbol{f}}_{nt} \in P$. Similarly, since $\boldsymbol{z}'_n - \bar{\boldsymbol{z}}_n \in Z_{n-1}^\perp$ which implies $\boldsymbol{F}'_{nt} - \bar{\boldsymbol{F}}_{nt} \in Q^\perp$.

To conclude with the proof, observe that, through Claims 1 and 2, $\boldsymbol{M}_n = \begin{bmatrix} \tilde{\boldsymbol{M}}_n \\ \boldsymbol{M}_{n-1} \end{bmatrix}$ satisfies the requirements of Lemma B.1 almost surely, namely, projection of $\tilde{\boldsymbol{M}}_n$ onto a subset of the null space of $\boldsymbol{M}_{n-1}$ being full rank. Thus, $\boldsymbol{M}_n$ is full rank almost surely. ∎

Lemma B.1 *Let $\boldsymbol{A} \in \mathbb{R}^{n\times p}, \boldsymbol{B} \in \mathbb{R}^{m\times p}$. Suppose $n+m \leq p$ and $\boldsymbol{A}$ is full row-rank. Denote the null space of $\boldsymbol{A}$ by $S_{\boldsymbol{A}}^\perp$. Let P be a subspace that is its subset i.e. $P \subseteq S_{\boldsymbol{A}}^\perp$. Let $\boldsymbol{B}'$ be the matrix obtained by projecting each of row of $\boldsymbol{B}$ on P and suppose $\boldsymbol{B}'$ is full rank. Then, the concatenation $\boldsymbol{C} = [\boldsymbol{A}; \boldsymbol{B}]$ is full row-rank.*

Proof. Let $(\boldsymbol{a}_i)_{i=1}^n$, $(\boldsymbol{b}_i)_{i=1}^m$, $(\boldsymbol{b}'_i)_{i=1}^m$ be the rows of $\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{B}'$, respectively. Suppose the set of rows of $\boldsymbol{A}$ and $\boldsymbol{B}$ are linearly dependent. Then, for some $(c_i)_{i=1}^n, (c'_i)_{i=1}^m$ (which are not all-zeros), we have that

$$\sum_{i=1}^{n} c_i \boldsymbol{a}_i + \sum_{i=1}^{m} c'_i \boldsymbol{b}_i = 0. \tag{B.1}$$

We now rewrite this as follows to decouple P and $P^\perp$:

$$\sum_{i=1}^{n} c_i \boldsymbol{a}_i + \sum_{i=1}^{m} c'_i \boldsymbol{b}'_i + \sum_{i=1}^{m} c'_i (\boldsymbol{b}'_i - \boldsymbol{b}_i) = 0.$$

Projecting above inequality to P, we find that $\sum_{i=1}^m c'_i \boldsymbol{b}'_i = 0$. Since $(\boldsymbol{b}'_i)_{i=1}^m$ are linearly independent, we find $c'_i = 0$ for all $i \in [m]$. This implies $\sum_{i=1}^n c_i \boldsymbol{a}_i = 0$. Since $(\boldsymbol{a}_i)_{i=1}^n$ are linearly independent, this implies $c_i = 0$ for all $i \in [n]$. Thus, (B.1) can only hold if all coefficients are zero which is a contradiction. ∎

B.2 Auxiliary Lemmas

B.2.1 Proof of Lemma 3.1

Let $\boldsymbol{W}_\diamond^{\mathrm{mm}}$ denote either solution of (W-SVM) or (W-SVM$_\star$). We claim that $\boldsymbol{W}_\diamond^{\mathrm{mm}}$ is at most rank n. Suppose the claim is wrong and row space of $\boldsymbol{W}_\diamond^{\mathrm{mm}}$ does not lie within $\mathcal{S} = \mathtt{span}(\{\boldsymbol{z}_i\}_{i=1}^n)$. Let $\boldsymbol{W} = \Pi_{\mathcal{S}}(\boldsymbol{W}_\diamond^{\mathrm{mm}})$ denote the matrix obtained by projecting the rows of $\boldsymbol{W}_\diamond^{\mathrm{mm}}$ on $\mathcal{S}$. Observe that $\boldsymbol{W}$ satisfies all SVM constraints since $\boldsymbol{W}\boldsymbol{z}_i = \boldsymbol{W}_\diamond^{\mathrm{mm}}\boldsymbol{z}_i$ for all $i \in [n]$. For Frobenius norm, using $\boldsymbol{W}_\diamond^{\mathrm{mm}} \neq \boldsymbol{W}$, we obtain a contradiction via $\|\boldsymbol{W}_\diamond^{\mathrm{mm}}\|_F^2 = \|\boldsymbol{W}\|_F^2 + \|\boldsymbol{W}_\diamond^{\mathrm{mm}} - \boldsymbol{W}\|_F^2 > \|\boldsymbol{W}\|_F^2$. For nuclear norm, we can write $\boldsymbol{W} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^\top$ with $\boldsymbol{\Sigma} \in \mathbb{R}^{r\times r}$ where r is dimension of $\mathcal{S}$ and $\mathtt{column\_span}(\boldsymbol{V}) = \mathcal{S}$.

To proceed, we split the problem into two scenarios.

Scenario 1: Let $\boldsymbol{U}_\perp, \boldsymbol{V}_\perp$ be orthogonal complements of $\boldsymbol{U}, \boldsymbol{V}$ – viewing matrices with orthonormal columns as subspaces. Suppose $\boldsymbol{U}_\perp^\top \boldsymbol{W}_\diamond^{\mathrm{mm}} \boldsymbol{V}_\perp \neq 0$. Then, singular value inequalities (which were also used in earlier works on nuclear norm analysis [167, 147, 148]) guarantee that $\|\boldsymbol{W}_\diamond^{\mathrm{mm}}\|_\star \geq \|\boldsymbol{U}^\top \boldsymbol{W}_\diamond^{\mathrm{mm}} \boldsymbol{V}\|_\star + \|\boldsymbol{U}_\perp^\top \boldsymbol{W}_\diamond^{\mathrm{mm}} \boldsymbol{V}_\perp\|_\star > \|\boldsymbol{W}\|_\star$.

Scenario 2: Now suppose $\boldsymbol{U}_\perp^\top \boldsymbol{W}_\diamond^{\mathrm{mm}} \boldsymbol{V}_\perp = 0$. Since $\boldsymbol{W}_\diamond^{\mathrm{mm}} \boldsymbol{V}_\perp \neq 0$, this implies $\boldsymbol{U}^\top \boldsymbol{W}_\diamond^{\mathrm{mm}} \boldsymbol{V}_\perp \neq 0$. Let $\boldsymbol{W}' = \boldsymbol{U}\boldsymbol{U}^\top \boldsymbol{W}_\diamond^{\mathrm{mm}}$ which is a rank-r matrix. Since $\boldsymbol{W}'$ is a subspace projection, we have $\|\boldsymbol{W}'\|_\star \leq \|\boldsymbol{W}_\diamond^{\mathrm{mm}}\|_\star$. Next, observe that $\|\boldsymbol{W}\|_\star = \mathtt{trace}(\boldsymbol{U}^\top \boldsymbol{W} \boldsymbol{V}) = \mathtt{trace}(\boldsymbol{U}^\top \boldsymbol{W}' \boldsymbol{V})$. On the other hand, $\mathtt{trace}(\boldsymbol{U}^\top \boldsymbol{W}' \boldsymbol{V}) < \|\boldsymbol{W}'\|_\star$ because the equality in *von Neumann's*

trace inequality happens if and only if the two matrices we are inner-producting, namely $(\boldsymbol{W}', \boldsymbol{U}\boldsymbol{V}^\top)$, share a joint set of singular vectors [28]. However, this is not true as the row space of $\boldsymbol{W}_\diamond^{\mathrm{mm}}$ does not lie within $\mathcal{S}$. Thus, we obtain $\|\boldsymbol{W}\|_\star < \|\boldsymbol{W}'\|_\star \leq \|\boldsymbol{W}_\diamond^{\mathrm{mm}}\|_\star$ concluding the proof via contradiction. ■

B.2.2 Proof of Lemma 3.2

We first show that $\mathcal{L}(\boldsymbol{W}) > \mathcal{L}_\star = \frac{1}{n}\sum_{i=1}^n \ell(\gamma_{i\mathrm{opt}_i})$. The token at the output of the attention layer is given by $\boldsymbol{a}_i = \boldsymbol{X}_i^\top \boldsymbol{s}_i$, where $\mathbb{S}(\boldsymbol{X}_i \boldsymbol{W} \boldsymbol{z}_i) = \boldsymbol{s}_i$. Here, $\boldsymbol{a}_i$ can be written as $\boldsymbol{a}_i = \sum_{t\in[T]} \boldsymbol{s}_{it}\boldsymbol{x}_{it}$, where $\boldsymbol{s}_{it} \geq 0$ and $\sum_{t\in[T]} \boldsymbol{s}_{it} = 1$. To proceed, using the linearity of $h(\boldsymbol{x}) = \boldsymbol{v}^\top \boldsymbol{x}$, we find that

$$\begin{aligned}\mathcal{L}(\boldsymbol{W}) = \frac{1}{n}\sum_{i=1}^n \ell(y_i \cdot h(\boldsymbol{a}_i)) &= \frac{1}{n}\sum_{i=1}^n \ell(y_i \cdot \sum_{t\in[T]} \boldsymbol{s}_{it} h(\boldsymbol{x}_{it})) \\ &\geq \frac{1}{n}\sum_{i=1}^n \ell(y_i \cdot h(\boldsymbol{x}_{i\mathrm{opt}_i})) = \frac{1}{n}\sum_{i=1}^n \ell(\gamma_{i\mathrm{opt}_i}) = \mathcal{L}_\star. \end{aligned} \tag{B.2}$$

Here, the inequality follows since $\gamma_{it} = y_i \cdot h(\boldsymbol{x}_{it}) = y_i \cdot \boldsymbol{v}^\top \boldsymbol{x}_{it} \leq \gamma_{i\mathrm{opt}_i}$ by Definition 3.1 and strictly-decreasing nature of the loss ℓ due to Assumption 3.1.

On the other hand, since not all tokens are optimal, there exists a token index (i,t) for which $y_i \cdot h(\boldsymbol{x}_{it}) < y_i \cdot h(\boldsymbol{x}_{i\mathrm{opt}_i})$. Since all softmax entries obey $\boldsymbol{s}_{it} > 0$ for finite $\boldsymbol{W}$, this implies the strict inequality $\ell(y_i \cdot h(\boldsymbol{a}_i)) > \ell(y_i \cdot h(\boldsymbol{x}_{i\mathrm{opt}_i}))$. This leads to the desired conclusion $\mathcal{L}(\boldsymbol{W}) > \mathcal{L}_\star$.

Next, we show that if (W-SVM) is feasible i.e. there exists a $\boldsymbol{W}$ separating some optimal indices $(\mathrm{opt}_i)_{i=1}^n$ from the other tokens, then $\lim_{R\to\infty} \mathcal{L}(R \cdot \boldsymbol{W}) = \mathcal{L}_\star$. Note that, this assumption does not exclude the existence of other optimal indices. This implies that, letting $\lim_{R\to\infty} \mathbb{S}(\boldsymbol{X}_i(R \cdot \boldsymbol{W})\boldsymbol{z}_i)$ saturates the softmax and will be equal to the indicator function at opt_i for all inputs $i \in [n]$. Thus, $\boldsymbol{s}_{it} \to 0$ for $t \neq \mathrm{opt}_i$ and $\boldsymbol{s}_{it} \to 1$ for $t = \mathrm{opt}_i$. Using M_1-Lipschitzness of ℓ, we can write

$$\left|\ell(y_i \cdot h(\boldsymbol{x}_{i\mathrm{opt}_i})) - \ell(y_i \cdot h(\boldsymbol{a}_i))\right| \leq M_1 \left|h(\boldsymbol{a}_i) - h(\boldsymbol{x}_{i\mathrm{opt}_i})\right|.$$

Since h is linear, it is $\|\boldsymbol{v}\|$-Lipschitz implying

$$\left|\ell(y_i \cdot h(\boldsymbol{x}_{i\mathrm{opt}_i})) - \ell(y_i \cdot h(\boldsymbol{a}_i))\right| \leq M_1 \|\boldsymbol{v}\| \cdot \|\boldsymbol{a}_i - \boldsymbol{x}_{i\mathrm{opt}_i}\|.$$

Since $\boldsymbol{a}_i \to \boldsymbol{x}_{i\mathrm{opt}_i}$ as $R \to \infty$, (B.2) gives $\lim_{R\to\infty} \mathcal{L}(R \cdot \boldsymbol{W}) = \mathcal{L}_\star$. ■

B.2.3 Proof of Lemma 3.4

Proof. Let

$$\gamma_i = y_i \cdot \boldsymbol{X}_i \boldsymbol{v}, \quad \boldsymbol{h}_i = \boldsymbol{X}_i \boldsymbol{W} \boldsymbol{z}_i.$$

From Assumption 3.1, $\ell : \mathbb{R} \to \mathbb{R}$ is differentiable. Hence, the gradient evaluated at $\boldsymbol{W}$ is given by

$$\nabla \mathcal{L}(\boldsymbol{W}) = \frac{1}{n} \sum_{i=1}^{n} \ell' \left(\gamma_i^\top \mathbb{S}(\boldsymbol{h}_i) \right) \cdot \boldsymbol{X}_i^\top \mathbb{S}'(\boldsymbol{h}_i) \gamma_i \boldsymbol{z}_i^\top, \tag{B.3}$$

where

$$\mathbb{S}'(\boldsymbol{h}) = \operatorname{diag}\left(\mathbb{S}(\boldsymbol{h}) \right) - \mathbb{S}(\boldsymbol{h}) \mathbb{S}(\boldsymbol{h})^\top \in \mathbb{R}^{T \times T}. \tag{B.4}$$

Note that

$$\| \mathbb{S}'(\boldsymbol{h}) \| \leq \| \mathbb{S}'(\boldsymbol{h}) \|_F \leq 1. \tag{B.5}$$

Hence, for any $\boldsymbol{W}, \dot{\boldsymbol{W}} \in \mathbb{R}^{d \times d}$, $i \in [n]$, we have

$$\left\| \mathbb{S}(\boldsymbol{h}_i) - \mathbb{S}(\dot{\boldsymbol{h}}_i) \right\| \leq \left\| \boldsymbol{h}_i - \dot{\boldsymbol{h}}_i \right\| \leq \| \boldsymbol{X}_i \| \ \| \boldsymbol{z}_i \| \ \left\| \boldsymbol{W} - \dot{\boldsymbol{W}} \right\|_F, \tag{B.6a}$$

where $\dot{\boldsymbol{h}}_i = \boldsymbol{X}_i \dot{\boldsymbol{W}} \boldsymbol{z}_i$.

Similarly,

$$\begin{aligned} \left\| \mathbb{S}'(\boldsymbol{h}_i) - \mathbb{S}'(\dot{\boldsymbol{h}}_i) \right\|_F &\leq \left\| \mathbb{S}(\boldsymbol{h}_i) - \mathbb{S}(\dot{\boldsymbol{h}}_i) \right\| + \left\| \mathbb{S}(\boldsymbol{h}_i) \mathbb{S}(\boldsymbol{h}_i)^\top - \mathbb{S}(\dot{\boldsymbol{h}}_i) \mathbb{S}(\dot{\boldsymbol{h}}_i)^\top \right\|_F \\ &\leq 3 \| \boldsymbol{X}_i \| \ \| \boldsymbol{z}_i \| \ \left\| \boldsymbol{W} - \dot{\boldsymbol{W}} \right\|_F. \end{aligned} \tag{B.6b}$$

Next, for any $\boldsymbol{W}, \dot{\boldsymbol{W}} \in \mathbb{R}^{d \times d}$, we get

$$\begin{aligned} \left\| \nabla \mathcal{L}(\boldsymbol{W}) - \nabla \mathcal{L}(\dot{\boldsymbol{W}}) \right\|_F &\leq \frac{1}{n} \sum_{i=1}^{n} \left\| \ell' \left(\gamma_i^\top \mathbb{S}(\boldsymbol{h}_i) \right) \cdot \boldsymbol{z}_i \gamma_i^\top \mathbb{S}'(\boldsymbol{h}_i) \boldsymbol{X}_i - \ell' \left(\gamma_i^\top \mathbb{S}(\dot{\boldsymbol{h}}_i) \right) \cdot \boldsymbol{z}_i \gamma_i^\top \mathbb{S}'(\dot{\boldsymbol{h}}_i) \boldsymbol{X}_i \right\|_F \\ &\leq \frac{1}{n} \sum_{i=1}^{n} \left\| \boldsymbol{z}_i \gamma_i^\top \mathbb{S}'(\dot{\boldsymbol{h}}_i) \boldsymbol{X}_i \right\|_F \ \left| \ell' \left(\gamma_i^\top \mathbb{S}(\boldsymbol{h}_i) \right) - \ell' \left(\gamma_i^\top \mathbb{S}(\dot{\boldsymbol{h}}_i) \right) \right| \\ &\quad + \frac{1}{n} \sum_{i=1}^{n} \left| \ell' \left(\gamma_i^\top \mathbb{S}(\boldsymbol{h}_i) \right) \right| \ \left\| \boldsymbol{z}_i \gamma_i^\top \mathbb{S}'(\boldsymbol{h}_i) \boldsymbol{X}_i - \boldsymbol{z}_i \gamma_i^\top \mathbb{S}'(\dot{\boldsymbol{h}}_i) \boldsymbol{X}_i \right\|_F \\ &\leq \frac{1}{n} \sum_{i=1}^{n} M_0 \ \| \gamma_i \|^2 \ \| \boldsymbol{z}_i \| \ \| \boldsymbol{X}_i \| \ \left\| \mathbb{S}(\boldsymbol{h}_i) - \mathbb{S}(\dot{\boldsymbol{h}}_i) \right\| \\ &\quad + \frac{1}{n} \sum_{i=1}^{n} M_1 \ \| \gamma_i \| \ \| \boldsymbol{z}_i \| \ \| \boldsymbol{X}_i \| \ \left\| \mathbb{S}'(\boldsymbol{h}_i) - \mathbb{S}'(\dot{\boldsymbol{h}}_i) \right\|_F, \end{aligned} \tag{B.7}$$

where the second inequality follows from the fact that $|ab - cd| \le |d||a - c| + |a||b - d|$ and the third inequality uses Assumption 3.1 and (B.5).

Substituting (B.6a) and (B.6b) into (B.7), we get

$$\begin{aligned}\left\|\nabla\mathcal{L}(\boldsymbol{W}) - \nabla\mathcal{L}(\dot{\boldsymbol{W}})\right\|_F &\le \frac{1}{n}\sum_{i=1}^{n}\left(M_0\ \|\gamma_i\|^2\|\boldsymbol{z}_i\|^2\|\boldsymbol{X}_i\|^2 + 3M_1\|\gamma_i\|\ \|\boldsymbol{z}_i\|^2\ \|\boldsymbol{X}_i\|^2\right)\|\boldsymbol{W} - \dot{\boldsymbol{W}}\|_F \\ &\le \frac{1}{n}\sum_{i=1}^{n}\left(M_0\ \|\boldsymbol{v}\|^2\|\boldsymbol{z}_i\|^2\|\boldsymbol{X}_i\|^4 + 3M_1\|\boldsymbol{v}\|\ \|\boldsymbol{z}_i\|^2\ \|\boldsymbol{X}_i\|^3\right)\|\boldsymbol{W} - \dot{\boldsymbol{W}}\|_F \\ &\le L_{\boldsymbol{W}}\ \|\boldsymbol{W} - \dot{\boldsymbol{W}}\|_F,\end{aligned}$$

where $L_{\boldsymbol{W}}$ is defined in (3.3).

Let $\boldsymbol{g}_i = \boldsymbol{X}_i\boldsymbol{W}_k\boldsymbol{W}_q^\top\boldsymbol{z}_i$. We have

$$\nabla_{\boldsymbol{W}_k}\mathcal{L}(\boldsymbol{W}_k, \boldsymbol{W}_q) = \frac{1}{n}\sum_{i=1}^{n}\ell'\left(\gamma_i^\top\mathbb{S}(\boldsymbol{g}_i)\right)\cdot\boldsymbol{z}_i\gamma_i^\top\mathbb{S}'(\boldsymbol{g}_i)\boldsymbol{X}_i\boldsymbol{W}_q, \tag{B.8a}$$

$$\nabla_{\boldsymbol{W}_q}\mathcal{L}(\boldsymbol{W}_k, \boldsymbol{W}_q) = \frac{1}{n}\sum_{i=1}^{n}\ell'\left(\gamma_i^\top\mathbb{S}(\boldsymbol{g}_i)\right)\cdot\boldsymbol{X}_i^\top\mathbb{S}'(\boldsymbol{g}_i)\gamma_i\boldsymbol{z}_i^\top\boldsymbol{W}_k. \tag{B.8b}$$

By the similar argument as in (B.7), for any $\boldsymbol{W}_q$ and $\dot{\boldsymbol{W}}_q \in \mathbb{R}^{d\times m}$, we have

$$\begin{aligned}&\left\|\nabla_{\boldsymbol{W}_q}\mathcal{L}(\boldsymbol{W}_k, \boldsymbol{W}_q) - \nabla_{\boldsymbol{W}_q}\mathcal{L}(\boldsymbol{W}_k, \dot{\boldsymbol{W}}_q)\right\|_F \\ &\le \frac{\|\boldsymbol{W}_k\|}{n}\sum_{i=1}^{n}\left\|\ell'\left(\gamma_i^\top\mathbb{S}(\boldsymbol{h}_i)\right)\cdot\boldsymbol{z}_i\gamma_i^\top\mathbb{S}'(\boldsymbol{h}_i)\boldsymbol{X}_i - \ell'\left(\gamma_i^\top\mathbb{S}(\dot{\boldsymbol{h}}_i)\right)\cdot\boldsymbol{z}_i\gamma_i^\top\mathbb{S}'(\dot{\boldsymbol{h}}_i)\boldsymbol{X}_i\right\|_F \\ &\le L_{\boldsymbol{W}}\|\boldsymbol{W}_k\|\ \|\boldsymbol{W}_q - \dot{\boldsymbol{W}}_q\|_F.\end{aligned} \tag{B.9}$$

Similarly, for any $\boldsymbol{W}_k, \dot{\boldsymbol{W}}_k \in \mathbb{R}^{d\times m}$, we get

$$\left\|\nabla_{\boldsymbol{W}_k}\mathcal{L}(\boldsymbol{W}_k, \boldsymbol{W}_q) - \nabla_{\boldsymbol{W}_k}\mathcal{L}(\dot{\boldsymbol{W}}_k, \boldsymbol{W}_q)\right\|_F \le L_{\boldsymbol{W}}\|\boldsymbol{W}_q\|\ \|\boldsymbol{W}_k - \dot{\boldsymbol{W}}_k\|_F.$$

■

B.2.4 A useful lemma for gradient descent analysis

Lemma B.2 *For any $\boldsymbol{X} \in \mathbb{R}^{T\times d}$, $\boldsymbol{W}, \boldsymbol{V} \in \mathbb{R}^{d\times d}$ and $\boldsymbol{z}, \boldsymbol{v} \in \mathbb{R}^d$, let $\boldsymbol{a} = \boldsymbol{X}\boldsymbol{V}\boldsymbol{z}$, $\boldsymbol{s} = \mathbb{S}(\boldsymbol{X}\boldsymbol{W}\boldsymbol{z})$, and $\gamma = \boldsymbol{X}\boldsymbol{v}$. Set*

$$\Gamma = \sup_{t,\tau\in[T]} |\gamma_t - \gamma_\tau| \quad \text{and} \quad A = \sup_{t\in[T]} \|\boldsymbol{a}_t\|.$$

We have that

$$\left| \boldsymbol{a}^\top \mathrm{diag}(\boldsymbol{s})\gamma - \boldsymbol{a}^\top \boldsymbol{s}\boldsymbol{s}^\top \gamma - \sum_{t\geq 2}^{T} (\boldsymbol{a}_1 - \boldsymbol{a}_t)\boldsymbol{s}_t(\gamma_1 - \gamma_t) \right| \leq 2\Gamma A(1 - \boldsymbol{s}_1)^2.$$

Proof. The proof is similar to [186, Lemma 4], but for the sake of completeness, we provide it here. Set $\bar{\gamma} = \sum_{t=1}^{T} \gamma_t \boldsymbol{s}_t$. We have

$$\gamma_1 - \bar{\gamma} = \sum_{t\geq 2}^{T} (\gamma_1 - \gamma_t)\boldsymbol{s}_t, \quad \text{and} \quad |\gamma_1 - \bar{\gamma}| \leq \Gamma(1 - \boldsymbol{s}_1).$$

Then,

$$\begin{aligned} \boldsymbol{a}^\top \mathrm{diag}(\boldsymbol{s})\gamma - \boldsymbol{a}^\top \boldsymbol{s}\boldsymbol{s}^\top \gamma &= \sum_{t=1}^{T} \boldsymbol{a}_t \gamma_t \boldsymbol{s}_t - \sum_{t=1}^{T} \boldsymbol{a}_t \boldsymbol{s}_t \sum_{t=1}^{T} \gamma_t \boldsymbol{s}_t \\ &= \boldsymbol{a}_1 \boldsymbol{s}_1 (\gamma_1 - \bar{\gamma}) - \sum_{t\geq 2}^{T} \boldsymbol{a}_t \boldsymbol{s}_t (\bar{\gamma} - \gamma_t). \end{aligned}$$

Since

$$\left| \sum_{t\geq 2}^{T} \boldsymbol{a}_t \boldsymbol{s}_t (\bar{\gamma} - \gamma_t) - \sum_{t\geq 2}^{T} \boldsymbol{a}_t \boldsymbol{s}_t (\gamma_1 - \gamma_t) \right| \leq A\Gamma(1 - \boldsymbol{s}_1)^2,$$

we obtain[1]

$$\begin{aligned}\boldsymbol{a}^\top \mathrm{diag}(\boldsymbol{s})\gamma - \boldsymbol{a}^\top \boldsymbol{s}\boldsymbol{s}^\top \gamma &= \boldsymbol{a}_1 \boldsymbol{s}_1(\gamma_1 - \bar{\gamma}) - \sum_{t\geq 2}^{T} \boldsymbol{a}_t \boldsymbol{s}_t(\gamma_1 - \gamma_t) \pm A\Gamma(1-\boldsymbol{s}_1)^2 \\ &= \boldsymbol{a}_1 \boldsymbol{s}_1 \sum_{t\geq 2}^{T}(\gamma_1 - \gamma_t)\boldsymbol{s}_t - \sum_{t\geq 2}^{T} \boldsymbol{a}_t \boldsymbol{s}_t(\gamma_1 - \gamma_t) \pm A\Gamma(1-\boldsymbol{s}_1)^2 \\ &= \sum_{t\geq 2}^{T}(\boldsymbol{a}_1 \boldsymbol{s}_1 - \boldsymbol{a}_t)\boldsymbol{s}_t(\gamma_1 - \gamma_t) \pm A\Gamma(1-\boldsymbol{s}_1)^2 \\ &= \sum_{t\geq 2}^{T}(\boldsymbol{a}_1 - \boldsymbol{a}_t)\boldsymbol{s}_t(\gamma_1 - \gamma_t) \pm 2A\Gamma(1-\boldsymbol{s}_1)^2.\end{aligned}$$

Here, $\pm$ on the right handside uses the fact that

$$\left|\sum_{t\geq 2}^{T}(\boldsymbol{a}_1 \boldsymbol{s}_1 - \boldsymbol{a}_1)\boldsymbol{s}_t(\gamma_1 - \gamma_t)\right| \leq (1-\boldsymbol{s}_1)A\Gamma \sum_{t\geq 2}^{T} \boldsymbol{s}_t = (1-\boldsymbol{s}_1)^2 A\Gamma.$$

■

B.3 Global Convergence of Gradient Descent

B.3.1 Divergence of norm of the iterates

The next lemma establishes the descent property of gradient descent for $\mathcal{L}(\boldsymbol{W})$ under Assumption 3.1.

Lemma B.3 (Descent Lemma) *Under Assumption 3.1, if $\eta \leq 1/L_{\boldsymbol{W}}$, then for any initialization $\boldsymbol{W}(0)$, Algorithm W-GD satisfies:*

$$\mathcal{L}(\boldsymbol{W}(k+1)) - \mathcal{L}(\boldsymbol{W}(k)) \leq -\frac{\eta}{2}\|\nabla \mathcal{L}(\boldsymbol{W}(k))\|_F^2, \tag{B.10}$$

for all $k \geq 0$. Additionally, it holds that $\sum_{k=0}^{\infty} \|\nabla \mathcal{L}(\boldsymbol{W}(k))\|_F^2 < \infty$, and $\lim_{k\to\infty} \|\nabla \mathcal{L}(\boldsymbol{W}(k))\|_F^2 = 0$.

Proof. The proof is similar to [186, Lemma 6]. ■

[1]For simplicity, we use $\pm$ on the right hand side to denote the upper and lower bounds.

B.3.1.1 Proof of Theorem 3.3

Proof of T1. Let

$$\bar{\boldsymbol{h}}_i = \boldsymbol{X}_i \boldsymbol{W}^{\mathrm{mm}} \boldsymbol{z}_i, \quad \gamma_i = y_i \cdot \boldsymbol{X}_i \boldsymbol{v}, \quad \text{and} \quad \boldsymbol{h}_i = \boldsymbol{X}_i \boldsymbol{W} \boldsymbol{z}_i. \tag{B.11}$$

Let us recall the gradient evaluated at $\boldsymbol{W}$ which is given by

$$\nabla \mathcal{L}(\boldsymbol{W}) = \frac{1}{n} \sum_{i=1}^{n} \ell' \left(\gamma_i^\top \mathbb{S}(\boldsymbol{h}_i) \right) \cdot \boldsymbol{X}_i^\top \mathbb{S}'(\boldsymbol{h}_i) \gamma_i \boldsymbol{z}_i^\top, \tag{B.12}$$

which implies that

$$\begin{aligned}
\langle \nabla \mathcal{L}(\boldsymbol{W}), \boldsymbol{W}^{\mathrm{mm}} \rangle &= \frac{1}{n} \sum_{i=1}^{n} \ell' \left(\gamma_i^\top \mathbb{S}(\boldsymbol{h}_i) \right) \cdot \left\langle \boldsymbol{X}_i^\top \mathbb{S}'(\boldsymbol{h}_i) \gamma_i \boldsymbol{z}_i^\top, \boldsymbol{W}^{\mathrm{mm}} \right\rangle \\
&= \frac{1}{n} \sum_{i=1}^{n} \ell_i' \cdot \mathtt{tr} \left((\boldsymbol{W}^{\mathrm{mm}})^\top \boldsymbol{X}_i^\top \mathbb{S}'(\boldsymbol{h}_i) \gamma_i \boldsymbol{z}_i^\top \right) \\
&= \frac{1}{n} \sum_{i=1}^{n} \ell_i' \cdot \bar{\boldsymbol{h}}_i^\top \mathbb{S}'(\boldsymbol{h}_i) \gamma_i \\
&= \frac{1}{n} \sum_{i=1}^{n} \ell_i' \cdot \left(\bar{\boldsymbol{h}}_i^\top \mathrm{diag}(\boldsymbol{s}_i) \gamma_i - \bar{\boldsymbol{h}}_i^\top \boldsymbol{s}_i \boldsymbol{s}_i^\top \gamma_i \right).
\end{aligned} \tag{B.13}$$

Here, let $\ell_i' := \ell'(\gamma_i^\top \mathbb{S}(\boldsymbol{h}_i))$, $\boldsymbol{s}_i = \mathbb{S}(\boldsymbol{h}_i)$ and the third equality uses $\mathtt{tr}\left(\boldsymbol{b}\boldsymbol{a}^\top\right) = \boldsymbol{a}^\top \boldsymbol{b}$.

In order to move forward, we will establish the following result, with a focus on the equal score condition (Assumption **A2**): Let $\gamma = \gamma_{t \geq 2}$ be a constant, and let γ_1 and $\bar{\boldsymbol{h}}_1$ represent the largest indices of vectors γ and $\bar{\boldsymbol{h}}$ respectively. For any vector $\boldsymbol{s}$ that satisfies $\sum_{t \in [T]} \boldsymbol{s}_t = 1$ and $\boldsymbol{s}_t > 0$, we aim to prove that $\bar{\boldsymbol{h}}^\top \mathrm{diag}(\boldsymbol{s}) \gamma - \bar{\boldsymbol{h}}^\top \boldsymbol{s}\boldsymbol{s}^\top \gamma > 0$. To demonstrate

this, we proceed by writing the following:

$$\begin{aligned}
\bar{\boldsymbol{h}}^\top \mathrm{diag}(\boldsymbol{s})\gamma - \bar{\boldsymbol{h}}^\top \boldsymbol{s}\boldsymbol{s}^\top \gamma &= \sum_{t=1}^{T} \bar{\boldsymbol{h}}_t \gamma_t \boldsymbol{s}_t - \sum_{t=1}^{T} \bar{\boldsymbol{h}}_t \boldsymbol{s}_t \sum_{t=1}^{T} \gamma_t \boldsymbol{s}_t \\
&= \left(\bar{\boldsymbol{h}}_1 \gamma_1 \boldsymbol{s}_1 + \gamma \sum_{t\geq 2}^{T} \bar{\boldsymbol{h}}_t \boldsymbol{s}_t \right) - \Big(\gamma_1 \boldsymbol{s}_1 + \gamma(1-\boldsymbol{s}_1) \Big) \left(\bar{\boldsymbol{h}}_1 \boldsymbol{s}_1 + \sum_{t\geq 2}^{T} \bar{\boldsymbol{h}}_t \boldsymbol{s}_t \right) \\
&= \bar{\boldsymbol{h}}_1 (\gamma_1 - \gamma) \boldsymbol{s}_1 (1-\boldsymbol{s}_1) - (\gamma_1 - \gamma) \boldsymbol{s}_1 \sum_{t\geq 2}^{T} \bar{\boldsymbol{h}}_t \boldsymbol{s}_t \\
&= (\gamma_1 - \gamma)(1-\boldsymbol{s}_1)\boldsymbol{s}_1 \left[\bar{\boldsymbol{h}}_1 - \frac{\sum_{t\geq 2}^{T} \bar{\boldsymbol{h}}_t \boldsymbol{s}_t}{\sum_{t\geq 2}^{T} \boldsymbol{s}_t} \right] \\
&\geq (\gamma_1 - \gamma)(1-\boldsymbol{s}_1)\boldsymbol{s}_1 (\bar{\boldsymbol{h}}_1 - \max_{t\geq 2} \bar{\boldsymbol{h}}_t).
\end{aligned} \tag{B.14}$$

To proceed, define

$$\boldsymbol{\gamma}^i_{\mathrm{gap}} = \gamma_{i\mathrm{opt}_i} - \max_{t\neq \mathrm{opt}_i} \gamma_{it} \quad \text{and} \quad \bar{h}^i_{\mathrm{gap}} = \bar{\boldsymbol{h}}_{i\mathrm{opt}_i} - \max_{t\neq \mathrm{opt}_i} \bar{\boldsymbol{h}}_{it}.$$

With these, we obtain

$$\bar{\boldsymbol{h}}_i^\top \mathrm{diag}(\boldsymbol{s}_i)\gamma_i - \bar{\boldsymbol{h}}_i^\top \boldsymbol{s}_i \boldsymbol{s}_i^\top \gamma_i \geq \boldsymbol{\gamma}^i_{\mathrm{gap}} \bar{h}^i_{\mathrm{gap}} (1 - \boldsymbol{s}_{i\mathrm{opt}_i}) \boldsymbol{s}_{i\mathrm{opt}_i}. \tag{B.15}$$

Note that

$$\begin{aligned}
&\bar{h}^i_{\mathrm{gap}} = \min_{t\neq \mathrm{opt}_i} \ (\boldsymbol{x}_{i\mathrm{opt}_i} - \boldsymbol{x}_{it})^\top \boldsymbol{W}^{\mathrm{mm}} \boldsymbol{z}_i \geq 1, \\
&\boldsymbol{\gamma}^i_{\mathrm{gap}} = \min_{t\neq \mathrm{opt}_i} \ \gamma_{i\mathrm{opt}_i} - \gamma_{it} > 0, \\
&\boldsymbol{s}_{i\mathrm{opt}_i}(1 - \boldsymbol{s}_{i\mathrm{opt}_i}) > 0.
\end{aligned}$$

Hence,

$$\min_{i\in[n]} \left\{ \left(\min_{t\neq \mathrm{opt}_i} \ (\boldsymbol{x}_{i\mathrm{opt}_i} - \boldsymbol{x}_{it})^\top \boldsymbol{W}^{\mathrm{mm}} \boldsymbol{z}_i \right) \cdot \left(\min_{t\neq \mathrm{opt}_i} \ \gamma_{i\mathrm{opt}_i} - \gamma_{it} \right) \cdot \boldsymbol{s}_{i\mathrm{opt}_i}(1 - \boldsymbol{s}_{i\mathrm{opt}_i}) \right\} > 0. \tag{B.16}$$

It follows from (B.15) and (B.16) that

$$\min_{i\in[n]} \left\{ \bar{\boldsymbol{h}}_i^\top \mathrm{diag}(\boldsymbol{s}_i)\gamma_i - \bar{\boldsymbol{h}}_i^\top \boldsymbol{s}_i \boldsymbol{s}_i^\top \gamma_i \right\} > 0. \tag{B.17}$$

Further, by Assumption 3.1, $\ell'_i < 0$, ℓ' is continuous and the domain is bounded, the maxi-

mum is attained and negative, and thus

$$\max_x \ell'(x) < 0. \tag{B.18}$$

Hence, using (B.17) and (B.18) in (B.13), we obtain

$$\langle \nabla \mathcal{L}(\boldsymbol{W}), \boldsymbol{W}^{\mathrm{mm}} \rangle < 0. \tag{B.19}$$

In the scenario that Assumption **A1** holds (all tokens are support), $\bar{\boldsymbol{h}}_t = \boldsymbol{x}_{it}^\top \boldsymbol{W}^{\mathrm{mm}} \boldsymbol{z}_i$ is constant for all $t \geq 2$. Hence, following similar steps as in (B.14) completes the proof.

Proof of T2. It follows from Lemma B.3 that under Assumption 3.1, $\eta \leq 1/L_{\boldsymbol{W}}$, and for any initialization $\boldsymbol{W}(0)$, the gradient descent sequence $\boldsymbol{W}(k+1) = \boldsymbol{W}(k) - \eta \nabla \mathcal{L}(\boldsymbol{W}(k))$ satisfies $\lim_{k\to\infty} \|\nabla \mathcal{L}\left(\boldsymbol{W}\left(k\right)\right)\|_F^2 = 0$.

Further, it follows from **T1** that $\langle \nabla \mathcal{L}(\boldsymbol{W}), \boldsymbol{W}^{\mathrm{mm}} \rangle < 0$ for all $\boldsymbol{W} \in \mathbb{R}^{d\times d}$. Hence, for any finite $\boldsymbol{W}$, $\langle \nabla \mathcal{L}(\boldsymbol{W}), \boldsymbol{W}^{\mathrm{mm}} \rangle$ cannot be equal to zero. Therefore, there are no finite critical points $\boldsymbol{W}$, for which $\nabla \mathcal{L}(\boldsymbol{W}) = 0$ which contradicts Lemma B.3. This implies that $\|\boldsymbol{W}\left(k\right)\| \to \infty$.

■

B.3.2 Global convergence under good initial gradient

To ensure global convergence, we identify an assumption that prevents GD from getting trapped at suboptimal tokens that offer no scoring advantage compared to other choices. To establish a foundation for providing the convergence of GD to the globally optimal solution $\boldsymbol{W}^{\mathrm{mm}}$, we present the following definitions. For parameters $\mu \in (0,1)$ and $R > 0$, consider the following subset of the sphere and its associated cone:

$$\bar{\mathcal{S}}_\mu(\boldsymbol{W}^{\mathrm{mm}}) := \left\{ \boldsymbol{W} \in \mathbb{R}^{d\times d} \;\middle|\; \left\langle (\boldsymbol{x}_{i\mathrm{opt}_i} - \boldsymbol{x}_{it}) \boldsymbol{z}_i^\top, \frac{\boldsymbol{W}}{\|\boldsymbol{W}\|_F} \right\rangle \geq \frac{\mu}{\|\boldsymbol{W}^{\mathrm{mm}}\|_F} \quad \forall t \neq \mathrm{opt}_i,\ i \in [n] \right\}, \tag{B.20a}$$

$$\bar{\mathcal{C}}_{\mu,R}(\boldsymbol{W}^{\mathrm{mm}}) := \left\{ \boldsymbol{W} \in \bar{\mathcal{S}}_\mu(\boldsymbol{W}^{\mathrm{mm}}) \;\middle|\; \|\boldsymbol{W}\|_F \geq R \right\}. \tag{B.20b}$$

Note that the $\bar{\mathcal{C}}_{\mu,R}(\boldsymbol{W}^{\mathrm{mm}})$ definition is equivalent to the $\bar{\mathcal{C}}_{\mu,R}$ definition in (3.8) with a change of variable $\mu \leftarrow \|\boldsymbol{W}^{\mathrm{mm}}\|_F \cdot \mu$.

Throughout this section, we will use the following notations:

$$\Theta = \frac{1}{\|\boldsymbol{W}^{\mathrm{mm}}\|_F}, \tag{B.21a}$$

$$A = \max\left\{\max_{i\in[n],t,\tau\in[T]} \frac{(\|\boldsymbol{x}_{it}\| \vee \|\boldsymbol{x}_{it} - \boldsymbol{x}_{i\tau}\|) \cdot \|\boldsymbol{z}_i\|}{\Theta}, 1\right\}. \tag{B.21b}$$

We will use the following notations: Recall scores $\gamma_{it} = y_i \cdot \boldsymbol{v}^\top \boldsymbol{x}_{it}$. We define the global scalar

$$\Gamma = \sup_{i\in[n],t,\tau\in[T]} |\gamma_{it} - \gamma_{i\tau}|. \tag{B.22a}$$

We also define the score gaps over $i \in \mathcal{I}$ and support indices:

$$\gamma_i^{\mathrm{gap}} = \gamma_{i\mathrm{opt}_i} - \max_{t\neq \mathrm{opt}_i} \gamma_{it}, \qquad \bar{\gamma}_i^{\mathrm{gap}} = \gamma_{i\mathrm{opt}_i} - \min_{t\neq \mathrm{opt}_i} \gamma_{it},$$
$$\gamma_{\min}^{\mathrm{gap}} = \min_{i\in[n]} \gamma_i^{\mathrm{gap}}, \quad \text{and} \quad \bar{\gamma}_{\max}^{\mathrm{gap}} = \max_{i\in[n]} \bar{\gamma}_i^{\mathrm{gap}}. \tag{B.22b}$$

Lemma B.4 *Suppose that Assumption 3.1 on the loss function ℓ holds. Let $opt = (opt_i)_{i=1}^n$ be the unique globally optimal indices (as defined in Definition 3.1), and let $\boldsymbol{W}^{mm}$ denote the solution to the optimization problem in* (W-SVM). *There exists a positive scalar μ such that for sufficiently large $\bar{R}_\mu$: For all $\boldsymbol{V} \in \bar{\mathcal{S}}_\mu(\boldsymbol{W}^{mm})$ with $\|\boldsymbol{V}\|_F = \|\boldsymbol{W}^{mm}\|_F$ and $\boldsymbol{W} \in \bar{\mathcal{C}}_{\mu,\bar{R}_\mu}(\boldsymbol{W}^{mm})$, there exist dataset dependent constants $C, \hat{C}, c > 0$ such that*

$$C \cdot \frac{1}{n}\sum_{i=1}^n \left(1 - \boldsymbol{s}_{iopt_i}\right) \geq -\left\langle \nabla\mathcal{L}(\boldsymbol{W}), \boldsymbol{V}\right\rangle \geq c \cdot \mu \cdot \frac{1}{n}\sum_{i=1}^n \left(1 - \boldsymbol{s}_{iopt_i}\right) > 0, \tag{B.23a}$$

$$\|\nabla\mathcal{L}(\boldsymbol{W})\|_F \leq \bar{A}\hat{C} \cdot \frac{1}{n}\sum_{i=1}^n \left(1 - \boldsymbol{s}_{iopt_i}\right), \tag{B.23b}$$

$$-\left\langle \frac{\boldsymbol{V}}{\|\boldsymbol{V}\|_F}, \frac{\nabla\mathcal{L}(\boldsymbol{W})}{\|\nabla\mathcal{L}(\boldsymbol{W})\|_F}\right\rangle \geq \frac{c}{\hat{C}} \cdot \frac{\Theta}{\bar{A}} > 0. \tag{B.23c}$$

Here, $\boldsymbol{s}_{iopt_i} = (\mathbb{S}(\boldsymbol{X}_i\boldsymbol{W}\boldsymbol{z}_i))_{opt_i}$, $\bar{A} = \max_{i\in[n],t,\tau\in[T]} \|\boldsymbol{x}_{it} - \boldsymbol{x}_{i\tau}\| \ \|\boldsymbol{z}_i\|$, and $\Theta = 1/\|\boldsymbol{W}^{mm}\|_F$.

Proof. Let $R = \|\boldsymbol{W}\|_F$,

$$\ell_i' = \ell'(y_i \cdot \boldsymbol{v}^\top \boldsymbol{X}_i^\top \mathbb{S}(\tilde{\boldsymbol{h}}_i)), \qquad \boldsymbol{h}_i = \boldsymbol{X}_i\boldsymbol{V}\boldsymbol{z}_i, \qquad \tilde{\boldsymbol{h}}_i = \boldsymbol{X}_i\boldsymbol{W}\boldsymbol{z}_i, \qquad \boldsymbol{s}_i = \mathbb{S}(\tilde{\boldsymbol{h}}_i). \tag{B.24}$$

The following inequalities hold for all $\boldsymbol{V} \in \bar{\mathcal{S}}_\mu$, $\|\boldsymbol{V}\|_F = \|\boldsymbol{W}^{\mathrm{mm}}\|_F$ and all $i \in [n], t \neq \mathrm{opt}_i$:

$$A \geq (\boldsymbol{x}_{i\mathrm{opt}_i} - \boldsymbol{x}_{it})^\top \boldsymbol{V}\boldsymbol{z}_i \geq \mu. \tag{B.25}$$

To proceed, we write the gradient correlation following (B.3) and (B.13)

$$\langle \nabla \mathcal{L}(\boldsymbol{W}), \boldsymbol{V} \rangle = \frac{1}{n} \sum_{i=1}^{n} \ell_i' \cdot \left\langle \boldsymbol{h}_i, \mathbb{S}'(\tilde{\boldsymbol{h}}_i)\gamma_i \right\rangle, \tag{B.26}$$

where we denoted $\ell_i' = \ell'(y_i \cdot \boldsymbol{v}^\top \boldsymbol{X}_i^\top \mathbb{S}(\tilde{\boldsymbol{h}}_i))$, $\boldsymbol{h}_i = \boldsymbol{X}_i \boldsymbol{V} \boldsymbol{z}_i$, $\tilde{\boldsymbol{h}}_i = \boldsymbol{X}_i \boldsymbol{W} \boldsymbol{z}_i$, $\boldsymbol{s}_i = \mathbb{S}(\tilde{\boldsymbol{h}}_i)$.

It follows from (B.21a) that

$$\begin{aligned} \max_{i\in[n],t\in[T]} \{|\boldsymbol{h}_{it}|\} = \max_{i\in[n],t\in[T]} \{|\boldsymbol{x}_{it}^\top \boldsymbol{V} \boldsymbol{z}_i|\} &= \max_{i\in[n],t\in[T]} \{|\langle \boldsymbol{x}_{it}\boldsymbol{z}_i^\top, \boldsymbol{V}\rangle|\} \\ &\le \max_{i\in[n],t\in[T]} \left\{ \|\boldsymbol{V}\|_F \, \|\boldsymbol{x}_{it}\boldsymbol{z}_i^\top\| \right\} \le A. \end{aligned} \tag{B.27}$$

Using (B.25), we can bound the softmax probabilities $\boldsymbol{s}_i = \mathbb{S}(\tilde{\boldsymbol{h}}_i)$ as follows, for all $i \in [n]$:

$$S_i := \sum_{\tau \neq \mathrm{opt}_i} \boldsymbol{s}_{i\tau} \le T e^{-R\mu\Theta} \boldsymbol{s}_{i\mathrm{opt}_i} \le T e^{-R\mu\Theta}. \tag{B.28}$$

Let us focus on a fixed datapoint $i \in [n]$, assume (without losing generality) $\mathrm{opt}_i = 1$, and drop subscripts i. Directly applying Lemma B.2, we obtain

$$\left| \boldsymbol{h}^\top \mathrm{diag}(\boldsymbol{s})\gamma - \boldsymbol{h}^\top \boldsymbol{s}\boldsymbol{s}^\top \gamma - \sum_{t\ge 2}^{T} (\boldsymbol{h}_1 - \boldsymbol{h}_t)\boldsymbol{s}_t(\gamma_1 - \gamma_t) \right| \le 2\Gamma A (1 - \boldsymbol{s}_1)^2.$$

To proceed, let us upper/lower bound the gradient correlation. Since $A \ge \boldsymbol{h}_1 - \boldsymbol{h}_t \ge \mu > 0$ from (B.25), setting $S := \sum_{t\neq \mathrm{opt}_i} \boldsymbol{s}_t = 1 - \boldsymbol{s}_1$, we find

$$A \cdot S \cdot \bar{\gamma}^{\mathrm{gap}} \ge \sum_{t \neq \mathrm{opt}} (\boldsymbol{h}_1 - \boldsymbol{h}_t)\boldsymbol{s}_t(\gamma_1 - \gamma_t) \ge \mu \cdot S \cdot \gamma^{\mathrm{gap}}. \tag{B.29}$$

Next we show that $S = 1 - \boldsymbol{s}_1$ dominates $(1 - \boldsymbol{s}_1)^2 = S^2$ for large R. Specifically, we wish for

$$\mu S \gamma^{\mathrm{gap}}/2 \ge 2\Gamma A (1 - \boldsymbol{s}_1)^2 \iff S \ge \frac{4}{\mu}\frac{\Gamma A}{\gamma^{\mathrm{gap}}} S^2 \iff S \le \frac{\mu\gamma^{\mathrm{gap}}}{4\Gamma A}. \tag{B.30}$$

Using (B.28), what we wish is ensured for all $i \in [n]$, by guaranteeing $T e^{-R\mu\Theta} \le \frac{\mu\gamma^{\mathrm{gap}}}{4\Gamma A}$. That is, by choosing

$$R \ge \frac{1}{\mu\Theta} \log\left(\frac{4T\Gamma A}{\mu \gamma_{\min}^{\mathrm{gap}}} \right), \tag{B.31}$$

where $\gamma_{\min}^{\text{gap}} = \min_{i\in[n]} \gamma_i^{\text{gap}}$ is the global scalar corresponding to the worst case score gap over all inputs.

With the above choice of R, we guaranteed

$$2A(1-\boldsymbol{s}_1)\cdot\bar{\gamma}^{\text{gap}} \geq 2A\cdot S\cdot\bar{\gamma}^{\text{gap}} \geq \sum_{t\neq\text{opt}} (\boldsymbol{h}_1-\boldsymbol{h}_t)\boldsymbol{s}_t(\gamma_1-\gamma_t) \geq \frac{\mu\cdot S\cdot\gamma^{\text{gap}}}{2} \geq \frac{\mu(1-\boldsymbol{s}_1)\gamma^{\text{gap}}}{2},$$

via (B.30) and (B.29).

Since this holds over all inputs, going back to the gradient correlation (B.26) and averaging above over all inputs $i\in[n]$ and plugging back the indices i, we obtain the advertised bound

$$\frac{2A}{n}\sum_{i\in[n]} -\ell_i'\cdot S_i\cdot\bar{\gamma}_i^{\text{gap}} \geq -\langle\nabla\mathcal{L}(\boldsymbol{W}),\boldsymbol{V}\rangle \geq \frac{\mu}{2n}\sum_{i\in[n]} -\ell_i'\cdot S_i\cdot\gamma_i^{\text{gap}}. \tag{B.32}$$

Let $-\ell'_{\min\,/\,\max}$ be the min/max values negative loss derivative admits over the ball $[-A,A]$ and note that $\max_{i\in[n]}\bar{\gamma}_i^{\text{gap}}>0$ and $\min_{i\in[n]}\gamma_i^{\text{gap}}>0$ are dataset dependent constants. Then, we declare the constants $C=-2A\ell'_{\max}\cdot\max_{i\in[n]}\bar{\gamma}_i^{\text{gap}}>0, c=-(1/2)\ell'_{\min}\cdot\min_{i\in[n]}\gamma_i^{\text{gap}}>0$ to obtain the bound (B.23a).

The proof of (B.23c) and (B.23b) follows similarly as the proof of Lemma B.7. ∎

The following lemma shows that as π approaches zero, the negative gradient of the loss function at $\boldsymbol{W}\in\bar{\mathcal{C}}_{\mu,R}(\boldsymbol{W}^{\text{mm}})$ becomes more correlated with the max-margin solution ($\boldsymbol{W}^{\text{mm}}$) than with $\boldsymbol{W}$ itself.

Lemma B.5 *Suppose Assumption 3.1 holds and let $opt=(opt_i)_{i=1}^n$ be the unique globally-optimal indices with $\boldsymbol{W}^{mm}$ denoting the W-SVM solution. For any choice of $\pi>0$, there exists $R_\pi\geq\bar{R}_\mu$ such that, for any $\boldsymbol{W}\in\bar{\mathcal{C}}_{\mu,R}(\boldsymbol{W}^{mm})$, we have*

$$\left\langle\nabla\mathcal{L}(\boldsymbol{W}),\frac{\boldsymbol{W}}{\|\boldsymbol{W}\|_F}\right\rangle \geq (1+\pi)\left\langle\nabla\mathcal{L}(\boldsymbol{W}),\frac{\boldsymbol{W}^{mm}}{\|\boldsymbol{W}^{mm}\|_F}\right\rangle.$$

Here, $\bar{\mathcal{C}}_{\mu,R}(\boldsymbol{W}^{mm})$ is the cone defined at (B.20).

Proof. Let

$$\begin{aligned}
&\bar{\boldsymbol{W}}=\|\boldsymbol{W}^{\text{mm}}\|_F\boldsymbol{W}/\|\boldsymbol{W}\|_F, \qquad \bar{\boldsymbol{h}}_i=\boldsymbol{X}_i\bar{\boldsymbol{W}}\boldsymbol{z}_i, \qquad \tilde{\boldsymbol{h}}_i=\boldsymbol{X}_i\boldsymbol{W}\boldsymbol{z}_i,\\
&\boldsymbol{h}_i=\boldsymbol{X}_i\boldsymbol{W}^{\text{mm}}\boldsymbol{z}_i, \qquad \boldsymbol{s}_i=\mathbb{S}(\tilde{\boldsymbol{h}}_i), \quad \text{and} \quad \ell_i'=\ell'(y_i\cdot\boldsymbol{v}^\top\boldsymbol{X}_i^\top\mathbb{S}(\tilde{\boldsymbol{h}}_i)).
\end{aligned} \tag{B.33}$$

To establish the result, we will prove that, for any $\pi>0$, there exists sufficiently large R_π

such that for any $\boldsymbol{W} \in \mathcal{C}_{\mu,R_\pi}(\boldsymbol{W}^{\mathrm{mm}})$:

$$\begin{aligned}\left\langle -\nabla\mathcal{L}(\boldsymbol{W}), \bar{\boldsymbol{W}} \right\rangle &= -\frac{1}{n}\sum_{i=1}^{n} \ell_i' \cdot \left\langle \bar{\boldsymbol{h}}_i, \mathbb{S}'(\tilde{\boldsymbol{h}}_i)\gamma_i \right\rangle \\ &\leq -\frac{1+\pi}{n}\sum_{i=1}^{n} \ell_i' \cdot \left\langle \boldsymbol{h}_i, \mathbb{S}'(\tilde{\boldsymbol{h}}_i)\gamma_i \right\rangle = (1+\pi)\left\langle -\nabla\mathcal{L}(\boldsymbol{W}), \boldsymbol{W}^{\mathrm{mm}} \right\rangle, \end{aligned} \tag{B.34}$$

or equivalently,

$$\sum_{i=1}^{n} -\ell_i' \cdot \left((1+\pi)\left(\boldsymbol{h}_i^\top \mathrm{diag}(\boldsymbol{s}_i)\gamma_i - \boldsymbol{h}_i^\top \boldsymbol{s}_i \boldsymbol{s}_i^\top \gamma_i\right) - \left(\bar{\boldsymbol{h}}_i^\top \mathrm{diag}(\boldsymbol{s}_i)\gamma_i - \bar{\boldsymbol{h}}_i \boldsymbol{s}_i \boldsymbol{s}_i^\top \gamma_i\right)\right) \geq 0. \tag{B.35}$$

It follows from Lemma B.2 that for $\boldsymbol{h}_i = \boldsymbol{X}_i \boldsymbol{W}^{\mathrm{mm}} \boldsymbol{z}_i$ and $\bar{\boldsymbol{h}}_i = \boldsymbol{X}_i \bar{\boldsymbol{W}} \boldsymbol{z}_i$, we have:

$$\left| \boldsymbol{h}_i^\top \mathrm{diag}(\boldsymbol{s}_i)\gamma_i - \boldsymbol{h}_i^\top \boldsymbol{s}_i \boldsymbol{s}_i^\top \gamma_i - \sum_{t \neq \mathrm{opt}_i} (\boldsymbol{h}_{i1} - \boldsymbol{h}_{it}) \boldsymbol{s}_{it} (\gamma_{i1} - \gamma_{it}) \right| \leq 2\Gamma A (1 - \boldsymbol{s}_{i1})^2, \tag{B.36a}$$

$$\left| \bar{\boldsymbol{h}}_i^\top \mathrm{diag}(\boldsymbol{s}_i)\gamma_i - \bar{\boldsymbol{h}}_i^\top \boldsymbol{s}_i \boldsymbol{s}_i^\top \gamma_i - \sum_{t \neq \mathrm{opt}_i} (\bar{\boldsymbol{h}}_{i1} - \bar{\boldsymbol{h}}_{it}) \boldsymbol{s}_{it} (\gamma_{i1} - \gamma_{it}) \right| \leq 2\Gamma A (1 - \boldsymbol{s}_{i1})^2. \tag{B.36b}$$

Using Lemma B.4 and its $\bar{R}_\mu$ constant, if $R_\pi > \bar{R}_\mu$, we just need to prove the statement for $0 < \pi < 1$. In this case, using (B.36), (B.35) is implied by the following stronger inequality

$$\sum_{i=1}^{n} (-\ell_i') \cdot \left((1+\pi) \sum_{t \neq \mathrm{opt}_i} (\boldsymbol{h}_{i1} - \boldsymbol{h}_{it}) \boldsymbol{s}_{it} (\gamma_{i1} - \gamma_{it}) - \sum_{t \neq \mathrm{opt}_i} (\bar{\boldsymbol{h}}_{i1} - \bar{\boldsymbol{h}}_{it}) \boldsymbol{s}_{it} (\gamma_{i1} - \gamma_{it}) - 6\Gamma A (1 - \boldsymbol{s}_{i1})^2 \right) \geq 0.$$

Recalling $\boldsymbol{h}_{i1} - \boldsymbol{h}_{it} \geq 1$, we note that $\sum_{t \neq \mathrm{opt}_i} \boldsymbol{s}_{it}(\gamma_{i1} - \gamma_{it}) \leq \sum_{t \neq \mathrm{opt}_i} (\boldsymbol{h}_{i1} - \boldsymbol{h}_{it}) \boldsymbol{s}_{it} (\gamma_{i1} - \gamma_{it})$. Now plugging in $\boldsymbol{h}$ in the bound above and assuming $\pi \leq 1$ (w.l.o.g.), (B.34) is implied by the following stronger inequality

$$\begin{aligned} &\sum_{i=1}^{n} (-\ell_i') \cdot \left(6\Gamma A (1 - \boldsymbol{s}_{i1})^2 + \sum_{t \neq \mathrm{opt}_i} (\boldsymbol{h}_{i1} - \boldsymbol{h}_{it}) \boldsymbol{s}_{it} (\gamma_{i1} - \gamma_{it}) \right) \\ &\leq (1+\pi) \sum_{i=1}^{n} (-\ell_i') \cdot \sum_{t \neq \mathrm{opt}_i} (\boldsymbol{h}_{i1} - \boldsymbol{h}_{it}) \boldsymbol{s}_{it} (\gamma_{i1} - \gamma_{it}) \\ &\leq (1+\pi) \sum_{i=1}^{n} (-\ell_i') \cdot \sum_{t \neq \mathrm{opt}_i} \boldsymbol{s}_{it} (\gamma_{i1} - \gamma_{it}). \end{aligned}$$

First, we claim that $0.5\pi \sum_{t \in \mathrm{opt}_i} \boldsymbol{s}_{it}(\gamma_{i1} - \gamma_{it}) \geq 6\Gamma A (1 - \boldsymbol{s}_{i1})^2$ for all $i \in [n]$. The proof of

this claim directly follows the argument in Lemma B.4, (namely following (B.28), (B.30), (B.31)) we have that $1 - \boldsymbol{s}_{i1} \le Te^{-R\mu\Theta}$ and $\gamma_{i1} - \gamma_{it} \ge \gamma_{\min}^{\text{gap}}$ for all $i \in [n]$. This leads to the choice

$$R \ge \frac{1}{\mu\Theta} \log\left(\frac{12 \cdot T\Gamma A}{\pi\gamma_{\min}^{\text{gap}}}\right). \tag{B.37}$$

Following this control over the perturbation term $6\Gamma A(1 - \boldsymbol{s}_{i1})^2$, to conclude with the result, what remains is proving the comparison

$$\sum_{i=1}^{n}(-\ell_i') \cdot \sum_{t\neq \text{opt}_i} (\boldsymbol{h}_{i1} - \boldsymbol{h}_{it})\boldsymbol{s}_{it}(\gamma_{i1} - \gamma_{it}) \le (1 + 0.5\pi)\sum_{i=1}^{n}(-\ell_i') \cdot \sum_{t\neq \text{opt}_i} \boldsymbol{s}_{it}(\gamma_{i1} - \gamma_{it}). \tag{B.38}$$

Scenario 1: $\|\bar{\boldsymbol{W}} - \boldsymbol{W}^{\text{mm}}\|_F \le \epsilon = \frac{\pi}{4A\Theta}$ for some $\epsilon > 0$. In this scenario, for any $t \neq \text{opt}_i$ and $i \in [n]$, we have

$$|\boldsymbol{h}_{it} - \boldsymbol{h}_{it}| = |\boldsymbol{x}_{it}^\top(\bar{\boldsymbol{W}} - \boldsymbol{W}^{\text{mm}})\boldsymbol{z}_i| \le A\Theta\epsilon = \frac{\pi}{4}.$$

Consequently, we obtain

$$\boldsymbol{h}_{i1} - \boldsymbol{h}_{it} \le \boldsymbol{h}_{i1} - \boldsymbol{h}_{it} + 2A\Theta\epsilon = 1 + 0.5\pi.$$

Similarly, $\boldsymbol{h}_{i1} - \boldsymbol{h}_{it} \ge 1 - 0.5\pi \ge 0.5$. Since all terms $\boldsymbol{h}_{i1} - \boldsymbol{h}_{it}, \boldsymbol{s}_{it}, \gamma_{i1} - \gamma_{it}$ in (B.38) are nonnegative, we obtain (B.38).

Scenario 2: $\|\bar{\boldsymbol{W}} - \boldsymbol{W}^{\text{mm}}\|_F \ge \epsilon = \frac{\pi}{4A\Theta}$. Since $\bar{\boldsymbol{W}}$ is not max-margin solution, in this scenario, for some $i \in [n]$, $\nu = \nu(\epsilon) > 0$, and $\tau \neq \text{opt}_i$, we have that

$$\boldsymbol{h}_{i1} - \boldsymbol{h}_{i\tau} \le 1 - 2\nu.$$

Here $\tau = \arg\max_{\tau\neq\text{opt}_i} \boldsymbol{x}_{i\tau}\bar{\boldsymbol{W}}\boldsymbol{z}_i$ denotes the nearest point to $\boldsymbol{h}_{i1}$ (along the $\bar{\boldsymbol{W}}$ direction). Recall that $\boldsymbol{s} = \mathbb{S}(\bar{R}\boldsymbol{h})$, where $\bar{R} = R\Theta = \|\boldsymbol{W}\|_F/\|\boldsymbol{W}^{\text{mm}}\|_F$. To proceed, let $\underline{\boldsymbol{h}}_i := \min_{t\neq\text{opt}_i} \boldsymbol{h}_{i1} - \boldsymbol{h}_{it}$,

$$\mathcal{I} := \{i \in [n] : \underline{\boldsymbol{h}}_i \le 1 - 2\nu\}, \qquad [n] - \mathcal{I} := \{i \in [n] : 1 - 2\nu < \underline{\boldsymbol{h}}_i\}.$$

For all $i \in [n] - \mathcal{I}$,

$$\begin{aligned}\sum_{t \neq \mathrm{opt}_i} (\boldsymbol{h}_{i1} - \boldsymbol{h}_{it}) \boldsymbol{s}_{it} (\gamma_{i1} - \gamma_{it}) - (1 + 0.5\pi) \sum_{t \neq \mathrm{opt}_i} \boldsymbol{s}_{it} (\gamma_{i1} - \gamma_{it}) \\ &\leq (2A - (1 + 0.5\pi)) \, \Gamma \sum_{t \neq \mathrm{opt}_i, \ \boldsymbol{h}_{i1} - \boldsymbol{h}_{it} \geq 1 + \frac{\pi}{2}} \boldsymbol{s}_{it} \\ &\leq (2A - (1 + 0.5\pi)) \, \Gamma T e^{-\bar{R}(1 + \frac{\pi}{2})} \\ &\leq 2A\Gamma T e^{-\bar{R}(1 + \frac{\pi}{2})}.\end{aligned} \tag{B.39}$$

For all $i \in \mathcal{I}$, split the tokens into two groups: Let $\mathcal{N}_i$ be the group of tokens obeying $\boldsymbol{h}_{i1} - \boldsymbol{h}_{it} \leq 1 - \nu$ and $\bar{\mathcal{N}}_i := [T] - \{\mathrm{opt}_i\} - \mathcal{N}_i$ be the rest of the neighbors. Observe that

$$\frac{\sum_{t \in \bar{\mathcal{N}}_i} \boldsymbol{s}_{it}}{\sum_{t \neq \mathrm{opt}_i} \boldsymbol{s}_{it}} \leq T \frac{e^{\nu \bar{R}}}{e^{2\nu \bar{R}}} = T e^{-\bar{R}\nu}.$$

Using $|\boldsymbol{h}_{i1} - \boldsymbol{h}_{it}| \leq 2A$ and $\gamma_{\min}^{\mathrm{gap}} = \min_{i \in [n]} \gamma_i^{\mathrm{gap}} = \min_{i \in [n]} (\gamma_{i1} - \max_{t \neq \mathrm{opt}_i} \gamma_{it})$, observe that

$$\sum_{t \in \bar{\mathcal{N}}_i} (\boldsymbol{h}_{i1} - \boldsymbol{h}_{it}) \boldsymbol{s}_{it} (\gamma_{i1} - \gamma_{it}) \leq \frac{2\Gamma A T e^{-\bar{R}\nu}}{\gamma_{\min}^{\mathrm{gap}}} \sum_{t \neq \mathrm{opt}_i} \boldsymbol{s}_{it} (\gamma_{i1} - \gamma_{it}).$$

Thus,

$$\begin{aligned}\sum_{t \neq \mathrm{opt}_i} (\boldsymbol{h}_{i1} - \boldsymbol{h}_{it}) \boldsymbol{s}_{it} (\gamma_{i1} - \gamma_{it}) &= \sum_{t \in \mathcal{N}_i} (\boldsymbol{h}_{i1} - \boldsymbol{h}_{it}) \boldsymbol{s}_{it} (\gamma_{i1} - \gamma_{it}) + \sum_{t \in \bar{\mathcal{N}}_i} (\boldsymbol{h}_{i1} - \boldsymbol{h}_{it}) \boldsymbol{s}_{it} (\gamma_{i1} - \gamma_{it}) \\ &\leq \sum_{t \in \mathcal{N}_i} (1 - \nu) \boldsymbol{s}_{it} (\gamma_{i1} - \gamma_{it}) + \frac{2\Gamma A T e^{-\bar{R}\nu}}{\gamma_{\min}^{\mathrm{gap}}} \sum_{t \neq \mathrm{opt}_i} \boldsymbol{s}_{it} (\gamma_{i1} - \gamma_{it}) \\ &\leq \left(1 - \nu + \frac{2\Gamma A T e^{-\bar{R}\nu}}{\gamma_{\min}^{\mathrm{gap}}}\right) \sum_{t \neq \mathrm{opt}_i} \boldsymbol{s}_{it} (\gamma_{i1} - \gamma_{it}) \\ &\leq \left(1 + \frac{2\Gamma A T e^{-\bar{R}\nu}}{\gamma_{\min}^{\mathrm{gap}}}\right) \sum_{t \neq \mathrm{opt}_i} \boldsymbol{s}_{it} (\gamma_{i1} - \gamma_{it}).\end{aligned}$$

Hence, choosing

$$R \geq \frac{1}{\nu \Theta} \log \left(\frac{8\Gamma A T}{\gamma_{\min}^{\mathrm{gap}} \pi}\right) \tag{B.40}$$

results in that

$$\begin{aligned}
&\sum_{t\neq \mathrm{opt}_i} (\boldsymbol{h}_{i1}-\boldsymbol{h}_{it})\boldsymbol{s}_{it}(\gamma_{i1}-\gamma_{it}) - \left(1+\frac{\pi}{2}\right)\sum_{t\neq \mathrm{opt}_i} \boldsymbol{s}_{it}(\gamma_{i1}-\gamma_{it}) \\
&\leq \left(\frac{2\Gamma ATe^{-\bar{R}\nu}}{\gamma_{\min}^{\mathrm{gap}}} - \frac{\pi}{2}\right)\sum_{t\neq \mathrm{opt}_i} \boldsymbol{s}_{it}(\gamma_{i1}-\gamma_{it}) \\
&\leq -\frac{\pi}{4}\sum_{t\neq \mathrm{opt}_i} \boldsymbol{s}_{it}(\gamma_{i1}-\gamma_{it}) \\
&\leq -\frac{\pi}{4T}\gamma_{\min}^{\mathrm{gap}} e^{-\bar{R}(1-2\nu)}.
\end{aligned} \tag{B.41}$$

Here, the last inequality follows from the fact that

$$\sum_{t\neq \mathrm{opt}_i} \boldsymbol{s}_{it} \geq \max_{t\neq \mathrm{opt}_i} \boldsymbol{s}_{it} \geq \frac{e^{-\bar{R}(1-2\nu)}}{\sum_{t=1}^{T} e^{-\bar{R}(\boldsymbol{h}_{i1}-\boldsymbol{h}_{it})}} \geq \frac{1}{T}e^{-\bar{R}(1-2\nu)}.$$

From Assumption 3.1, we have $c_{\min} \leq -\ell' \leq c_{\max}$ for some positive constants $c_{\min}$ and $c_{\max}$. It follows from (B.39) and (B.41) that

$$\begin{aligned}
&-\frac{1}{n}\sum_{i}^{n} \ell_i' \cdot \left(\sum_{t\neq \mathrm{opt}_i} (\boldsymbol{h}_{i1}-\boldsymbol{h}_{it})\boldsymbol{s}_{it}(\gamma_{i1}-\gamma_{it}) - \sum_{t\neq \mathrm{opt}_i} (1+0.5\pi)\boldsymbol{s}_{it}(\gamma_{i1}-\gamma_{it})\right) \\
&\qquad \leq c_{\max} 2A\Gamma T\Gamma e^{-\bar{R}(1+\frac{\pi}{2})} - \frac{c_{\min}}{nT}\cdot\frac{\pi\gamma_{\min}^{\mathrm{gap}}}{4}e^{-\bar{R}(1-2\nu)} \\
&\qquad \leq 0.
\end{aligned}$$

Combing with (B.40), this is guaranteed by choosing

$$R \geq R_\pi = \max\left\{\frac{1}{\nu\Theta}\log\left(\frac{C_0\Gamma AT}{\gamma_{\min}^{\mathrm{gap}}\pi}\right), \frac{1}{(2\nu+\pi/2)\Theta}\log\left(\frac{C_0 n\Gamma AT^2 c_{\max}}{c_{\min}\gamma_{\min}^{\mathrm{gap}}\pi}\right)\right\},$$

where $\nu = \nu(\frac{\pi}{4A\Theta})$ depends only on π and global problem variables. Combining this with the prior R choice (B.37) (by taking the maximum), we shall choose C_0 sufficiently large such that $R_\pi \geq \bar{R}_\mu$, where $\bar{R}_\mu$ is defined in Lemma B.4. ∎

B.3.2.1 Proof of Theorem 3.4

The following theorem is a restatement of Theorem 3.4. Two minor differences are: (1) We state with the change of variable $\mu \leftarrow \|\boldsymbol{W}^{\mathrm{mm}}\|_F \cdot \mu$; see discussions below (B.20). (2) We also include (**L3.**) in the statement of the theorem.

Theorem B.1 *Suppose Assumption 3.1 on the loss function ℓ and Assumption 3.3 on the initial gradient hold.*

L1. *For any $\mu > 0$, there exists $R > 0$ such that $\bar{\mathcal{C}}_{\mu,R}(\boldsymbol{W}^{mm})$ defined in* (B.20) *does not contain any stationary points.*

L2. *Fix any $\mu \in (0, \min(1, \iota\|\boldsymbol{W}^{mm}\|_F/\|\nabla\mathcal{L}(0)\|_F)$. Consider GD iterations with $\boldsymbol{W}(0) = 0$, $\boldsymbol{W}(1) = -R\nabla\mathcal{L}(0)/\|\nabla\mathcal{L}(0)\|_F$, and $\boldsymbol{W}(k+1) = \boldsymbol{W}(k) - \eta\nabla\mathcal{L}(\boldsymbol{W}(k))$ for $\eta \le 1/L_{\boldsymbol{W}}$, $k \ge 1$, and R sufficiently large. If all iterates remain within $\bar{\mathcal{C}}_{\mu,R}$, then $\lim_{k\to\infty}\|\boldsymbol{W}(k)\|_F = \infty$ and $\lim_{k\to\infty}\frac{\boldsymbol{W}(k)}{\|\boldsymbol{W}(k)\|_F} = \frac{\boldsymbol{W}^{mm}}{\|\boldsymbol{W}^{mm}\|_F}$.*

L3. *Assume $\eta \le 1/L_{\boldsymbol{W}}$ and for all $\boldsymbol{W} \in \bar{\mathcal{C}}_{\mu,R}(\boldsymbol{W}^{mm})$ with sufficiently large R*

$$\min_{i\in[n]}\left\langle(\boldsymbol{x}_{iopt_i} - \boldsymbol{x}_{it})\boldsymbol{z}_i^\top, \boldsymbol{W} - \eta\nabla\mathcal{L}(\boldsymbol{W})\right\rangle \ge \min_{i\in[n]}\left\langle(\boldsymbol{x}_{iopt_i} - \boldsymbol{x}_{it})\boldsymbol{z}_i^\top, \boldsymbol{W}\right\rangle - \frac{2\eta\mu}{\|\boldsymbol{W}^{mm}\|_F^2}\left\langle\nabla\mathcal{L}(\boldsymbol{W}), \boldsymbol{W}^{mm}\right\rangle, \tag{B.42}$$

then all GD iterations remain within $\bar{\mathcal{C}}_{\mu,R}(\boldsymbol{W}^{mm})$.

Proof. Note that **L1.** is a direct corollary of Lemma B.4. We proceed with the proof of **L2.** and **L3.**. We provide the proof in four steps:

Step 1: $\bar{\mathcal{C}}_{\mu,R^0_\mu}(\boldsymbol{W}^{\mathbf{mm}})$ construction. Let us denote the initialization lower bound as $R^0_\mu := R$, where R is given in the Theorem B.1's statement. Consider an arbitrary value of $\epsilon \in (0, \mu/2)$ and let $1/(1+\pi) = 1 - \epsilon$. We additionally denote $R_\epsilon \leftarrow R_\pi \vee 1/2$ where R_π was defined in Lemma B.5. At initialization $\boldsymbol{W}(0)$, we set $\epsilon = \mu/2$ to obtain $R^0_\mu = R_{\mu/2}$.

We proceed to show $\mu \in (0, \min(1, \iota\|\boldsymbol{W}^{\mathrm{mm}}\|_F/\|\nabla\mathcal{L}(0)\|_F)$. It follows from Assumption 3.3 and under zero initialization for GD ($\boldsymbol{W}(0) = 0$) that

$$\left\langle(\boldsymbol{x}_{i\mathrm{opt}_i} - \boldsymbol{x}_{it})\boldsymbol{z}_i^\top, -\nabla\mathcal{L}(\boldsymbol{W}(0))\right\rangle = \left\langle(\boldsymbol{x}_{i\mathrm{opt}_i} - \boldsymbol{x}_{it})\boldsymbol{z}_i^\top, -\nabla\mathcal{L}(0)\right\rangle \ge \iota > 0,$$

for some positive constant ι. Hence, for any initial step size $\eta(0) > 0$ and $\boldsymbol{W}(1) = -\eta(0)\nabla\mathcal{L}(0)$,

$$\begin{aligned}\left\langle(\boldsymbol{x}_{i\mathrm{opt}_i} - \boldsymbol{x}_{it})\boldsymbol{z}_i^\top, \frac{\boldsymbol{W}(1)}{\|\boldsymbol{W}(1)\|_F}\right\rangle &= \frac{\eta(0)}{\|\boldsymbol{W}(1)\|_F}\left\langle(\boldsymbol{x}_{i\mathrm{opt}_i} - \boldsymbol{x}_{it})\boldsymbol{z}_i^\top, -\nabla\mathcal{L}(0)\right\rangle \\ &\ge \frac{\iota\eta(0)}{\|\boldsymbol{W}(1)\|_F} = \frac{\iota}{\|\nabla\mathcal{L}(0)\|_F} \\ &\ge \frac{\mu}{\|\boldsymbol{W}^{\mathrm{mm}}\|_F}.\end{aligned} \tag{B.43}$$

Here, the last inequality follows from our choice of μ in the theorem statement, i.e.

$$\mu \in \left(0, \min\left(1, \frac{\iota \|\boldsymbol{W}^{\mathrm{mm}}\|_F}{\|\nabla \mathcal{L}(0)\|_F}\right)\right). \tag{B.44}$$

This μ choice induces the conic set $\bar{\mathcal{C}}_{\mu,R_\mu^0}(\boldsymbol{W}^{\mathrm{mm}})$ with $R_\mu^0 = R_{\mu/2}$, where $R_{\mu/2}$ was defined in Lemma B.5. Now, given the parameter μ satisfying (B.44), we can choose $\eta(0)$ such that $\|\boldsymbol{W}(1)\|_F \geq R_\mu^0$ and $\boldsymbol{W}(1) \in \bar{\mathcal{C}}_{\mu,R_\mu^0}(\boldsymbol{W}^{\mathrm{mm}})$. To achieve this, since $\boldsymbol{W}(0) = 0$, we obtain

$$\eta(0) = \frac{R_\mu^0}{\|\nabla \mathcal{L}(0)\|_F}. \tag{B.45}$$

Since by our definition, $R_\mu^0 \leftarrow R$, (B.45) gives $\boldsymbol{W}(1)$ in the theorem's statement.

Step 2: There are no stationary points within $\bar{\mathcal{C}}_{\mu,R_\mu^0}(\boldsymbol{W}^{\mathrm{mm}})$. This step follows from **L1.**. Specifically, we can apply Lemma B.4 to find that: For all $\boldsymbol{V}, \boldsymbol{W} \in \bar{\mathcal{S}}_\mu(\boldsymbol{W}^{\mathrm{mm}})$ with $\|\boldsymbol{W}\|_F \neq 0$ and $\|\boldsymbol{W}\|_F \geq R_\mu^0$, we have that $-\langle \boldsymbol{V}, \nabla \mathcal{L}(\boldsymbol{W})\rangle$ is strictly positive.

Gradient correlation holds for large parameter norm. It follows from Lemma B.5 that, there exists $R_\epsilon \geq \bar{R}_\mu \vee 1/2$ such that all $\boldsymbol{W} \in \bar{\mathcal{C}}_{\mu,R_\epsilon}(\boldsymbol{W}^{\mathrm{mm}})$ satisfy

$$\left\langle -\nabla \mathcal{L}(\boldsymbol{W}), \frac{\boldsymbol{W}^{\mathrm{mm}}}{\|\boldsymbol{W}^{\mathrm{mm}}\|_F} \right\rangle \geq (1-\epsilon) \left\langle -\nabla \mathcal{L}(\boldsymbol{W}), \frac{\boldsymbol{W}}{\|\boldsymbol{W}\|_F} \right\rangle. \tag{B.46}$$

The following argument applies to a general $\epsilon \in (0, \mu/2)$. However, at initialization $\boldsymbol{W}(0) = 0$, we have set $\epsilon = \mu/2$ and defined the initialization radius as $R_\mu^0 = R_{\mu/2}$. To proceed, we will prove the main statements (**L2.**) and (**L3.**) as follows.

- Proving **L3.**: In **Step 3**, we will assume Condition (B.42) to prove that gradient iterates remain within $\bar{\mathcal{C}}_{\mu,R_\epsilon}(\boldsymbol{W}^{\mathrm{mm}})$. Concretely, for any $\epsilon \in (0, \mu/2)$, we will show that after gradient descent enters the conic set $\bar{\mathcal{C}}_{\mu,R_\epsilon}(\boldsymbol{W}^{\mathrm{mm}})$ for the first time, it will never leave the set under Condition (B.42) of the theorem statement and (B.46). In what follows, let us denote k_ϵ to be the first time gradient descent enters $\bar{\mathcal{C}}_{\mu,R_\epsilon}(\boldsymbol{W}^{\mathrm{mm}})$. Note that for $\epsilon \leftarrow \mu/2$, $k_\epsilon = 0$ i.e. the point of initialization.
- Proving **L2.**: In **Step 4**, assuming iterates within $\bar{\mathcal{C}}_{\mu,R_\epsilon}(\boldsymbol{W}^{\mathrm{mm}})$, we will prove that the norm diverges (as a result such k_ϵ is guaranteed to exist) and, additionally, the gradient updates asymptotically aligns with $\boldsymbol{W}^{\mathrm{mm}}$.

Step 3 (Proof of L3.): Updates remain inside the cone $\bar{\mathcal{C}}_{\mu,R_\epsilon}(\boldsymbol{W}^{\mathrm{mm}})$. Note that if $\boldsymbol{W}(k) \in \bar{\mathcal{C}}_{\mu,R_\epsilon}(\boldsymbol{W}^{\mathrm{mm}})$ for all $k \geq 1$, the required condition in **L2.** holds, and we proceed to **Step 4**. In this step, we show **L3.**. Specifically, we show that under Condition (B.42) and using (B.46), all iterates $\boldsymbol{W}(k) \in \bar{\mathcal{C}}_{\mu,R_\epsilon}(\boldsymbol{W}^{\mathrm{mm}})$ remain within $\bar{\mathcal{C}}_{\mu,R_\epsilon}(\boldsymbol{W}^{\mathrm{mm}})$.

To proceed, by leveraging the results from **Step 1** and **Step 2**, we demonstrate that the gradient iterates, with an appropriate constant step size, starting from $\boldsymbol{W}(k_\epsilon) \in \bar{\mathcal{C}}_{\mu,R_\epsilon}(\boldsymbol{W}^{\mathrm{mm}})$, remain within this set. We proceed by induction. Suppose that the claim holds up to iteration $k \geq k_\epsilon$. This implies that $\boldsymbol{W}(k) \in \bar{\mathcal{C}}_{\mu,R_\epsilon}(\boldsymbol{W}^{\mathrm{mm}})$. Hence, recalling $\bar{\mathcal{C}}_{\mu,R_\epsilon}(\boldsymbol{W}^{\mathrm{mm}})$ defined in (B.20), there exists scalar $\mu = \mu(\boldsymbol{\alpha}) \in (0,1)$ and R_ϵ such that $\|\boldsymbol{W}(k)\|_F \geq R_\epsilon$, and

$$\left\langle (\boldsymbol{x}_{i\mathrm{opt}_i} - \boldsymbol{x}_{it})\boldsymbol{z}_i^\top, \frac{\boldsymbol{W}(k)}{\|\boldsymbol{W}(k)\|_F} \right\rangle \geq \mu\Theta,$$

where $\Theta = 1/\|\boldsymbol{W}^{\mathrm{mm}}\|_F$.

Let

$$\frac{1}{1-\epsilon}\left\langle \frac{\boldsymbol{W}^{\mathrm{mm}}}{\|\boldsymbol{W}^{\mathrm{mm}}\|_F}, -\nabla\mathcal{L}(\boldsymbol{W}(k)) \right\rangle =: \rho(k) > 0. \tag{B.47a}$$

Using (B.42), we have

$$\begin{aligned}\left\langle (\boldsymbol{x}_{i\mathrm{opt}_i} - \boldsymbol{x}_{it})\boldsymbol{z}_i^\top, \frac{\boldsymbol{W}(k+1)}{\|\boldsymbol{W}(k)\|_F} \right\rangle &= \left\langle (\boldsymbol{x}_{i\mathrm{opt}_i} - \boldsymbol{x}_{it})\boldsymbol{z}_i^\top, \frac{\boldsymbol{W}(k)}{\|\boldsymbol{W}(k)\|_F} - \frac{\eta}{\|\boldsymbol{W}(k)\|_F}\nabla\mathcal{L}(\boldsymbol{W}(k)) \right\rangle \\ &\geq \mu\Theta + \frac{2\eta(1-\epsilon)\mu\Theta\rho(k)}{\|\boldsymbol{W}(k)\|_F}.\end{aligned} \tag{B.48}$$

From Lemma B.4, we have $\langle \nabla\mathcal{L}(\boldsymbol{W}(k)), \boldsymbol{W}(k)\rangle < 0$ which implies that $\|\boldsymbol{W}(k+1)\|_F \geq \|\boldsymbol{W}(k)\|_F$. This together with R_ϵ definition and $\|\boldsymbol{W}(k)\|_F \geq 1/2$ implies that

$$\begin{aligned}\|\boldsymbol{W}(k+1)\|_F &\leq \frac{1}{2\|\boldsymbol{W}(k)\|_F}\left(\|\boldsymbol{W}(k+1)\|_F^2 + \|\boldsymbol{W}(k)\|_F^2\right) \\ &= \frac{1}{2\|\boldsymbol{W}(k)\|_F}\left(2\|\boldsymbol{W}(k)\|_F^2 - 2\eta\left\langle \nabla\mathcal{L}(\boldsymbol{W}(k)), \boldsymbol{W}(k)\right\rangle + \eta^2\|\nabla\mathcal{L}(\boldsymbol{W}(k))\|_F^2\right) \\ &\leq \|\boldsymbol{W}(k)\|_F - \frac{\eta}{\|\boldsymbol{W}(k)\|_F}\left\langle \nabla\mathcal{L}(\boldsymbol{W}(k)), \boldsymbol{W}(k)\right\rangle + \eta^2\|\nabla\mathcal{L}(\boldsymbol{W}(k))\|_F^2.\end{aligned}$$

Thus,

$$\begin{aligned}\frac{\|\boldsymbol{W}(k+1)\|_F}{\|\boldsymbol{W}(k)\|_F} &\leq 1 - \frac{\eta}{\|\boldsymbol{W}(k)\|_F}\left\langle \nabla\mathcal{L}(\boldsymbol{W}(k)), \frac{\boldsymbol{W}(k)}{\|\boldsymbol{W}(k)\|_F}\right\rangle + \eta^2\frac{\|\nabla\mathcal{L}(\boldsymbol{W}(k))\|_F^2}{\|\boldsymbol{W}(k)\|_F} \\ &\leq 1 - \frac{\eta}{(1-\epsilon)\|\boldsymbol{W}(k)\|_F}\left\langle \nabla\mathcal{L}(\boldsymbol{W}(k)), \frac{\boldsymbol{W}^{\mathrm{mm}}}{\|\boldsymbol{W}^{\mathrm{mm}}\|_F}\right\rangle + \eta^2\frac{\|\nabla\mathcal{L}(\boldsymbol{W}(k))\|_F^2}{\|\boldsymbol{W}(k)\|_F} \\ &\leq 1 + \frac{\eta\rho(k)}{\|\boldsymbol{W}(k)\|_F} + \frac{\eta^2\|\nabla\mathcal{L}(\boldsymbol{W}(k))\|_F^2}{\|\boldsymbol{W}(k)\|_F} =: C_1(\rho(k),\eta).\end{aligned} \tag{B.49}$$

Here, the second inequality uses (B.46).

Now, it follows from (B.48) and (B.49) that

$$\begin{aligned}
\min_{t\neq \mathrm{opt}_i,\ i\in[n]} \ & \left\langle (\boldsymbol{x}_{i\mathrm{opt}_i} - \boldsymbol{x}_{it})\boldsymbol{z}_i^\top, \frac{\boldsymbol{W}(k+1)}{\|\boldsymbol{W}(k+1)\|_F} \right\rangle \\
& \geq \frac{1}{C_1(\rho(k),\eta)} \left(\mu\Theta + \frac{2\eta(1-\epsilon)\mu\Theta\rho(k)}{\|\boldsymbol{W}(k)\|_F} \right) \\
& = \mu\Theta + \frac{\eta\mu\Theta}{C_1(\rho(k),\eta)} \left(\frac{\big(2(1-\epsilon)-1\big)\rho(k)}{\|\boldsymbol{W}(k)\|_F} - \eta\frac{\|\nabla\mathcal{L}(\boldsymbol{W}(k))\|_F^2}{\|\boldsymbol{W}(k)\|_F} \right) \\
& = \mu\Theta + \frac{\eta\mu\Theta}{C_1(\rho(k),\eta)} \left(\frac{(1-2\epsilon)\rho(k)}{\|\boldsymbol{W}(k)\|_F} - \eta\frac{\|\nabla\mathcal{L}(\boldsymbol{W}(k))\|_F^2}{\|\boldsymbol{W}(k)\|_F} \right) \\
& \geq \mu\Theta,
\end{aligned} \tag{B.50}$$

where the last inequality uses our choice of stepsize $\eta \leq 1/L_W$ in Theorem 3.4's statement. Specifically, we need η to be small to ensure the last inequality. This can be obtained by following similar steps as in the proof of Theorem 3.5. We will guarantee this by choosing a proper R_ϵ in Lemma B.5. Hence, (B.50) implies (B.50) and $\boldsymbol{W}(k+1) \in \bar{\mathcal{C}}_{\mu,R_\epsilon}(\boldsymbol{W}^{\mathrm{mm}})$.

Step 4 (Proof of L2.): $\boldsymbol{W}(k)$ and $\boldsymbol{W}^{\mathrm{mm}}$ perfectly align over time. By theorem statement (alternatively via **Step 3**), we have that all iterates remain within the initial conic set i.e. $\boldsymbol{W}(k) \in \bar{\mathcal{C}}_{\mu,R_\mu^0}(\boldsymbol{W}^{\mathrm{mm}})$ for all $k \geq 0$. Note that it follows from Lemma B.4 that $\langle \nabla\mathcal{L}(\boldsymbol{W}), \boldsymbol{W}^{\mathrm{mm}}/\|\boldsymbol{W}^{\mathrm{mm}}\|_F \rangle < 0$, for any finite $\boldsymbol{W} \in \bar{\mathcal{C}}_{\mu,R_\mu^0}(\boldsymbol{W}^{\mathrm{mm}})$. Hence, there are no finite critical points $\boldsymbol{W} \in \bar{\mathcal{C}}_{\mu,R_\mu^0}(\boldsymbol{W}^{\mathrm{mm}})$, for which $\nabla\mathcal{L}(\boldsymbol{W}) = 0$. Now, based on Lemma B.3, which guarantees that $\nabla\mathcal{L}(\boldsymbol{W}(k)) \to 0$, this implies that $\|\boldsymbol{W}(k)\| \to \infty$. Consequently, for any choice of $\epsilon \in (0, \mu/2)$ there is an iteration k_ϵ such that, for all $k \geq k_\epsilon$, $\boldsymbol{W}(k) \in \bar{\mathcal{C}}_{\mu,R_\epsilon}(\boldsymbol{W}^{\mathrm{mm}})$. Once within $\bar{\mathcal{C}}_{\mu,R_\epsilon}(\boldsymbol{W}^{\mathrm{mm}})$, multiplying both sides of (B.46) by the stepsize η and using the gradient descent update, we get

$$\begin{aligned}
& \left\langle \boldsymbol{W}(k+1) - \boldsymbol{W}(k), \frac{\boldsymbol{W}^{\mathrm{mm}}}{\|\boldsymbol{W}^{\mathrm{mm}}\|_F} \right\rangle \\
& \geq (1-\epsilon) \left\langle \boldsymbol{W}(k+1) - \boldsymbol{W}(k), \frac{\boldsymbol{W}(k)}{\|\boldsymbol{W}(k)\|_F} \right\rangle \\
& = \frac{(1-\epsilon)}{2\|\boldsymbol{W}(k)\|_F} \left(\|\boldsymbol{W}(k+1)\|_F^2 - \|\boldsymbol{W}(k)\|_F^2 - \|\boldsymbol{W}(k+1) - \boldsymbol{W}(k)\|_F^2 \right) \\
& \geq (1-\epsilon) \left(\frac{1}{2\|\boldsymbol{W}(k)\|_F} \left(\|\boldsymbol{W}(k+1)\|_F^2 - \|\boldsymbol{W}(k)\|_F^2 \right) - \|\boldsymbol{W}(k+1) - \boldsymbol{W}(k)\|_F^2 \right) \\
& \geq (1-\epsilon) \left(\|\boldsymbol{W}(k+1)\|_F - \|\boldsymbol{W}(k)\|_F - \|\boldsymbol{W}(k+1) - \boldsymbol{W}(k)\|_F^2 \right) \\
& \geq (1-\epsilon) \Big(\|\boldsymbol{W}(k+1)\|_F - \|\boldsymbol{W}(k)\|_F - 2\eta \left(\mathcal{L}(\boldsymbol{W}(k)) - \mathcal{L}(\boldsymbol{W}(k+1)) \right) \Big).
\end{aligned}$$

Here, the second inequality is obtained from $\|\boldsymbol{W}(k)\|_F \geq 1/2$; the third inequality follows since for any $a, b > 0$, we have $(a^2 - b^2)/(2b) - (a - b) \geq 0$; and the last inequality uses Lemma B.3.

Summing the above inequality over $k \geq k_\epsilon$ gives

$$\left\langle \frac{\boldsymbol{W}(k)}{\|\boldsymbol{W}(k)\|_F}, \frac{\boldsymbol{W}^{\mathrm{mm}}}{\|\boldsymbol{W}^{\mathrm{mm}}\|_F} \right\rangle \geq 1 - \epsilon + \frac{C(\epsilon, \eta)}{\|\boldsymbol{W}(k)\|_F}, \qquad \boldsymbol{W}(k) \in \bar{\mathcal{C}}_{\mu, R_\epsilon}(\boldsymbol{W}^{\mathrm{mm}}),$$

where $\mathcal{L}_\star \leq \mathcal{L}(\boldsymbol{W}(k))$ for all $k \geq k_\epsilon$, and

$$C(\epsilon, \eta) = \left\langle \boldsymbol{W}(k_\epsilon), \frac{\boldsymbol{W}^{\mathrm{mm}}}{\|\boldsymbol{W}^{\mathrm{mm}}\|_F} \right\rangle - (1 - \epsilon)\|\boldsymbol{W}(k_\epsilon)\|_F - 2\eta(1 - \epsilon)(\mathcal{L}(\boldsymbol{W}(k_\epsilon)) - \mathcal{L}_\star).$$

Consequently,

$$\liminf_{k \to \infty} \left\langle \frac{\boldsymbol{W}(k)}{\|\boldsymbol{W}(k)\|_F}, \frac{\boldsymbol{W}^{\mathrm{mm}}}{\|\boldsymbol{W}^{\mathrm{mm}}\|_F} \right\rangle \geq 1 - \epsilon, \qquad \boldsymbol{W}(k) \in \bar{\mathcal{C}}_{\mu, R_\epsilon}(\boldsymbol{W}^{\mathrm{mm}}).$$

Since $\epsilon \in (0, \mu/2)$ is arbitrary, this implies $\boldsymbol{W}(k)/\|\boldsymbol{W}(k)\|_F \to \boldsymbol{W}^{\mathrm{mm}}/\|\boldsymbol{W}^{\mathrm{mm}}\|_F$. ∎

B.4 Local Convergence of Gradient Descent

To provide a basis for discussing local convergence of GD, we establish a cone centered around $\boldsymbol{W}_{\boldsymbol{\alpha}}^{\mathrm{mm}}$ using the following construction. For parameters $\mu \in (0, 1)$ and $R > 0$, we define $\mathcal{C}_{\mu,R}(\boldsymbol{W}_{\boldsymbol{\alpha}}^{\mathrm{mm}})$ as the set of matrices $\boldsymbol{W} \in \mathbb{R}^{d \times d}$ such that $\|\boldsymbol{W}\|_F \geq R$ and the correlation coefficient between $\boldsymbol{W}$ and $\boldsymbol{W}_{\boldsymbol{\alpha}}^{\mathrm{mm}}$ is at least $1 - \mu$:

$$\mathcal{S}_\mu(\boldsymbol{W}_{\boldsymbol{\alpha}}^{\mathrm{mm}}) := \left\{ \boldsymbol{W} \in \mathbb{R}^{d \times d} \ : \ \left\langle \frac{\boldsymbol{W}}{\|\boldsymbol{W}\|_F}, \frac{\boldsymbol{W}_{\boldsymbol{\alpha}}^{\mathrm{mm}}}{\|\boldsymbol{W}_{\boldsymbol{\alpha}}^{\mathrm{mm}}\|_F} \right\rangle \geq 1 - \mu \right\}, \tag{B.51a}$$

$$\mathcal{C}_{\mu,R}(\boldsymbol{W}_{\boldsymbol{\alpha}}^{\mathrm{mm}}) := \mathcal{S}_\mu(\boldsymbol{W}_{\boldsymbol{\alpha}}^{\mathrm{mm}}) \cap \left\{ \boldsymbol{W} \in \mathbb{R}^{d \times d} \ : \ \|\boldsymbol{W}\|_F \geq R \right\}. \tag{B.51b}$$

Let $(\mathcal{T}_i)_{i=1}^n$ be the set of all support indices per Definition 3.2. Let $\bar{\mathcal{T}}_i = [T] - \mathcal{T}_i - \{\alpha_i\}$ be the non-support indices. Let $\mathcal{I} \subseteq [n]$ be the sample index set that contains samples with support for locally optimal tokens as per Definition 3.2, and $\bar{\mathcal{I}}$ be the sample index set without support tokens, that is

$$\mathcal{I} = \left\{ i \in [n] \;\middle|\; \mathcal{T}_i \neq \emptyset \right\}, \quad \text{and} \quad \bar{\mathcal{I}} = \left\{ i \in [n] \;\middle|\; \mathcal{T}_i = \emptyset \right\} = [n] - \mathcal{I}. \tag{B.52}$$

Throughout, we will use the following notations:

$$\Theta = \frac{1}{\|\boldsymbol{W}^{\mathrm{mm}}\|_F}, \tag{B.53a}$$

$$\delta = \min\left\{\frac{1}{2}\min_{i\in[n]}\min_{\tau\in\bar{\mathcal{T}}_i}\left((\boldsymbol{x}_{i\alpha_i}-\boldsymbol{x}_{i\tau})^\top \boldsymbol{W}^{\mathrm{mm}}\boldsymbol{z}_i - 1\right), \frac{1}{4}\right\}, \tag{B.53b}$$

$$A = \max\left\{\max_{i\in[n],t,\tau\in[T]}\frac{(\|\boldsymbol{x}_{it}\| \vee \|\boldsymbol{x}_{it}-\boldsymbol{x}_{i\tau}\|)\cdot\|\boldsymbol{z}_i\|}{\Theta}, 1\right\}, \tag{B.53c}$$

$$\mu(\delta) = \frac{\delta^2}{32A^2}. \tag{B.53d}$$

When $\bar{\mathcal{T}}_i = \emptyset$ for all $i \in [n]$ (i.e. globally-optimal indices), we set $\delta = \infty$ as all non-neighbor related terms will disappear. Recall scores $\gamma_{it} = y_i \cdot \boldsymbol{v}^\top \boldsymbol{x}_{it}$. We define the global scalar

$$\Gamma = \sup_{i\in[n],t,\tau\in[T]} |\gamma_{it} - \gamma_{i\tau}|. \tag{B.54a}$$

We also define the score gaps over $i \in \mathcal{I}$ and support indices:

$$\begin{aligned}
&\gamma_i^{\mathrm{gap}} = \gamma_{i\alpha_i} - \max_{t\in\mathcal{T}_i}\gamma_{it}, \qquad \bar{\gamma}_i^{\mathrm{gap}} = \gamma_{i\alpha_i} - \min_{t\in\mathcal{T}_i}\gamma_{it},\\
&\gamma_{\min}^{\mathrm{gap}} = \min_{i\in\mathcal{I}}\gamma_i^{\mathrm{gap}}, \quad \text{and} \quad \bar{\gamma}_{\max}^{\mathrm{gap}} = \max_{i\in\mathcal{I}}\bar{\gamma}_i^{\mathrm{gap}}.
\end{aligned} \tag{B.54b}$$

Furthermore, letting $\boldsymbol{s}_{it} = (\mathbb{S}(\boldsymbol{X}_i\boldsymbol{W}\boldsymbol{z}_i))_t$, we define

$$S_i := \sum_{t\in\mathcal{T}_i}\boldsymbol{s}_{it}, \quad \text{and} \quad Q_i := \sum_{\tau\in\bar{\mathcal{T}}_i}\boldsymbol{s}_{i\tau}. \tag{B.55}$$

In the following lemma, we analyze the behaviors of the (W-SVM) constraint $(\boldsymbol{x}_{it}-\boldsymbol{x}_{i\tau})^\top \boldsymbol{V}\boldsymbol{z}_i$ for all $\boldsymbol{V} \in \mathcal{S}_\mu(\boldsymbol{W}_{\boldsymbol{\alpha}}^{\mathrm{mm}})$ satisfying $\|\boldsymbol{V}\|_F = \|\boldsymbol{W}_{\boldsymbol{\alpha}}^{\mathrm{mm}}\|_F$.

Lemma B.6 *Let $\boldsymbol{W}^{mm} = \boldsymbol{W}_{\boldsymbol{\alpha}}^{mm}$ denote the SVM solution obtained via* (W-SVM) *by replacing $(opt_i)_{i=1}^n$ with $\boldsymbol{\alpha} = (\alpha_i)_{i=1}^n$ and let δ be defined in* (B.53b)*. For any $\boldsymbol{V} \in \mathtt{cone}_\mu(\boldsymbol{W}^{mm})$ with $\|\boldsymbol{V}\|_F = \|\boldsymbol{W}^{mm}\|_F$ and $\mu \le \mu(\delta)$ (cf.* (B.53d)*), and for all $i \in [n]$, $t \in \mathcal{T}_i$, and $\tau \in \bar{\mathcal{T}}_i$, we have*

$$(\boldsymbol{x}_{it}-\boldsymbol{x}_{i\tau})^\top \boldsymbol{V}\boldsymbol{z}_i \ge \frac{3}{2}\delta > 0, \tag{B.56a}$$

$$(\boldsymbol{x}_{i\alpha_i}-\boldsymbol{x}_{i\tau})^\top \boldsymbol{V}\boldsymbol{z}_i \ge 1 + \frac{3}{2}\delta, \tag{B.56b}$$

$$1 + \frac{1}{2}\delta \ge (\boldsymbol{x}_{i\alpha_i}-\boldsymbol{x}_{it})^\top \boldsymbol{V}\boldsymbol{z}_i \ge 1 - \frac{1}{2}\delta. \tag{B.56c}$$

Proof. To show (B.56), we note that for any $\boldsymbol{V} \in \mathtt{cone}_{\mu}(\boldsymbol{W}^{\mathrm{mm}})$ such that $\|\boldsymbol{V}\|_F = \|\boldsymbol{W}^{\mathrm{mm}}\|_F$,

$$\begin{aligned}
\|\boldsymbol{V} - \boldsymbol{W}^{\mathrm{mm}}\|_F &= \sqrt{2\|\boldsymbol{W}^{\mathrm{mm}}\|_F^2 - 2\left\langle \boldsymbol{V}, \boldsymbol{W}^{\mathrm{mm}} \right\rangle} \\
&= \|\boldsymbol{W}^{\mathrm{mm}}\|_F \sqrt{2 - 2\left\langle \frac{\boldsymbol{V}}{\|\boldsymbol{W}^{\mathrm{mm}}\|_F}, \frac{\boldsymbol{W}^{\mathrm{mm}}}{\|\boldsymbol{W}^{\mathrm{mm}}\|_F} \right\rangle} \\
&\leq \frac{1}{\Theta}\sqrt{2 - 2(1-\mu)} \\
&\leq \frac{\delta}{4A\Theta}.
\end{aligned} \tag{B.57}$$

Here, the first inequality follows from (B.51a), and the last equality uses (B.53d).

We will proceed to show a bound on $(\boldsymbol{x}_{it_1} - \boldsymbol{x}_{it_2})^\top (\boldsymbol{V} - \boldsymbol{W}^{\mathrm{mm}})\boldsymbol{z}_i$ for any $i \in [n]$ and any token indices $t_1, t_2 \in [T]$. For any token index $t \in [T]$, we have

$$\begin{aligned}
\left|\boldsymbol{x}_{it}^\top (\boldsymbol{V} - \boldsymbol{W}^{\mathrm{mm}})\boldsymbol{z}_i\right| &= \left|\langle \boldsymbol{V} - \boldsymbol{W}^{\mathrm{mm}}, \boldsymbol{x}_{it}\boldsymbol{z}_i^\top \rangle\right| \\
&\leq \|\boldsymbol{V} - \boldsymbol{W}^{\mathrm{mm}}\|_F \|\boldsymbol{x}_{it}\boldsymbol{z}_i^\top\|_F \\
&\leq \frac{\delta}{4A\Theta} \cdot \|\boldsymbol{x}_{it}\boldsymbol{z}_i^\top\|_F \\
&\leq \frac{\delta}{4A\Theta} \cdot \Theta A \\
&= \frac{\delta}{4}.
\end{aligned} \tag{B.58}$$

Here, the first inequality uses Cauchy-Schwarz; the second inequality follows from (B.57); and the third inequality is obtained from (B.53c).

Hence, it follows from (B.58) that

$$\text{for all } \; t_1, t_2 \in [T] \; \text{ and any } \; i \in [n], \qquad \left|(\boldsymbol{x}_{it_1} - \boldsymbol{x}_{it_2})^\top (\boldsymbol{V} - \boldsymbol{W}^{\mathrm{mm}})\, \boldsymbol{z}_i\right| \leq \frac{1}{2}\delta. \tag{B.59}$$

Note that $\boldsymbol{W}^{\mathrm{mm}} = \boldsymbol{W}^{\mathrm{mm}}_{\boldsymbol{\alpha}}$ is the SVM solution obtained via (W-SVM) by replacing $(\mathrm{opt}_i)_{i=1}^n$ with $\boldsymbol{\alpha} = (\alpha_i)_{i=1}^n$. Hence, from the definition of δ in (B.53b), we have for all $i \in [n]$, $t \in \mathcal{T}_i$ and $\tau \in \bar{\mathcal{T}}_i$ that

$$\delta \leq \frac{1}{2}\left((\boldsymbol{x}_{i\alpha_i} - \boldsymbol{x}_{i\tau})^\top \boldsymbol{W}^{\mathrm{mm}} \boldsymbol{z}_i - 1\right) \leq \frac{1}{2}(\boldsymbol{x}_{it} - \boldsymbol{x}_{i\tau})^\top \boldsymbol{W}^{\mathrm{mm}} \boldsymbol{z}_i.$$

This together with (B.59) implies that for all $t \in \mathcal{T}_i$ and $\tau \in \bar{\mathcal{T}}_i$,

$$\begin{aligned}
(\boldsymbol{x}_{it} - \boldsymbol{x}_{i\tau})^\top \boldsymbol{V}\boldsymbol{z}_i &\geq (\boldsymbol{x}_{it} - \boldsymbol{x}_{i\tau})^\top \boldsymbol{W}^{\mathrm{mm}}\boldsymbol{z}_i + (\boldsymbol{x}_{it} - \boldsymbol{x}_{i\tau})^\top (\boldsymbol{V} - \boldsymbol{W}^{\mathrm{mm}})\, \boldsymbol{z}_i \\
&\geq 2\delta - \frac{1}{2}\delta = \frac{3}{2}\delta, \\
(\boldsymbol{x}_{i\alpha_i} - \boldsymbol{x}_{i\tau})^\top \boldsymbol{V}\boldsymbol{z}_i &\geq (\boldsymbol{x}_{i\alpha_i} - \boldsymbol{x}_{i\tau})^\top \boldsymbol{W}^{\mathrm{mm}}\boldsymbol{z}_i + (\boldsymbol{x}_{i\alpha_i} - \boldsymbol{x}_{i\tau})^\top (\boldsymbol{V} - \boldsymbol{W}^{\mathrm{mm}})\boldsymbol{z}_i \\
&\geq 1 + 2\delta - \frac{1}{2}\delta = 1 + \frac{3}{2}\delta, \\
\big|(\boldsymbol{x}_{i\alpha_i} - \boldsymbol{x}_{it})^\top \boldsymbol{V}\boldsymbol{z}_i - 1\big| &= \big|(\boldsymbol{x}_{i\alpha_i} - \boldsymbol{x}_{it})^\top \boldsymbol{W}^{\mathrm{mm}}\boldsymbol{z}_i + (\boldsymbol{x}_{i\alpha_i} - \boldsymbol{x}_{it})^\top (\boldsymbol{V} - \boldsymbol{W}^{\mathrm{mm}})\boldsymbol{z}_i - 1\big| \\
&= \big|(\boldsymbol{x}_{i\alpha_i} - \boldsymbol{x}_{it})^\top (\boldsymbol{V} - \boldsymbol{W}^{\mathrm{mm}})\boldsymbol{z}_i\big| \leq \frac{1}{2}\delta.
\end{aligned}$$

This completes the proof. ∎

Lemma B.7 *Suppose Assumption 3.1 on the loss function ℓ holds, and let $\boldsymbol{\alpha} = (\alpha_i)_{i=1}^n$ be locally optimal tokens according to Definition 3.2. Let $\boldsymbol{W}^{mm} = \boldsymbol{W}_{\boldsymbol{\alpha}}^{mm}$ denote the SVM solution obtained via* (W-SVM) *by replacing $(opt_i)_{i=1}^n$ with $\boldsymbol{\alpha} = (\alpha_i)_{i=1}^n$. There exists a positive scalar μ such that for sufficiently large $\bar{R}_\mu$: For all $\boldsymbol{V} \in \mathcal{S}_\mu(\boldsymbol{W}^{mm})$ with $\|\boldsymbol{V}\|_F = \|\boldsymbol{W}^{mm}\|_F$ and $\boldsymbol{W} \in \mathcal{C}_{\mu,\bar{R}_\mu}(\boldsymbol{W}^{mm})$, there exist dataset dependent positive constants C, $\hat{C}$, and c such that*

$$C \cdot \frac{1}{n}\sum_{i\in\mathcal{I}} (1 - \boldsymbol{s}_{i\alpha_i}) \geq -\Big\langle \nabla\mathcal{L}(\boldsymbol{W}), \boldsymbol{V}\Big\rangle \geq c \cdot \frac{1}{n}\sum_{i\in\mathcal{I}} (1 - \boldsymbol{s}_{i\alpha_i}) > 0, \tag{B.60a}$$

$$\|\nabla\mathcal{L}(\boldsymbol{W})\|_F \leq A\hat{C} \cdot \frac{1}{n}\sum_{i\in\mathcal{I}} (1 - \boldsymbol{s}_{i\alpha_i}), \tag{B.60b}$$

$$-\left\langle \frac{\boldsymbol{V}}{\|\boldsymbol{V}\|_F}, \frac{\nabla\mathcal{L}(\boldsymbol{W})}{\|\nabla\mathcal{L}(\boldsymbol{W})\|_F} \right\rangle \geq \frac{c}{\hat{C}} \cdot \frac{\Theta}{A} > 0. \tag{B.60c}$$

Here, $\boldsymbol{s}_{i\alpha_i} = (\mathbb{S}(\boldsymbol{X}_i\boldsymbol{W}\boldsymbol{z}_i))_{\alpha_i}$; $\mathcal{I} \subseteq [n]$ is defined in (B.52)*; and Θ, A are defined in* (B.53).

Proof. Let $R = \|\boldsymbol{W}\|_F$,

$$\ell_i' = \ell'(y_i \cdot \boldsymbol{v}^\top \boldsymbol{X}_i^\top \mathbb{S}(\tilde{\boldsymbol{h}}_i)), \qquad \boldsymbol{h}_i = \boldsymbol{X}_i\boldsymbol{V}\boldsymbol{z}_i, \qquad \tilde{\boldsymbol{h}}_i = \boldsymbol{X}_i\boldsymbol{W}\boldsymbol{z}_i, \qquad \boldsymbol{s}_i = \mathbb{S}(\tilde{\boldsymbol{h}}_i). \tag{B.61}$$

Before the main proof of the lemma, we provide bounds for $|\boldsymbol{h}_{it}|$, Q_i, S_i, and Q_i/S_i (cf. (B.55)). To proceed, recall scores $\gamma_{it} = y_i \cdot \boldsymbol{v}^\top \boldsymbol{x}_{it}$. Not that using (B.53c), for all $\boldsymbol{V} \in \mathbb{R}^{d\times d}$ with $\|\boldsymbol{V}\|_F = \|\boldsymbol{W}^{\mathrm{mm}}\|_F$, and for all $i \in [n], t \in [T]$, we have

$$|\boldsymbol{h}_{it}| = |\boldsymbol{x}_{it}^\top \boldsymbol{V}\boldsymbol{z}_i| = |\langle \boldsymbol{x}_{it}\boldsymbol{z}_i^\top, \boldsymbol{V}\rangle| \leq \|\boldsymbol{V}\|_F\, \|\boldsymbol{x}_{it}\boldsymbol{z}_i^\top\| \leq A. \tag{B.62}$$

Here, the last inequality uses (B.53c).

Let us assume (without losing generality) $\alpha_i = 1$ for all $i \in [n]$. Note that $\mathcal{T}_i$ can be empty for some $i \in [n]$. In this case, we set $S_i = 0$. Similarly, $\bar{\mathcal{T}}_i$ can also be empty for some $i \in [n]$, in which case we set $Q_i = 0$ which satisfies (B.63a). Note also that for all $\boldsymbol{W} \in \mathcal{S}_\mu(\boldsymbol{W}^{\mathrm{mm}})$, $\|\boldsymbol{W}^{\mathrm{mm}}\|_F \boldsymbol{W}/\|\boldsymbol{W}\|_F$ satisfies the requirement of Lemma B.6. Hence, we can provide bounds for Q_i and S_i in the following. If Q_i is positive, for any $\boldsymbol{W} \in \mathcal{S}_\mu(\boldsymbol{W}^{\mathrm{mm}})$ and with $\bar{\mathcal{T}}_i$ as the set of all non-support indices, we have

$$Q_i = \frac{\sum_{t\in\bar{\mathcal{T}}_i} e^{\boldsymbol{x}_{it}^\top \boldsymbol{W}\boldsymbol{z}_i}}{\sum_{t=1}^T e^{\boldsymbol{x}_{it}^\top \boldsymbol{W}\boldsymbol{z}_i}} \le \frac{\sum_{t\in\bar{\mathcal{T}}_i} e^{\boldsymbol{x}_{it}^\top \boldsymbol{W}\boldsymbol{z}_i}}{e^{\boldsymbol{x}_{i1}^\top \boldsymbol{W}\boldsymbol{z}_i}} \le T e^{-R\Theta\left(1+\frac{3}{2}\delta\right)}. \tag{B.63a}$$

Here, the last inequality uses our definition $R = \|\boldsymbol{W}\|_F$ for all $\boldsymbol{W} \in \mathcal{S}_\mu(\boldsymbol{W}^{\mathrm{mm}})$, $\Theta = 1/\|\boldsymbol{W}^{\mathrm{mm}}\|_F$, and (B.56b).

Similarly, in the case S_i is positive, for any $\boldsymbol{W} \in \mathcal{S}_\mu(\boldsymbol{W}^{\mathrm{mm}})$ and $\mathcal{T}_i$ as the set of all support indices, we have

$$S_i = \frac{\sum_{t\in\mathcal{T}_i} e^{\boldsymbol{x}_{it}^\top \boldsymbol{W}\boldsymbol{z}_i}}{\sum_{t=1}^T e^{\boldsymbol{x}_{it}^\top \boldsymbol{W}\boldsymbol{z}_i}} \ge \frac{\sum_{t\in\mathcal{T}_i} e^{\boldsymbol{x}_{it}^\top \boldsymbol{W}\boldsymbol{z}_i}}{T e^{\boldsymbol{x}_{i1}^\top \boldsymbol{W}\boldsymbol{z}_i}} \ge \frac{1}{T} e^{-R\Theta(1+\frac{1}{2}\delta)}, \tag{B.63b}$$

$$S_i = \frac{\sum_{t\in\mathcal{T}_i} e^{\boldsymbol{x}_{it}^\top \boldsymbol{W}\boldsymbol{z}_i}}{\sum_{t=1}^T e^{\boldsymbol{x}_{it}^\top \boldsymbol{W}\boldsymbol{z}_i}} \le \frac{\sum_{t\in\mathcal{T}_i} e^{\boldsymbol{x}_{it}^\top \boldsymbol{W}\boldsymbol{z}}}{e^{\boldsymbol{x}_{i1}^\top \boldsymbol{W}\boldsymbol{z}_i}} \le T e^{-R\Theta(1-\frac{1}{2}\delta)}. \tag{B.63c}$$

Here, the last inequalities in (B.63b) and (B.63c) follow from (B.56c).

Further, for positive S_i and any $\boldsymbol{W} \in \mathcal{S}_\mu(\boldsymbol{W}^{\mathrm{mm}})$, from (B.56a), we have

$$\frac{Q_i}{S_i} = \frac{\sum_{t\in\bar{\mathcal{T}}_i} e^{\boldsymbol{x}_{it}^\top \boldsymbol{W}\boldsymbol{z}_i}}{\sum_{t\in\mathcal{T}_i} e^{\boldsymbol{x}_{it}^\top \boldsymbol{W}\boldsymbol{z}_i}} \le T e^{-\frac{3}{2}R\Theta\delta}. \tag{B.63d}$$

Proof of (B.60a):

To proceed, we write the gradient correlation following (B.3) and (B.14) as

$$\begin{aligned} -\langle \nabla \mathcal{L}(\boldsymbol{W}), \boldsymbol{V}\rangle &= \frac{1}{n}\sum_{i=1}^n (-\ell_i') \cdot \boldsymbol{h}_i^\top \mathbb{S}'(\tilde{\boldsymbol{h}}_i)\gamma_i, \\ &= \frac{1}{n}\sum_{i\in\mathcal{I}} (-\ell_i') \cdot \boldsymbol{h}_i^\top \mathbb{S}'(\tilde{\boldsymbol{h}}_i)\gamma_i + \frac{1}{n}\sum_{i\in\bar{\mathcal{I}}} (-\ell_i') \cdot \boldsymbol{h}_i^\top \mathbb{S}'(\tilde{\boldsymbol{h}}_i)\gamma_i. \end{aligned} \tag{B.64}$$

Note that to show (B.60a), intuitively, we will show that there exists $\bar{R}_\mu$ such that for any $R \geq \bar{R}_\mu$, $-\frac{1}{n}\sum_{i\in\mathcal{I}} \ell_i' \cdot \boldsymbol{h}_i^\top \mathbb{S}'(\tilde{\boldsymbol{h}}_i)\gamma_i$ dominates $-\frac{1}{n}\sum_{i\in\bar{\mathcal{I}}} \ell_i' \cdot \boldsymbol{h}_i^\top \mathbb{S}'(\tilde{\boldsymbol{h}}_i)\gamma_i$. Directly applying Lemma B.2, for any $i \in [n]$, we obtain

$$\left| \boldsymbol{h}_i^\top \mathrm{diag}(\boldsymbol{s}_i)\gamma_i - \boldsymbol{h}_i^\top \boldsymbol{s}_i \boldsymbol{s}_i^\top \gamma_i - \sum_{t\geq 2}^{T} (\boldsymbol{h}_{i1} - \boldsymbol{h}_{it})\boldsymbol{s}_{it}(\gamma_{i1} - \gamma_{it}) \right| \leq 2\Gamma A (1 - \boldsymbol{s}_{i1})^2. \tag{B.65a}$$

To proceed, let us decouple the non-support indices within $\sum_{t\geq 2}^{T}(\boldsymbol{h}_{i1} - \boldsymbol{h}_{it})\boldsymbol{s}_{it}(\gamma_{i1} - \gamma_{it})$. Recall $\bar{\mathcal{T}}_i$ is the set of all non-support indices per Definition 3.2. We have

$$\left| \sum_{t\in\bar{\mathcal{T}}_i} (\boldsymbol{h}_{i1} - \boldsymbol{h}_{it})\boldsymbol{s}_{it}(\gamma_{i1} - \gamma_{it}) \right| \leq 2Q_i \Gamma A. \tag{B.65b}$$

It follows from (B.65a) and (B.65b) for $\mathcal{T}_i$ as the set of all support indices,

$$\left| \boldsymbol{h}_i^\top \mathrm{diag}(\boldsymbol{s}_i)\gamma_i - \boldsymbol{h}_i^\top \boldsymbol{s}_i \boldsymbol{s}_i^\top \gamma_i - \sum_{t\in\mathcal{T}_i} (\boldsymbol{h}_{i1} - \boldsymbol{h}_{it})\boldsymbol{s}_{it}(\gamma_{i1} - \gamma_{it}) \right| \leq 2\Gamma A \left((1 - \boldsymbol{s}_{i1})^2 + Q_i \right). \tag{B.66}$$

In the following, we consider bounding (B.64) for samples from $\mathcal{I}$ and $\bar{\mathcal{I}}$ separately.

Case 1: $i \in \mathcal{I}$ (Samples with support tokens). Recall $\boldsymbol{h}_{i1} - \boldsymbol{h}_{it} = (\boldsymbol{x}_{i1} - \boldsymbol{x}_{it})^\top \boldsymbol{V}\boldsymbol{z}_i$ for all $\boldsymbol{V} \in \mathcal{S}_\mu(\boldsymbol{W}^{\mathrm{mm}})$ with $\|\boldsymbol{V}\|_F = \|\boldsymbol{W}^{\mathrm{mm}}\|_F$. Hence, it follows from (B.56c) that

$$\left(1 + \frac{\delta}{2}\right) \sum_{t\in\mathcal{T}_i} \boldsymbol{s}_{it}(\gamma_{i1} - \gamma_{it}) \geq \sum_{t\in\mathcal{T}_i} (\boldsymbol{h}_{i1} - \boldsymbol{h}_{it})\, \boldsymbol{s}_{it}\, (\gamma_{i1} - \gamma_{it}) \geq \left(1 - \frac{\delta}{2}\right) \sum_{t\in\mathcal{T}_i} \boldsymbol{s}_{it}(\gamma_{i1} - \gamma_{it}). \tag{B.67}$$

From (B.54b), we get

$$\left(1 + \frac{\delta}{2}\right) \bar{\gamma}_i^{\mathrm{gap}} \sum_{t\in\mathcal{T}_i} \boldsymbol{s}_{it} \geq \sum_{t\in\mathcal{T}_i} (\boldsymbol{h}_{i1} - \boldsymbol{h}_{it})\boldsymbol{s}_{it}(\gamma_{i1} - \gamma_{it}) \geq \left(1 - \frac{\delta}{2}\right) \gamma_i^{\mathrm{gap}} \sum_{t\in\mathcal{T}_i} \boldsymbol{s}_{it}. \tag{B.68}$$

Note that $S_i = \sum_{t\in\mathcal{T}_i} \boldsymbol{s}_{it}$. The above inequality together with (B.56) implies that

$$\boldsymbol{h}_i^\top \mathrm{diag}(\boldsymbol{s}_i)\gamma_i - \boldsymbol{h}_i^\top \boldsymbol{s}_i \boldsymbol{s}_i^\top \gamma_i \leq \left(1 + \frac{\delta}{2}\right) S_i \bar{\gamma}_i^{\mathrm{gap}} + 2\Gamma A \left((1 - \boldsymbol{s}_{i1})^2 + Q_i \right), \tag{B.69a}$$

$$\boldsymbol{h}_i^\top \mathrm{diag}(\boldsymbol{s}_i)\gamma_i - \boldsymbol{h}_i^\top \boldsymbol{s}_i \boldsymbol{s}_i^\top \gamma_i \geq \left(1 - \frac{\delta}{2}\right) S_i \gamma_i^{\mathrm{gap}} - 2\Gamma A \left((1 - \boldsymbol{s}_{i1})^2 + Q_i \right). \tag{B.69b}$$

We claim that for each sample $i \in \mathcal{I}$, S_i dominates $((1 - \boldsymbol{s}_{i1})^2 + Q_i)$ for large R. Specifically,

we aim for

$$\left(1-\frac{\delta}{2}\right) S_i \cdot \gamma_i^{\text{gap}} \geq 8\Gamma A \max\left((1-\boldsymbol{s}_{i1})^2, Q_i\right) \iff S_i \geq 8\left(1-\frac{\delta}{2}\right)\frac{\Gamma A}{\gamma_i^{\text{gap}}}\max\left((1-\boldsymbol{s}_{i1})^2, Q_i\right). \tag{B.70}$$

It follows from (B.63d),

$$R \geq R_1 = \frac{2\log(T)}{3\delta\Theta}, \quad \text{implies that} \quad Q_i \leq S_i \qquad \text{for all } i \in \mathcal{I}. \tag{B.71}$$

Consequently,

$$(1-\boldsymbol{s}_{i1})^2 = (Q_i + S_i)^2 \leq 4S_i^2 \leq 4S_i T e^{-R\Theta\left(1-\frac{1}{2}\delta\right)}. \tag{B.72}$$

Substituting (B.63d) and (B.72) into (B.70), we achieve our goal by ensuring that

$$S_i \geq 8\left(1-\frac{\delta}{2}\right)\frac{\Gamma A}{\gamma_i^{\text{gap}}}\max\left(4S_i T e^{-R\Theta(1-\frac{1}{2}\delta)}, S_i T e^{-R\Theta\frac{3}{2}\delta}\right). \tag{B.73}$$

Equivalently, using the fact that $\delta \leq 1/4$ by (B.53b), we achieve (B.70) by ensuring that

$$1 \geq 32\left(1-\frac{\delta}{2}\right)\frac{\Gamma A}{\gamma_i^{\text{gap}}} T e^{-R\Theta\frac{3}{2}\delta}.$$

This in turn is ensured for all inputs $i \in \mathcal{I}$ by choosing

$$R \geq \max(R_1, R_2), \quad \text{where} \quad R_2 = \frac{2}{3\delta\Theta}\log\left(\frac{16\,(2-\delta)\,T\Gamma A}{\gamma_{\min}^{\text{gap}}}\right). \tag{B.74}$$

Here, R_1 is defined in (B.71), and $\gamma_{\min}^{\text{gap}} = \min_{i\in\mathcal{I}} \gamma_i^{\text{gap}}$ is defined in (B.54b) as the global scalar which is the worst case score gap over all inputs.

With the choice of R in (B.74), we have ensured (B.70), which, together with (B.69), implies that

$$2\left(1+\frac{\delta}{2}\right) S_i \bar{\gamma}_i^{\text{gap}} \geq \boldsymbol{h}_i^\top \operatorname{diag}(\boldsymbol{s}_i)\gamma_i - \boldsymbol{h}_i^\top \boldsymbol{s}_i \boldsymbol{s}_i^\top \gamma_i \geq \frac{1}{2}\left(1-\frac{\delta}{2}\right) S_i \gamma_i^{\text{gap}}. \tag{B.75}$$

Let $-\ell'_{\min}$ and $-\ell'_{\max}$ be the min and max values negative loss derivative admits over the ball $[-A, A]$, respectively. Note that $\max_{i\in\mathcal{I}} \bar{\gamma}_i^{\text{gap}} > 0$ and $\min_{i\in\mathcal{I}} \gamma_i^{\text{gap}} > 0$ are dataset dependent constants. Then, using (B.54b), (B.75), and the choice of R in (B.74), the first

term in (B.64) can be bounded as

$$(-\ell'_{\max}) \cdot \frac{(2+\delta)\bar{\gamma}^{\text{gap}}_{\max}}{n} \sum_{i\in\mathcal{I}} S_i \geq -\frac{1}{n} \sum_{i\in\mathcal{I}} \ell'_i \cdot \boldsymbol{h}_i^\top \mathbb{S}'(\tilde{\boldsymbol{h}}_i)\gamma_i \geq (-\ell'_{\min}) \cdot \frac{(2-\delta)\gamma^{\text{gap}}_{\min}}{4n} \sum_{i\in\mathcal{I}} S_i. \quad \text{(B.76)}$$

Case 2: $i \in \bar{\mathcal{I}}$ (Samples without support tokens). Note that if $i \in \bar{\mathcal{I}}$, then $S_i = 0$ and it follows from (B.69) that for all $i \in \bar{\mathcal{I}}$,

$$\left|\boldsymbol{h}_i^\top \text{diag}(\boldsymbol{s}_i)\gamma_i - \boldsymbol{h}_i^\top \boldsymbol{s}_i \boldsymbol{s}_i^\top \gamma_i\right| \leq 2\Gamma A \left((1-\boldsymbol{s}_{i1})^2 + Q_i\right) = 2\Gamma A \left(Q_i^2 + Q_i\right) \leq 4\Gamma A Q_i, \quad \text{(B.77)}$$

where the last inequality uses $(1-\boldsymbol{s}_{i1})^2 = (Q_i + S_i)^2 = Q_i^2 \leq Q_i$.

Recall $-\ell'_{\max}$ be the max values negative loss derivative admits over the ball $[-A, A]$. Using (B.77) and (B.64) we have that

$$(-\ell'_{\max}) \cdot \frac{4\Gamma A}{n} \sum_{i\in\bar{\mathcal{I}}} Q_i \geq -\frac{1}{n} \sum_{i\in\bar{\mathcal{I}}} \ell'_i \cdot \boldsymbol{h}_i^\top \mathbb{S}'(\tilde{\boldsymbol{h}}_i)\gamma_i \geq \ell'_{\max} \cdot \frac{4\Gamma A}{n} \sum_{i\in\bar{\mathcal{I}}} Q_i. \quad \text{(B.78)}$$

Using (B.76) and (B.78), we will show that

$$(-\ell'_{\min}) \cdot \frac{(2-\delta)\gamma^{\text{gap}}_{\min}}{8n} \sum_{i\in\mathcal{I}} S_i \geq (-\ell'_{\max}) \cdot \frac{4\Gamma A}{n} \sum_{i\in\bar{\mathcal{I}}} Q_i. \quad \text{(B.79)}$$

Using (B.63a) and (B.63b), the above inequality is implied by

$$-\frac{(2-\delta)\gamma^{\text{gap}}_{\min}\ell'_{\min}}{8} |\mathcal{I}| \cdot \frac{1}{T} e^{-R\Theta(1+\frac{1}{2}\delta)} \geq -4\Gamma A \ell'_{\max} |\bar{\mathcal{I}}| \cdot T e^{-R\Theta(1+\frac{3}{2}\delta)}.$$

This gives the following choice of R

$$R \geq R_3 = \frac{1}{\delta\Theta} \log\left(\frac{32|\bar{\mathcal{I}}|T^2\Gamma A \ell'_{\max}}{(2-\delta)|\mathcal{I}|\gamma^{\text{gap}}_{\min}\ell'_{\min}}\right). \quad \text{(B.80)}$$

Hence, the above choice of R ensures that (B.78) holds.

Considering both *Case 1* and *Case 2*, and combining (B.74) with (B.80), we can select

$$R \geq \max\left(R_1, R_2, R_3\right), \quad \text{(B.81)}$$

which, from (B.76) and (B.78), implies that

$$-\frac{2(2+\delta)\bar{\gamma}_{\max}^{\text{gap}}\ell'_{\max}}{n}\sum_{i\in\mathcal{I}}S_i \geq -\frac{1}{n}\sum_{i\in\mathcal{I}}\ell'_i\cdot\boldsymbol{h}_i^\top\mathbb{S}'(\tilde{\boldsymbol{h}}_i)\gamma_i - \frac{1}{n}\sum_{i\in\bar{\mathcal{I}}}\ell'_i\cdot\boldsymbol{h}_i^\top\mathbb{S}'(\tilde{\boldsymbol{h}}_i)\gamma_i$$
$$\geq -\frac{(2-\delta)\gamma_{\min}^{\text{gap}}\ell'_{\min}}{8n}\sum_{i\in\mathcal{I}}S_i.$$

The above inequality together with (B.64) implies that

$$-2(2+\delta)\bar{\gamma}_{\max}^{\text{gap}}\ell'_{\max}\cdot\frac{1}{n}\sum_{i\in\mathcal{I}}(1-\boldsymbol{s}_{i1}) \geq \langle-\nabla\mathcal{L}(\boldsymbol{W}),\boldsymbol{V}\rangle$$
$$\geq -\frac{(2-\delta)\gamma_{\min}^{\text{gap}}\ell'_{\min}}{16}\cdot\frac{1}{n}\sum_{i\in\mathcal{I}}(1-\boldsymbol{s}_{i1}). \tag{B.82}$$

Here, the first inequality follows since $(1-\boldsymbol{s}_{i1})\geq S_i$ and the last inequality is obtained from the fact that $(1-\boldsymbol{s}_{i1})/2=(Q_i+S_i)/2\leq S_i$ due to (B.63d) and the choice of R in (B.74).

Note that since $-\ell'_{\min}>0$ due to Assumption 3.1 and $\sum_{i\in\mathcal{I}}(1-\boldsymbol{s}_{i1})>0$, the right-hand side of (B.82) is positive. Hence, by choosing $R\geq\bar{R}_\mu$, there is no stationary point within $\mathcal{C}_{\mu,\bar{R}_\mu}(\boldsymbol{W}^{\text{mm}})$. Finally, we declare the constants

$$C=2(2+\delta)\bar{\gamma}_{\max}^{\text{gap}}\cdot(-\ell'_{\max})>0, \quad\text{and}\quad c=\frac{1}{16}(2-\delta)\gamma_{\min}^{\text{gap}}\cdot(-\ell'_{\min})>0. \tag{B.83}$$

This gives (B.60a).

Proof of (B.60b) **and** (B.60c):

Let $\hat{\boldsymbol{h}}_i=\boldsymbol{X}_i\hat{\boldsymbol{V}}\boldsymbol{z}_i$ for all $\hat{\boldsymbol{V}}\in\mathbb{R}^{d\times d}$ with $\|\hat{\boldsymbol{V}}\|_F=\|\boldsymbol{W}^{\text{mm}}\|_F$. Note that similar to (B.62), we have for all $i\in[n]$ and $t\in[T]$ that

$$|\hat{\boldsymbol{h}}_{it}|\leq\|\hat{\boldsymbol{V}}\|\,\|\boldsymbol{x}_{it}\boldsymbol{z}_i^\top\|\leq A. \tag{B.84}$$

Recall that $\boldsymbol{s}_i=\mathbb{S}(\boldsymbol{X}_i\boldsymbol{W}\boldsymbol{z}_i)$, with $\boldsymbol{W}\in\mathcal{S}_\mu(\boldsymbol{W}^{\text{mm}})$, as stated in Lemma B.7. Following similar argument as in (B.65)–(B.66), we obtain

$$\left|\hat{\boldsymbol{h}}_i^\top\text{diag}(\boldsymbol{s}_i)\gamma_i-\hat{\boldsymbol{h}}_i^\top\boldsymbol{s}_i\boldsymbol{s}_i^\top\gamma_i-\sum_{t\in\mathcal{T}_i}(\hat{\boldsymbol{h}}_{i1}-\hat{\boldsymbol{h}}_{it})\boldsymbol{s}_{it}(\gamma_{i1}-\gamma_{it})\right|\leq 2\Gamma A\left((1-\boldsymbol{s}_{i1})^2+Q_i\right). \tag{B.85}$$

Next, we consider bounding (B.69) with $\boldsymbol{h}$ replaced with $\hat{\boldsymbol{h}}$ for samples from $\mathcal{I}$ and $\bar{\mathcal{I}}$,

separably.

Case A: $i \in \mathcal{I}$ (Samples with support tokens). Note that $\hat{\boldsymbol{h}}_{i1} - \hat{\boldsymbol{h}}_{it} = (\boldsymbol{x}_{i1} - \boldsymbol{x}_{it})^\top \hat{\boldsymbol{V}} \boldsymbol{z}_i$ for all $\hat{\boldsymbol{V}} \in \mathbb{R}^{d\times d}$ with $\|\hat{\boldsymbol{V}}\|_F = \|\boldsymbol{W}^{\mathrm{mm}}\|_F$. Hence,

$$2A \sum_{t\in\mathcal{T}_i} \boldsymbol{s}_{it}(\gamma_{i1} - \gamma_{it}) \geq \sum_{t\in\mathcal{T}_i} \left(\hat{\boldsymbol{h}}_{i1} - \hat{\boldsymbol{h}}_{it}\right) \boldsymbol{s}_{it} \left(\gamma_{i1} - \gamma_{it}\right). \tag{B.86}$$

From (B.54b), we get

$$2A\bar{\gamma}_i^{\mathrm{gap}} \sum_{t\in\mathcal{T}_i} \boldsymbol{s}_{it} \geq \sum_{t\in\mathcal{T}_i} (\hat{\boldsymbol{h}}_{i1} - \hat{\boldsymbol{h}}_{it}) \boldsymbol{s}_{it} (\gamma_{i1} - \gamma_{it}). \tag{B.87}$$

Using (B.85) and (B.87), we get

$$2A\bar{\gamma}_i^{\mathrm{gap}} S_i + 2A\Gamma \left((1 - \boldsymbol{s}_{i1})^2 + Q_i\right) \geq \hat{\boldsymbol{h}}_i^\top \mathrm{diag}(\boldsymbol{s}_i)\gamma_i - \hat{\boldsymbol{h}}_i^\top \boldsymbol{s}_i \boldsymbol{s}_i^\top \gamma_i, \quad \text{for all } \ i \in \mathcal{I}. \tag{B.88}$$

Since $\boldsymbol{s}_i = \mathbb{S}(\tilde{\boldsymbol{h}}_i)$ with $\tilde{\boldsymbol{h}}_i = \boldsymbol{X}_i \boldsymbol{W} \boldsymbol{z}_i$ and $\boldsymbol{W} \in \mathcal{S}_\mu(\boldsymbol{W}^{\mathrm{mm}})$, similar to (B.71), $R \geq R_1$ gives $Q_i \leq S_i$ for all $i \in \mathcal{I}$. Consequently, $(1 - \boldsymbol{s}_{i1})^2 = (Q_i + S_i)^2 \leq 4S_i^2 \leq 4S_i$. This, together with (B.54b), implies that

$$(2A\bar{\gamma}_{\max}^{\mathrm{gap}} + 10A\Gamma) S_i \geq \hat{\boldsymbol{h}}_i^\top \mathrm{diag}(\boldsymbol{s}_i)\gamma_i - \hat{\boldsymbol{h}}_i^\top \boldsymbol{s}_i \boldsymbol{s}_i^\top \gamma_i, \quad \text{for all } \ i \in \mathcal{I}, \tag{B.89}$$

Recall $-\ell'_{\max} > 0$ be the max values negative loss derivative admits over the ball $[-A, A]$. It follows from (B.89) that

$$(2A\bar{\gamma}_{\max}^{\mathrm{gap}} + 10A\Gamma) \cdot (-\ell'_{\max}) \cdot \frac{1}{n} \sum_{i\in\mathcal{I}} S_i \geq -\frac{1}{n} \sum_{i\in\mathcal{I}} \ell'_i \cdot \hat{\boldsymbol{h}}_i^\top \mathbb{S}'(\tilde{\boldsymbol{h}}_i)\gamma_i. \tag{B.90a}$$

Case B: $i \in \bar{\mathcal{I}}$ (Samples without support tokens). Since $i \in \bar{\mathcal{I}}$, we have $S_i = 0$. Hence, from (B.85), we obtain

$$2A\Gamma Q_i \geq \hat{\boldsymbol{h}}_i^\top \mathrm{diag}(\boldsymbol{s}_i)\gamma_i - \hat{\boldsymbol{h}}_i^\top \boldsymbol{s}_i \boldsymbol{s}_i^\top \gamma_i, \quad \text{for all } \ i \in \bar{\mathcal{I}},$$

which implies that

$$2A\Gamma \cdot (-\ell'_{\max}) \cdot \frac{1}{n} \sum_{i\in\bar{\mathcal{I}}} Q_i \geq -\frac{1}{n} \sum_{i\in\bar{\mathcal{I}}} \ell'_i \cdot \hat{\boldsymbol{h}}_i^\top \mathbb{S}'(\tilde{\boldsymbol{h}}_i)\gamma_i. \tag{B.90b}$$

In the following, we show that for a sufficiently large choice of R, we have

$$\sum_{i\in\bar{\mathcal{I}}} Q_i \le \sum_{i\in\mathcal{I}} S_i. \tag{B.91}$$

To do so, using (B.63), the above inequality is guaranteed if, for δ defined in (B.53b), we have

$$|\bar{\mathcal{I}}| \cdot T e^{-R\Theta(1+\frac{3}{2}\delta)} \le |\mathcal{I}| \cdot \frac{1}{T} e^{-R\Theta(1+\frac{1}{2}\delta)}.$$

This gives the following choice of R

$$R \ge R_4 = \frac{1}{\delta\Theta} \log\left(\frac{|\bar{\mathcal{I}}|T^2}{|\mathcal{I}|}\right). \tag{B.92}$$

Taking both *Case 1* and *Case 2*, and combining (B.74) and (B.92), by choosing

$$R \ge \max\left(R_1, R_4\right), \tag{B.93}$$

we obtain

$$\begin{aligned}(2A\bar{\gamma}^{\mathrm{gap}}_{\max} + 10A\Gamma + 2A\Gamma) \cdot (-\ell'_{\max}) \cdot \frac{1}{n}\sum_{i\in\mathcal{I}} S_i &\ge -\frac{1}{n}\sum_{i=1}^{n} \ell'_i \cdot \hat{\boldsymbol{h}}_i^\top \mathbb{S}'(\tilde{\boldsymbol{h}}_i)\gamma_i \\ &= -\left\langle \nabla\mathcal{L}(\boldsymbol{W}), \hat{\boldsymbol{V}} \right\rangle. \end{aligned} \tag{B.94}$$

Recall $\Theta = 1/\|\boldsymbol{W}^{\mathrm{mm}}\|_F$. We declare the constant

$$\hat{C} = 2\Theta\left(\bar{\gamma}^{\mathrm{gap}}_{\max} + 5\Gamma + \Gamma \cdot (-\ell'_{\max})\right). \tag{B.95}$$

Thus,

$$\frac{A\hat{C}}{n\Theta}\sum_{i\in\mathcal{I}}(1 - \boldsymbol{s}_{i\alpha_i}) \ge -\left\langle \nabla\mathcal{L}(\boldsymbol{W}), \hat{\boldsymbol{V}} \right\rangle. \tag{B.96}$$

To proceed, since (B.96) holds for any $\hat{\boldsymbol{V}} \in \mathbb{R}^{d\times d}$ with $\|\hat{\boldsymbol{V}}\|_F = \|\boldsymbol{W}^{\mathrm{mm}}\|_F$, we replace it with $-\frac{\|\boldsymbol{W}^{\mathrm{mm}}\|_F}{\|\nabla\mathcal{L}(\boldsymbol{W})\|_F} \cdot \nabla\mathcal{L}(\boldsymbol{W})$, to obtain

$$\left\langle \nabla\mathcal{L}(\boldsymbol{W}), \frac{\|\boldsymbol{W}^{\mathrm{mm}}\|_F}{\|\nabla\mathcal{L}(\boldsymbol{W})\|_F} \cdot \nabla\mathcal{L}(\boldsymbol{W}) \right\rangle = \|\nabla\mathcal{L}(\boldsymbol{W})\|_F \, \|\boldsymbol{W}^{\mathrm{mm}}\|_F \le \frac{A\hat{C}}{\Theta n}\sum_{i\in\mathcal{I}}(1 - \boldsymbol{s}_{i\alpha_i}). \tag{B.97a}$$

Simplifying $\Theta = 1/\|\boldsymbol{W}^{\mathrm{mm}}\|_F$ on both sides gives (B.60b).

On the other hand, it follows from (B.60a) for all $\boldsymbol{V} \in \mathcal{S}_\mu(\boldsymbol{W}^{\mathrm{mm}})$ with $\|\boldsymbol{V}\|_F = \|\boldsymbol{W}^{\mathrm{mm}}\|_F$ and $\boldsymbol{W} \in \mathcal{C}_{\mu,\bar{R}_\mu}(\boldsymbol{W}^{\mathrm{mm}})$ that

$$-\Big\langle \nabla \mathcal{L}(\boldsymbol{W}), \boldsymbol{V} \Big\rangle \geq \frac{c}{n} \sum_{i \in \mathcal{I}} (1 - \boldsymbol{s}_{i\alpha_i}) > 0. \tag{B.97b}$$

Combining (B.97a) with (B.97b), we obtain (B.60c) for all $\boldsymbol{V}, \boldsymbol{W} \in \mathcal{S}_\mu(\boldsymbol{W}^{\mathrm{mm}})$.

Finally, using (B.54) and (B.53), we have $A \geq 1$ and $\Gamma/\gamma_{\min}^{\mathrm{gap}} \geq 1$. Using (B.81), (B.92), and (B.93), $\bar{R}_\mu$ in the statement of the lemma is defined as

$$\bar{R}_\mu = \frac{1}{\delta\Theta} \log \left(32AT^2 \cdot \frac{\Gamma}{\gamma_{\min}^{\mathrm{gap}}} \cdot \frac{\ell'_{\max}}{\ell'_{\min}} \cdot \frac{|\bar{\mathcal{I}}|}{|\mathcal{I}|} \right). \tag{B.98}$$

Here, we assume that $|\bar{\mathcal{I}}|/|\mathcal{I}|$ is greater than or equal to 1 to simplify the expression. Otherwise, one can replace it with 1. ∎

Lemma B.8 *Suppose Assumption 3.1 on the loss function ℓ holds, and let $\boldsymbol{\alpha} = (\alpha_i)_{i=1}^n$ be locally optimal tokens according to Definition 3.2. Let $\boldsymbol{W}^{mm} = \boldsymbol{W}_{\boldsymbol{\alpha}}^{mm}$ denote the SVM solution obtained via* (W-SVM) *by replacing $(opt_i)_{i=1}^n$ with $\boldsymbol{\alpha} = (\alpha_i)_{i=1}^n$. Let $\mu > 0$ and $\bar{R}_\mu$ be defined as in Lemma B.7. For any choice of $\pi > 0$, there exists $R_\pi \geq \bar{R}_\mu$ such that, for any $\boldsymbol{W} \in \mathcal{C}_{\mu,R_\pi}(\boldsymbol{W}^{mm})$, we have*

$$\left\langle \nabla \mathcal{L}(\boldsymbol{W}), \frac{\boldsymbol{W}}{\|\boldsymbol{W}\|_F} \right\rangle \geq (1+\pi) \left\langle \nabla \mathcal{L}(\boldsymbol{W}), \frac{\boldsymbol{W}^{mm}}{\|\boldsymbol{W}^{mm}\|_F} \right\rangle.$$

Proof. Let

$$\bar{\boldsymbol{W}} = \frac{\|\boldsymbol{W}^{\mathrm{mm}}\|_F}{\|\boldsymbol{W}\|_F} \boldsymbol{W}, \quad \bar{\boldsymbol{h}}_i = \boldsymbol{X}_i \bar{\boldsymbol{W}} \boldsymbol{z}_i, \quad \tilde{\boldsymbol{h}}_i = \boldsymbol{X}_i \boldsymbol{W} \boldsymbol{z}_i, \quad \boldsymbol{h}_i = \boldsymbol{X}_i \boldsymbol{W}^{\mathrm{mm}} \boldsymbol{z}_i,$$
$$\boldsymbol{s}_i = \mathbb{S}(\tilde{\boldsymbol{h}}_i), \quad \text{and} \quad \ell'_i = \ell'(y_i \cdot \boldsymbol{v}^\top \boldsymbol{X}_i^\top \mathbb{S}(\tilde{\boldsymbol{h}}_i)).$$

To establish the result, we will prove that, for any $\pi > 0$, there exists sufficiently large R_π such that for any $\boldsymbol{W} \in \mathcal{C}_{\mu,R_\pi}(\boldsymbol{W}^{\mathrm{mm}})$:

$$\begin{aligned} \left\langle -\nabla \mathcal{L}(\boldsymbol{W}), \bar{\boldsymbol{W}} \right\rangle &= -\frac{1}{n} \sum_{i=1}^n \ell'_i \cdot \left\langle \bar{\boldsymbol{h}}_i, \mathbb{S}'(\tilde{\boldsymbol{h}}_i) \gamma_i \right\rangle \\ &\leq -\frac{1+\pi}{n} \sum_{i=1}^n \ell'_i \cdot \left\langle \boldsymbol{h}_i, \mathbb{S}'(\tilde{\boldsymbol{h}}_i) \gamma_i \right\rangle = (1+\pi) \left\langle -\nabla \mathcal{L}(\boldsymbol{W}), \boldsymbol{W}^{\mathrm{mm}} \right\rangle. \end{aligned} \tag{B.99}$$

Using $\mathbb{S}'(\tilde{\boldsymbol{h}}_i) = \operatorname{diag}(\boldsymbol{s}_i) - \boldsymbol{s}_i\boldsymbol{s}_i^\top$, (B.99) is equivalent to

$$\sum_{i=1}^{n} -\ell_i' \cdot \left((1+\pi)\left(\boldsymbol{h}_i^\top \operatorname{diag}(\boldsymbol{s}_i)\gamma_i - \boldsymbol{h}_i^\top \boldsymbol{s}_i\boldsymbol{s}_i^\top\gamma_i\right) - \left(\bar{\boldsymbol{h}}_i^\top \operatorname{diag}(\boldsymbol{s}_i)\gamma_i - \bar{\boldsymbol{h}}_i\boldsymbol{s}_i\boldsymbol{s}_i^\top\gamma_i\right)\right) \geq 0. \quad \text{(B.100)}$$

To proceed, recall $\bar{\mathcal{T}}_i$ is the set of all non-support indices per Definition 3.2, and $\mathcal{T}_i$ is the set of support indices. It follows from (B.66) that

$$\left|\boldsymbol{h}_i^\top \operatorname{diag}(\boldsymbol{s}_i)\gamma_i - \boldsymbol{h}_i^\top \boldsymbol{s}_i\boldsymbol{s}_i^\top\gamma_i - \sum_{t\in\mathcal{T}_i}(\boldsymbol{h}_{i1} - \boldsymbol{h}_{it})\boldsymbol{s}_{it}(\gamma_{i1} - \gamma_{it})\right| \leq 2\Gamma A\left((1-\boldsymbol{s}_{i1})^2 + Q_i\right). \quad \text{(B.101)}$$

Note that $\bar{\boldsymbol{W}}/\|\bar{\boldsymbol{W}}\|_F = \boldsymbol{W}/\|\boldsymbol{W}\|_F$. Hence, $\bar{\boldsymbol{W}} \in \mathcal{S}_\mu(\boldsymbol{W}^{\mathrm{mm}})$ and $\|\bar{\boldsymbol{W}}\|_F = \|\boldsymbol{W}^{\mathrm{mm}}\|_F$ which together with (B.66) implies that

$$\left|\bar{\boldsymbol{h}}_i^\top \operatorname{diag}(\boldsymbol{s}_i)\gamma_i - \bar{\boldsymbol{h}}_i^\top \boldsymbol{s}_i\boldsymbol{s}_i^\top\gamma_i - \sum_{t\in\mathcal{T}_i}(\bar{\boldsymbol{h}}_{i1} - \bar{\boldsymbol{h}}_{it})\boldsymbol{s}_{it}(\gamma_{i1} - \gamma_{it})\right| \leq 2\Gamma A\left((1-\boldsymbol{s}_{i1})^2 + Q_i\right). \quad \text{(B.102)}$$

Let us assume (without losing generality) $\alpha_i = 1$ for all $i \in [n]$. Using Lemma B.7 and its $\bar{R}_\mu$ constant, if $R_\pi > \bar{R}_\mu$, we just need to prove the statement for $0 < \pi < 1$. In this case, using (B.101)–(B.102), (B.100) is implied by the following stronger inequality

$$\sum_{i=1}^{n} -\ell_i' \cdot \left((1+\pi)\sum_{t\in\mathcal{T}_i}(\boldsymbol{h}_{i1} - \boldsymbol{h}_{it})\boldsymbol{s}_{it}(\gamma_{i1} - \gamma_{it}) - \sum_{t\in\mathcal{T}_i}(\bar{\boldsymbol{h}}_{i1} - \bar{\boldsymbol{h}}_{it})\boldsymbol{s}_{it}(\gamma_{i1} - \gamma_{it}) - 6\Gamma A\left((1-\boldsymbol{s}_{i1})^2 + Q_i\right)\right) \geq 0.$$

Note that since $\boldsymbol{h}_{i1} - \boldsymbol{h}_{it} = 1$ for $t \in \mathcal{T}_i$, using the above inequality, (B.100) is implied by the following inequality

$$\sum_{i=1}^{n} (-\ell_i') \cdot \left((1+\pi)\sum_{t\in\mathcal{T}_i}\boldsymbol{s}_{it}(\gamma_{i1} - \gamma_{it}) - \sum_{t\in\mathcal{T}_i}(\bar{\boldsymbol{h}}_{i1} - \bar{\boldsymbol{h}}_{it})\boldsymbol{s}_{it}(\gamma_{i1} - \gamma_{it}) - 6\Gamma A\left((1-\boldsymbol{s}_{i1})^2 + Q_i\right)\right) \geq 0. \quad \text{(B.103)}$$

Recall from Lemma B.7 where we define $\mathcal{I} \subseteq [n]$ be the index set that contains samples with support non-(locally) optimal tokens per Definition 3.2 and let $\bar{\mathcal{I}} = [n] - \mathcal{I}$. For $i \in \mathcal{I}$, $\mathcal{T}_i = \emptyset$. Therefore, (B.103) implies

$$\sum_{i\in\mathcal{I}} (-\ell_i') \cdot \left((1+\pi)\sum_{t\in\mathcal{T}_i}\boldsymbol{s}_{it}(\gamma_{i1} - \gamma_{it}) - \sum_{t\in\mathcal{T}_i}(\bar{\boldsymbol{h}}_{i1} - \bar{\boldsymbol{h}}_{it})\boldsymbol{s}_{it}(\gamma_{i1} - \gamma_{it})\right) - \sum_{i=1}^{n} (-\ell_i') \cdot 6\Gamma A\left((1-\boldsymbol{s}_{i1})^2 + Q_i\right) \geq 0.$$

In the following, we claim that

$$\begin{aligned}
&0.5\pi \sum_{i\in\mathcal{I}} \sum_{t\in\mathcal{T}_i} (-\ell_i') \cdot \boldsymbol{s}_{it}(\gamma_{i1} - \gamma_{it}) \\
&\geq \sum_{i=1}^{n} (-\ell_i') \cdot 6\Gamma A \left((1-\boldsymbol{s}_{i1})^2 + Q_i\right) \\
&= \sum_{i\in\mathcal{I}} (-\ell_i') \cdot 6\Gamma A \left((1-\boldsymbol{s}_{i1})^2 + Q_i\right) + \sum_{i\in\bar{\mathcal{I}}} (-\ell_i') \cdot 6\Gamma A \left(Q_i^2 + Q_i\right).
\end{aligned}$$

We will prove the claim by showing

$$0.25\pi \sum_{i\in\mathcal{I}} \sum_{t\in\mathcal{T}_i} (-\ell_i') \cdot \boldsymbol{s}_{it}(\gamma_{i1} - \gamma_{it}) \geq \sum_{i\in\mathcal{I}} (-\ell_i') \cdot 6\Gamma A((1-\boldsymbol{s}_{i1})^2 + Q_i), \qquad \text{and} \tag{B.104a}$$

$$0.25\pi \sum_{i\in\mathcal{I}} \sum_{t\in\mathcal{T}_i} (-\ell_i') \cdot \boldsymbol{s}_{it}(\gamma_{i1} - \gamma_{it}) \geq \sum_{i\in\bar{\mathcal{I}}} (-\ell_i') \cdot 6\Gamma A \left(Q_i^2 + Q_i\right). \tag{B.104b}$$

Proof of (B.104a) directly follows the earlier argument, namely, following (B.70), (B.73), and (B.74) which leads to the choice

$$R \geq \max(R_1, \hat{R}_2), \quad \text{where} \quad \hat{R}_2 = \frac{2}{3\delta\Theta} \log\left(\frac{48\,(2-\delta)\,T\Gamma A}{\pi\gamma_{\min}^{\text{gap}}}\right), \tag{B.105a}$$

where R_1 is defined in (B.71).

Additionally, (B.104b) follows (B.70), (B.77), (B.79), and (B.92), which leads to the choice

$$R \geq hat R_3 = \frac{1}{\delta\Theta} \log\left(\frac{96|\bar{\mathcal{I}}|T^2\Gamma A\ell'_{\max}}{(2-\delta)|\mathcal{I}|\pi\gamma_{\min}^{\text{gap}}\ell'_{\min}}\right). \tag{B.105b}$$

To conclude with the result, what remains is proving the comparison

$$\sum_{i\in\mathcal{I}} -\ell_i' \cdot \left(\sum_{t\in\mathcal{T}_i} (1+0.5\pi)\boldsymbol{s}_{it}(\gamma_{i1} - \gamma_{it}) - (\bar{\boldsymbol{h}}_{i1} - \bar{\boldsymbol{h}}_{it})\boldsymbol{s}_{it}(\gamma_{i1} - \gamma_{it})\right) \geq 0. \tag{B.106}$$

To proceed, we split the problem into two scenarios.

Scenario 1: $\|\bar{\boldsymbol{W}} - \boldsymbol{W}^{\text{mm}}\|_F < \epsilon = \frac{\pi}{4A\Theta}$ for some $\epsilon > 0$. In this scenario, for the first token (as an optimal token) and any $t \in \mathcal{T}_i$ and $i \in [n]$, we have

$$|\bar{\boldsymbol{h}}_{it} - \boldsymbol{h}_{it}| = |\boldsymbol{x}_{it}^\top(\bar{\boldsymbol{W}} - \boldsymbol{W}^{\text{mm}})\boldsymbol{z}_{it}| < A\Theta\epsilon = \frac{\pi}{4}.$$

Note that $\boldsymbol{h}_{i1} - \boldsymbol{h}_{it} = 1$ for $t \in \mathcal{T}_i$. Consequently, we obtain

$$\begin{aligned}\bar{\boldsymbol{h}}_{i1} - \bar{\boldsymbol{h}}_{it} = \boldsymbol{h}_{i1} - \boldsymbol{h}_{it} + \bar{\boldsymbol{h}}_{i1} - \boldsymbol{h}_{i1} - \bar{\boldsymbol{h}}_{it} + \boldsymbol{h}_{it} &< \boldsymbol{h}_{i1} - \boldsymbol{h}_{it} + 2A\Theta\epsilon \\ &= 1 + 0.5\pi.\end{aligned}$$

Similarly, $\bar{\boldsymbol{h}}_{i1} - \bar{\boldsymbol{h}}_{it} > 1 - 0.5\pi \geq 0.5$. Since all terms $\bar{\boldsymbol{h}}_{i1} - \bar{\boldsymbol{h}}_{it}, \boldsymbol{s}_{it}, \gamma_{i1} - \gamma_{it}$ in (B.106) are nonnegative and $(\bar{\boldsymbol{h}}_{i1} - \bar{\boldsymbol{h}}_{it})\boldsymbol{s}_{it}(\gamma_{i1} - \gamma_{it}) < (1 + 0.5\pi)\boldsymbol{s}_{it}(\gamma_{i1} - \gamma_{it})$, above implies the desired result in (B.106).

Scenario 2: $\|\bar{\boldsymbol{W}} - \boldsymbol{W}^{\mathrm{mm}}\|_F \geq \epsilon = \frac{\pi}{4A\Theta}$. Since $\bar{\boldsymbol{W}}$ is not (locally) max-margin, in this scenario, for some $i \in [n]$, $\nu = \nu(\epsilon) > 0$, and $\tau \in \mathcal{T}_i$, we have that

$$\bar{\boldsymbol{h}}_{i1} - \bar{\boldsymbol{h}}_{i\tau} \leq 1 - 2\nu.$$

Here $\tau = \arg\max_{\tau \in \mathcal{T}_i} \boldsymbol{x}_{i\tau}^\top \bar{\boldsymbol{W}} \boldsymbol{z}_i$ denotes the nearest point to $\bar{\boldsymbol{h}}_{i1}$ (along the $\bar{\boldsymbol{W}}$ direction). Note that a non-neighbor $t \in \bar{\mathcal{T}}_i$ cannot be nearest because $\bar{\boldsymbol{W}} \in \mathtt{cone}_\mu(\boldsymbol{p}^{\mathrm{mm}})$ and (B.56) holds. Recall that $\boldsymbol{s}_i = \mathbb{S}(\bar{R}\bar{\boldsymbol{h}}_i)$ where $\bar{R} = \|\boldsymbol{W}\|_F\Theta$. To proceed, let $\underline{\bar{\boldsymbol{h}}}_i := \min_{t \in \mathcal{T}_i} \bar{\boldsymbol{h}}_{i1} - \bar{\boldsymbol{h}}_{it}$,

$$\mathcal{I}_1 := \left\{ i \in \mathcal{I} : \underline{\bar{\boldsymbol{h}}}_i \leq 1 - 2\nu \right\}, \qquad \mathcal{I} - \mathcal{I}_1 := \left\{ i \in \mathcal{I} : 1 - 2\nu < \underline{\bar{\boldsymbol{h}}}_i \right\}.$$

For all $i \in \mathcal{I} - \mathcal{I}_1$,

$$\begin{aligned}\sum_{t \in \mathcal{T}_i} (\bar{\boldsymbol{h}}_{i1} - \bar{\boldsymbol{h}}_{it})\boldsymbol{s}_{it}(\gamma_{i1} - \gamma_{it}) - (1 + 0.5\pi) \sum_{t \in \mathcal{T}_i} \boldsymbol{s}_{it}(\gamma_{i1} - \gamma_{it}) \\ \leq \sum_{t \in \mathcal{T}_i,\ \bar{\boldsymbol{h}}_{i1} - \bar{\boldsymbol{h}}_{it} \geq 1 + \frac{\pi}{2}} (\bar{\boldsymbol{h}}_{i1} - \bar{\boldsymbol{h}}_{it})\boldsymbol{s}_{it}(\gamma_{i1} - \gamma_{it}) \\ \leq 2A\Gamma \sum_{t \in \mathcal{T}_i,\ \bar{\boldsymbol{h}}_{i1} - \bar{\boldsymbol{h}}_{it} \geq 1 + \frac{\pi}{2}} \boldsymbol{s}_{it} \\ \leq 2A\Gamma T e^{-\bar{R}(1 + \frac{\pi}{2})}.\end{aligned} \tag{B.107}$$

Here, the first inequality follows since $\boldsymbol{s}_{it} > 0$ and $\gamma_{i1} - \gamma_{it}$ are non-negative; second inequality is due to the definition of global scalar $\Gamma = \sup_{i \in [n], t, \tau \in [T]} |\gamma_{it} - \gamma_{i\tau}|$, (B.53c), and since $\bar{\boldsymbol{W}} \in \mathcal{S}_\mu(\boldsymbol{W}^{\mathrm{mm}})$ with $\|\bar{\boldsymbol{W}}\|_F = \|\boldsymbol{W}^{\mathrm{mm}}\|_F$ and $\max_{t \in [T]} |\bar{\boldsymbol{h}}_t| = \max_{t \in [T]} |(\boldsymbol{X}\bar{\boldsymbol{W}}\boldsymbol{z})_t| \leq A$.

For all $i \in \mathcal{I}_1$, split the tokens into two groups: Let $\mathcal{N}_i$ be the group of tokens obeying $\bar{\boldsymbol{h}}_{i1} - \bar{\boldsymbol{h}}_{it} \leq 1 - \nu$ and $\mathcal{T}_i - \mathcal{N}_i$ be the rest of the neighbors. Observe that

$$\frac{\sum_{t \in \mathcal{T}_i - \mathcal{N}_i} \boldsymbol{s}_{it}}{\sum_{t \in \mathcal{T}_i} \boldsymbol{s}_{it}} \leq T \frac{e^{\nu\bar{R}}}{e^{2\nu\bar{R}}} = T e^{-\bar{R}\nu}.$$

Using $|\bar{\boldsymbol{h}}_{i1} - \bar{\boldsymbol{h}}_{it}| \leq 2A$ and $\gamma_{\min}^{\mathrm{gap}} = \min_{i\in[n]} \gamma_i^{\mathrm{gap}} = \min_{i\in[n]}(\gamma_{i1} - \max_{t\in\mathcal{T}_i} \gamma_{it})$, observe that

$$\sum_{t\in\mathcal{T}_i-\mathcal{N}_i} (\bar{\boldsymbol{h}}_{i1} - \bar{\boldsymbol{h}}_{it})\boldsymbol{s}_{it}(\gamma_{i1} - \gamma_{it}) \leq \frac{2\Gamma AT e^{-\bar{R}\nu}}{\gamma_{\min}^{\mathrm{gap}}} \sum_{t\in\mathcal{T}_i} \boldsymbol{s}_{it}(\gamma_{i1} - \gamma_{it}).$$

Thus,

$$\begin{aligned}
\sum_{t\in\mathcal{T}_i}(\bar{\boldsymbol{h}}_{i1} - \bar{\boldsymbol{h}}_{it})\boldsymbol{s}_{it}(\gamma_{i1} - \gamma_{it}) &= \sum_{t\in\mathcal{N}_i}(\bar{\boldsymbol{h}}_{i1} - \bar{\boldsymbol{h}}_{it})\boldsymbol{s}_{it}(\gamma_{i1} - \gamma_{it}) + \sum_{t\in\mathcal{T}_i-\mathcal{N}_i} (\bar{\boldsymbol{h}}_{i1} - \bar{\boldsymbol{h}}_{it})\boldsymbol{s}_{it}(\gamma_{i1} - \gamma_{it}) \\
&\leq \sum_{t\in\mathcal{N}_i}(1-\nu)\boldsymbol{s}_{it}(\gamma_{i1} - \gamma_{it}) + \frac{2\Gamma AT e^{-\bar{R}\nu}}{\gamma_{\min}^{\mathrm{gap}}} \sum_{t\in\mathcal{T}_i} \boldsymbol{s}_{it}(\gamma_{i1} - \gamma_{it}) \\
&\leq \left(1 - \nu + \frac{2\Gamma AT e^{-\bar{R}\nu}}{\gamma_{\min}^{\mathrm{gap}}}\right) \sum_{t\in\mathcal{T}_i} \boldsymbol{s}_{it}(\gamma_{i1} - \gamma_{it}) \\
&\leq \left(1 + \frac{2\Gamma AT e^{-\bar{R}\nu}}{\gamma_{\min}^{\mathrm{gap}}}\right) \sum_{t\in\mathcal{T}_i} \boldsymbol{s}_{it}(\gamma_{i1} - \gamma_{it}).
\end{aligned}$$

Hence, choosing

$$R \geq \hat{R}_4 = \frac{1}{\nu\Theta} \log\left(\frac{8\Gamma AT}{\gamma_{\min}^{\mathrm{gap}}\pi}\right) \tag{B.108}$$

results in that

$$\begin{aligned}
&\sum_{t\in\mathcal{T}_i}(\bar{\boldsymbol{h}}_{i1} - \bar{\boldsymbol{h}}_{it})\boldsymbol{s}_{it}(\gamma_{i1} - \gamma_{it}) - \left(1 + \frac{\pi}{2}\right) \sum_{t\in\mathcal{T}_i} \boldsymbol{s}_{it}(\gamma_{i1} - \gamma_{it}) \\
&\leq \left(\frac{2\Gamma AT e^{-\bar{R}\nu}}{\gamma_{\min}^{\mathrm{gap}}} - \frac{\pi}{2}\right) \sum_{t\in\mathcal{T}_i} \boldsymbol{s}_{it}(\gamma_{i1} - \gamma_{it}) \\
&\leq -\frac{\pi}{4} \sum_{t\in\mathcal{T}_i} \boldsymbol{s}_{it}(\gamma_{i1} - \gamma_{it}) \\
&\leq -\frac{\pi}{4T} \gamma_{\min}^{\mathrm{gap}} e^{-\bar{R}(1-2\nu)}.
\end{aligned} \tag{B.109}$$

Here, the last inequality follows from the fact that for all $i \in \mathcal{I}_1$,

$$\sum_{t\in\mathcal{T}_i} \boldsymbol{s}_{it} \geq \max_{t\in\mathcal{T}_i} \boldsymbol{s}_{it} \geq \frac{e^{-\bar{R}(1-2\nu)}}{\sum_{t=1}^{T} e^{-\bar{R}(\bar{\boldsymbol{h}}_{i1} - \bar{\boldsymbol{h}}_{it})}} \geq \frac{e^{-\bar{R}(1-2\nu)}}{T},$$

where $\boldsymbol{s}_i = \mathbb{S}(\bar{R}\bar{\boldsymbol{h}}_i)$ and $\bar{R} = \|\boldsymbol{W}\|_F\Theta$.

From Assumption 3.1, let $-\ell'_{\min}$ and $-\ell'_{\max}$ be the min and max values negative loss derivative admits over the ball $[-A, A]$, respectively. It follows from (B.107) and (B.109)

that

$$
\begin{aligned}
-\sum_{i\in\mathcal{I}}\ell_i'\cdot\left(\sum_{t\in\mathcal{T}_i}(\bar{\boldsymbol{h}}_{i1}-\bar{\boldsymbol{h}}_{it})\boldsymbol{s}_{it}(\gamma_{i1}-\gamma_{it})-\sum_{t\in\mathcal{T}_i}(1+0.5\pi)\boldsymbol{s}_{it}(\gamma_{i1}-\gamma_{it})\right)&\\
\leq(-\ell'_{\max})|\mathcal{I}|2A\Gamma Te^{-\bar{R}(1+\frac{\pi}{2})}-\frac{(-\ell'_{\min})}{T}\cdot\frac{\pi\gamma_{\min}^{\mathrm{gap}}}{4}e^{-\bar{R}(1-2\nu)}&\\
\leq 0.&
\end{aligned}
$$

Here, the last inequality is obtained by using

$$
R\geq\hat{R}_5=\frac{1}{(2\nu+\pi/2)\Theta}\log\left(\frac{8|\mathcal{I}|\Gamma AT^2\ell'_{\max}}{\ell'_{\min}\gamma_{\min}^{\mathrm{gap}}\pi}\right),\tag{B.110}
$$

Combing with (B.108), (B.106) is guaranteed by choosing

$$
R\geq\max\left\{\hat{R}_4,\hat{R}_5\right\},
$$

where $\nu=\nu(\frac{\pi}{4A\Theta})$ depends only on π and global problem variables.

Combining this with the prior R choice (B.105), (B.108), and (B.110), i.e., $\max\{R_1,\hat{R}_2,\ldots,\hat{R}_5\}$, gives

$$
R_\pi=\frac{2}{\min(\delta,\nu,\pi)\Theta}\log\left(\frac{C_0AT^2}{\pi}\cdot\frac{\Gamma}{\gamma_{\min}^{\mathrm{gap}}}\cdot\frac{\ell'_{\max}}{\ell'_{\min}}\cdot\frac{|\bar{\mathcal{I}}|}{|\mathcal{I}|}\right)\quad\text{with}\quad C_0\geq 96.\tag{B.111}
$$

Here, similar to (B.98), we assume that $|\bar{\mathcal{I}}|/|\mathcal{I}|$ is greater than or equal to 1 to simplify the expression. Otherwise, one can replace $|\bar{\mathcal{I}}|/|\mathcal{I}|$ with 1 in (B.111). ■

B.4.1 Proof of Theorem 3.5

Proof. The proof of this theorem follows the proof of [186, Theorem 3]. Let us denote the initialization lower bound as $R_\mu^0:=R$, where R is given in the Theorem 3.5's statement. Consider an arbitrary value of $\epsilon\in(0,\mu/2)$ and let $1/(1+\pi)=1-\epsilon$. We additionally denote $R_\epsilon\leftarrow R_\pi\vee 1/2$ where R_π was defined in Lemma B.8. At initialization $\boldsymbol{W}(0)$, we set $\epsilon=\mu/2$ to obtain $R_\mu^0=R_{\mu/2}$, and provide the proof in four steps:

Step 1: There are no stationary points within $\mathcal{C}_{\mu,R_\mu^0}(\boldsymbol{W}^{\mathrm{mm}})$. We begin by proving that there are no stationary points within $\mathcal{C}_{\mu,R_\mu^0}(\boldsymbol{W}^{\mathrm{mm}})$. Let $(\mathcal{T}_i)_{i=1}^n$ denote the sets of support indices as defined in Definition 3.2. We define $\bar{\mathcal{T}}_i=[T]-\mathcal{T}_i-\{\alpha_i\}$ as the tokens that are non-support indices. Additionally, let μ be defined as in (B.53). Then, since $R_\mu^0\geq\bar{R}_\mu$ per Lemma B.8, we can apply Lemma B.7 to find that: For all $\boldsymbol{V},\boldsymbol{W}\in\mathcal{S}_\mu(\boldsymbol{W}^{\mathrm{mm}})$ with

$\|\boldsymbol{W}\|_F \neq 0$ and $\|\boldsymbol{W}\|_F \geq R_\mu^0$, we have that $-\langle \boldsymbol{V}, \nabla\mathcal{L}(\boldsymbol{W})\rangle$ is strictly positive.

Step 2: It follows from Lemma B.8 that, there exists $R_\epsilon \geq \bar{R}_\mu \vee 1/2$ such that all $\boldsymbol{W} \in \mathcal{C}_{\mu,R_\epsilon}(\boldsymbol{W}^{\mathrm{mm}})$ satisfy

$$\left\langle -\nabla\mathcal{L}(\boldsymbol{W}), \frac{\boldsymbol{W}^{\mathrm{mm}}}{\|\boldsymbol{W}^{\mathrm{mm}}\|_F} \right\rangle \geq (1-\epsilon) \left\langle -\nabla\mathcal{L}(\boldsymbol{W}), \frac{\boldsymbol{W}}{\|\boldsymbol{W}\|_F} \right\rangle. \tag{B.112}$$

The argument below applies to a general $\epsilon \in (0, \mu/2)$. However, at initialization $\boldsymbol{W}(0)$, we set $\epsilon = \mu/2$ and, recalling above, initialization lower bound was defined as $R_\mu^0 := R_{\mu/2}$. To proceed, for any $\epsilon \in (0, \mu/2)$, we will show that after gradient descent enters the conic set $\mathcal{C}_{\mu,R_\epsilon}(\boldsymbol{W}^{\mathrm{mm}})$ for the first time, it will never leave the set. Let t_ϵ be the first time gradient descent enters $\mathcal{C}_{\mu,R_\epsilon}(\boldsymbol{W}^{\mathrm{mm}})$. In **Step 4**, we will prove that such t_ϵ is guaranteed to exist. Additionally, for $\epsilon \leftarrow \mu/2$, note that $t_\epsilon = 0$ i.e. the point of initialization.

Step 3: Updates remain inside the cone $\mathcal{C}_{\mu,R_\epsilon}(\boldsymbol{W}^{\mathrm{mm}})$. By leveraging the results from **Step 1** and **Step 2**, we demonstrate that the gradient iterates, with an appropriate constant step size, starting from $\boldsymbol{W}(k_\epsilon) \in \mathcal{C}_{\mu,R_\epsilon}(\boldsymbol{W}^{\mathrm{mm}})$, remain within this cone.

We proceed by induction. Suppose that the claim holds up to iteration $k \geq k_\epsilon$. This implies that $\boldsymbol{W}(k) \in \mathcal{C}_{\mu,R_\epsilon}(\boldsymbol{W}^{\mathrm{mm}})$. Hence, recalling cone definition, there exists scalar $\mu = \mu(\boldsymbol{\alpha}) \in (0,1)$ and R such that $\|\boldsymbol{W}(k)\|_F \geq R$, and

$$\left\langle \frac{\boldsymbol{W}(k)}{\|\boldsymbol{W}(k)\|_F}, \frac{\boldsymbol{W}^{\mathrm{mm}}}{\|\boldsymbol{W}^{\mathrm{mm}}\|_F} \right\rangle \geq 1-\mu.$$

For all $k \geq 1$, let

$$\rho(k) := -\frac{1}{1-\epsilon} \left\langle \nabla\mathcal{L}(\boldsymbol{W}(k)), \frac{\boldsymbol{W}^{\mathrm{mm}}}{\|\boldsymbol{W}^{\mathrm{mm}}\|_F} \right\rangle. \tag{B.113}$$

Note that $\rho(k) > 0$ due to **Step 1**. This together with the gradient descent update rule gives

$$\begin{aligned}
\left\langle \frac{\boldsymbol{W}(k+1)}{\|\boldsymbol{W}(k)\|_F}, \frac{\boldsymbol{W}^{\mathrm{mm}}}{\|\boldsymbol{W}^{\mathrm{mm}}\|_F} \right\rangle &= \left\langle \frac{\boldsymbol{W}(k)}{\|\boldsymbol{W}(k)\|_F} - \frac{\eta}{\|\boldsymbol{W}(k)\|_F}\nabla\mathcal{L}(\boldsymbol{W}(k)), \frac{\boldsymbol{W}^{\mathrm{mm}}}{\|\boldsymbol{W}^{\mathrm{mm}}\|_F} \right\rangle \\
&\geq 1-\mu - \frac{\eta}{\|\boldsymbol{W}(k)\|_F} \left\langle \nabla\mathcal{L}(\boldsymbol{W}(k)), \frac{\boldsymbol{W}^{\mathrm{mm}}}{\|\boldsymbol{W}^{\mathrm{mm}}\|_F} \right\rangle \qquad \text{(B.114a)} \\
&\geq 1-\mu + \frac{\eta\rho(k)(1-\epsilon)}{\|\boldsymbol{W}(k)\|_F}.
\end{aligned}$$

Note that from Lemma B.7, we have $\langle \nabla\mathcal{L}(\boldsymbol{W}(k)), \boldsymbol{W}(k)\rangle < 0$ which implies that $\|\boldsymbol{W}(k+1)\|_F \geq \|\boldsymbol{W}(k)\|_F$. This together with R_ϵ definition and $\|\boldsymbol{W}(k)\|_F \geq 1/2$ implies

that

$$\begin{aligned}\|\boldsymbol{W}(k+1)\|_F &\le \frac{1}{2\|\boldsymbol{W}(k)\|_F}\left(\|\boldsymbol{W}(k+1)\|_F^2+\|\boldsymbol{W}(k)\|_F^2\right)\\ &= \frac{1}{2\|\boldsymbol{W}(k)\|_F}\left(2\|\boldsymbol{W}(k)\|_F^2-2\eta\left\langle\nabla\mathcal{L}(\boldsymbol{W}(k)),\boldsymbol{W}(k)\right\rangle+\eta^2\|\nabla\mathcal{L}(\boldsymbol{W}(k))\|_F^2\right)\\ &\le \|\boldsymbol{W}(k)\|_F-\frac{\eta}{\|\boldsymbol{W}(k)\|_F}\left\langle\nabla\mathcal{L}(\boldsymbol{W}(k)),\boldsymbol{W}(k)\right\rangle+\eta^2\|\nabla\mathcal{L}(\boldsymbol{W}(k))\|_F^2,\end{aligned}$$

which gives

$$\begin{aligned}\frac{\|\boldsymbol{W}(k+1)\|_F}{\|\boldsymbol{W}(k)\|_F} &\le 1-\frac{\eta}{\|\boldsymbol{W}(k)\|_F}\left\langle\nabla\mathcal{L}(\boldsymbol{W}(k)),\frac{\boldsymbol{W}(k)}{\|\boldsymbol{W}(k)\|_F}\right\rangle+\eta^2\frac{\|\nabla\mathcal{L}(\boldsymbol{W}(k))\|^2}{\|\boldsymbol{W}(k)\|_F}\\ &\le 1-\frac{\eta}{(1-\epsilon)\|\boldsymbol{W}(k)\|_F}\left\langle\nabla\mathcal{L}(\boldsymbol{W}(k)),\frac{\boldsymbol{W}^{\mathrm{mm}}}{\|\boldsymbol{W}^{\mathrm{mm}}\|_F}\right\rangle+\eta^2\frac{\|\nabla\mathcal{L}(\boldsymbol{W}(k))\|^2}{\|\boldsymbol{W}(k)\|_F}\\ &\le 1+\frac{\eta\rho(k)}{\|\boldsymbol{W}(k)\|_F}+\frac{\eta^2\|\nabla\mathcal{L}(\boldsymbol{W}(k))\|^2}{\|\boldsymbol{W}(k)\|_F}=:C_1(\rho(k),\eta).\end{aligned}\tag{B.114b}$$

Here, the second inequality follows from (B.112) and (B.113).

Now, it follows from (B.114a) and (B.114b) that

$$\begin{aligned}&\left\langle\frac{\boldsymbol{W}(k+1)}{\|\boldsymbol{W}(k+1)\|},\frac{\boldsymbol{W}^{\mathrm{mm}}}{\|\boldsymbol{W}^{\mathrm{mm}}\|}\right\rangle\\ &\ge \frac{1}{C_1(\rho(k),\eta)}\left(1-\mu+\frac{\eta\rho(k)(1-\epsilon)}{\|\boldsymbol{W}(k)\|_F}\right)\\ &= 1-\mu+\frac{1}{C_1(\rho(k),\eta)}\left((1-\mu)(1-C_1(\rho(k),\eta))+\frac{\eta\rho(k)(1-\epsilon)}{\|\boldsymbol{W}(k)\|_F}\right)\\ &= 1-\mu+\frac{\eta}{C_1(\rho(k),\eta)}\left((\mu-1)(\frac{\rho(k)}{\|\boldsymbol{W}(k)\|_F}+\frac{\eta\|\nabla\mathcal{L}(\boldsymbol{W}(k))\|^2}{\|\boldsymbol{W}(k)\|_F})+\frac{\rho(k)(1-\epsilon)}{\|\boldsymbol{W}(k)\|_F}\right)\\ &= 1-\mu+\frac{\eta}{C_1(\rho(k),\eta)}\left(\frac{\rho(k)(\mu-\epsilon)}{\|\boldsymbol{W}(k)\|_F}-\eta(1-\mu)\frac{\|\nabla\mathcal{L}(\boldsymbol{W}(k))\|^2}{\|\boldsymbol{W}(k)\|_F}\right)\\ &\ge 1-\mu,\end{aligned}\tag{B.115}$$

where the last inequality uses our choice of stepsize $\eta \le 1/L_W$ in Theorem 3.5's statement. Specifically, we need η to be small to ensure the last inequality. We will guarantee this by choosing a proper R_ϵ in Lemma B.8 as follows.

Note that using Lemma B.7, and (B.63c), we have

$$\frac{\rho(k)}{\|\nabla\mathcal{L}(\boldsymbol{W}(k))\|_F} = -\frac{1}{1-\epsilon}\left\langle \frac{\nabla\mathcal{L}(\boldsymbol{W}(k))}{\|\nabla\mathcal{L}(\boldsymbol{W}(k))\|_F}, \frac{\boldsymbol{W}^{\mathrm{mm}}}{\|\boldsymbol{W}^{\mathrm{mm}}\|_F}\right\rangle \geq \frac{c\Theta}{(1-\epsilon)\hat{C}A},$$
$$\frac{1}{\|\nabla\mathcal{L}(\boldsymbol{W}(k))\|_F} \geq \frac{1}{A\hat{C}\cdot\frac{1}{n}\sum_{i=1}^{n}(1-\boldsymbol{s}_{i\alpha_i})} \geq \frac{1}{A\hat{C}T}e^{\frac{R_\mu^0\Theta\delta}{2}}$$

for some dataset-dependent positive constants C, $\hat{C}$, and c in (B.60); and A, Θ defined in (B.53).

Hence, using our choice of $\epsilon \in (0, \mu/2)$ (see **Step 2**), to ensure the last inequality in (B.115), we have

$$\begin{aligned}\eta \leq \frac{\mu c\Theta e^{\frac{R_\mu^0\Theta\delta}{2}}}{2(1-\mu)\left(1-\frac{\mu}{2}\right)\hat{C}^2A^2T} &\leq \frac{(\mu-\epsilon)c\Theta e^{\frac{R_\mu^0\Theta\delta}{2}}}{(1-\mu)(1-\epsilon)\hat{C}^2A^2T} \\ &\leq \frac{(\mu-\epsilon)\rho(k)}{(1-\mu)\|\nabla\mathcal{L}(\boldsymbol{W}(k))\|_F^2}. \end{aligned} \tag{B.116}$$

We will demonstrate that the choice of η in (B.116) does indeed meet our step size condition as stated in the theorem, i.e., $\eta \leq 1/L_{\boldsymbol{W}}$. Recall that $1/(1+\pi) = 1-\epsilon$, which implies that $\pi = \epsilon/(1-\epsilon)$. Combining this with (B.111), we obtain:

$$R_\pi \geq \frac{2}{\min(\delta,\nu,\pi)\Theta}\log\left(\frac{D_0}{\pi}\right) \tag{B.117}$$

which implies that

$$R_\epsilon \geq \frac{2}{\min(\delta,\nu,\pi)\Theta}\log\left(\frac{D_0(1-\epsilon)}{\epsilon}\right) \quad \text{with} \quad D_0 = \frac{C_0AT^2\Gamma\ell'_{\max}|\bar{\mathcal{I}}|}{\gamma_{\min}^{\mathrm{gap}}\ell'_{\min}|\mathcal{I}|}.$$

On the other hand, at the initialization, we have $\epsilon = \mu/2$ which implies that

$$R_\mu^0 \geq \frac{2}{\min(\delta,\nu,\pi)\Theta}\log\left(\frac{D_0(2-\mu)}{\mu}\right). \tag{B.118}$$

In the following, we will determine a lower bound on D_0 such that our step size condition in Theorem 3.5's statement, i.e., $\eta \leq 1/L_{\boldsymbol{W}}$, is satisfied. Note that for the choice of η in (B.116) to meet the condition $\eta \leq 1/L_{\boldsymbol{W}}$, the following condition must hold:

$$\frac{1}{L_{\boldsymbol{W}}} \leq \frac{\mu c\Theta e^{\frac{R_\mu^0\Theta\delta}{2}}}{2(1-\mu)\left(1-\frac{\mu}{2}\right)\hat{C}^2A^2T} \tag{B.119}$$

which implies that

$$R_\mu^0 \geq \frac{2}{\delta\Theta}\log\left(\frac{D_1(1-\mu/2)}{\mu/2}\right), \quad \text{where} \quad D_1 = \frac{(1-\mu)}{L_{\boldsymbol{W}}}\frac{A^2\hat{C}^2T}{\Theta c}.$$

Hence, using (B.118) and since at the initialization, we have $\epsilon = \mu/2$, we need $D_0 \geq D_1$ which implies that

$$C_0 \geq \frac{(1-\mu)A\hat{C}^2\gamma_{\min}^{\text{gap}}\ell'_{\min}|\mathcal{I}|}{L_{\boldsymbol{W}}\Theta cT\Gamma\ell'_{\max}|\bar{\mathcal{I}}|}. \tag{B.120}$$

Therefore, with this lower bound on C_0, the step size bound in (B.116) is sufficiently large to ensure that $\eta \leq 1/L_{\boldsymbol{W}}$ guarantees (B.115).

Hence, it follows from (B.115) that $\boldsymbol{W}(k+1) \in \mathcal{C}_{\mu,R_\epsilon}(\boldsymbol{W}^{\text{mm}})$.

Step 4: The correlation of $\boldsymbol{W}(k)$ and $\boldsymbol{W}^{\text{mm}}$ increases over k. From Step 3, we have that all iterates remain within the initial conic set i.e. $\boldsymbol{W}(k) \in \mathcal{C}_{\mu,R_\mu^0}(\boldsymbol{W}^{\text{mm}})$ for all $k \geq 0$. Note that it follows from Lemma B.7 that $\langle \nabla\mathcal{L}(\boldsymbol{W}), \boldsymbol{W}^{\text{mm}}/\|\boldsymbol{W}^{\text{mm}}\|_F\rangle < 0$, for any finite $\boldsymbol{W} \in \mathcal{C}_{\mu,R_\mu^0}(\boldsymbol{W}^{\text{mm}})$. Hence, there are no finite critical points $\boldsymbol{W} \in \mathcal{C}_{\mu,R_\mu^0}(\boldsymbol{W}^{\text{mm}})$, for which $\nabla\mathcal{L}(\boldsymbol{W}) = 0$. Now, based on Lemma B.3, which guarantees that $\nabla\mathcal{L}(\boldsymbol{W}(k)) \to 0$, this implies that $\|\boldsymbol{W}(t)\|_F \to \infty$. Consequently, for any choice of $\epsilon \in (0, \mu/2)$ there is an iteration k_ϵ such that, for all $k \geq k_\epsilon$, $\boldsymbol{W}(k) \in \mathcal{C}_{\mu,R_\epsilon}(\boldsymbol{W}^{\text{mm}})$. Once within $\mathcal{C}_{\mu,R_\epsilon}(\boldsymbol{W}^{\text{mm}})$, multiplying both sides (B.112) by the stepsize η and using the gradient descent update, we get

$$\begin{aligned}
&\left\langle \boldsymbol{W}(k+1) - \boldsymbol{W}(k), \frac{\boldsymbol{W}^{\text{mm}}}{\|\boldsymbol{W}^{\text{mm}}\|_F} \right\rangle \\
&\geq (1-\epsilon)\left\langle \boldsymbol{W}(k+1) - \boldsymbol{W}(k), \frac{\boldsymbol{W}(k)}{\|\boldsymbol{W}(k)\|_F} \right\rangle \\
&= \frac{(1-\epsilon)}{2\|\boldsymbol{W}(k)\|_F}\left(\|\boldsymbol{W}(k+1)\|_F^2 - \|\boldsymbol{W}(k)\|_F^2 - \|\boldsymbol{W}(k+1) - \boldsymbol{W}(k)\|_F^2\right) \\
&\geq (1-\epsilon)\left(\frac{1}{2\|\boldsymbol{W}(k)\|_F}\left(\|\boldsymbol{W}(k+1)\|_F^2 - \|\boldsymbol{W}(k)\|_F^2\right) - \|\boldsymbol{W}(k+1) - \boldsymbol{W}(k)\|_F^2\right) \\
&\geq (1-\epsilon)\left(\|\boldsymbol{W}(k+1)\|_F - \|\boldsymbol{W}(k)\|_F - \|\boldsymbol{W}(k+1) - \boldsymbol{W}(k)\|_F^2\right) \\
&\geq (1-\epsilon)\Big(\|\boldsymbol{W}(k+1)\|_F - \|\boldsymbol{W}(k)\|_F - 2\eta\left(\mathcal{L}(\boldsymbol{W}(k)) - \mathcal{L}(\boldsymbol{W}(k+1))\right)\Big).
\end{aligned}$$

Here, the second inequality is obtained from $\|\boldsymbol{W}(k)\|_F \geq 1/2$; the third inequality follows since for any $a, b > 0$, we have $(a^2 - b^2)/(2b) - (a - b) \geq 0$; and the last inequality uses Lemma B.3.

Summing the above inequality over $k \geq k_\epsilon$ gives

$$\left\langle \frac{\boldsymbol{W}(k)}{\|\boldsymbol{W}(k)\|_F}, \frac{\boldsymbol{W}^{\mathrm{mm}}}{\|\boldsymbol{W}^{\mathrm{mm}}\|_F} \right\rangle \geq 1 - \epsilon + \frac{C(\epsilon, \eta)}{\|\boldsymbol{W}(k)\|_F}, \qquad \boldsymbol{W}(k) \in \mathcal{C}_{\mu, R_\epsilon}(\boldsymbol{W}^{\mathrm{mm}}),$$

where $\mathcal{L}_\star \leq \mathcal{L}\left(\boldsymbol{W}\left(k\right)\right)$ for all $k \geq 0$, and

$$C(\epsilon, \eta) = \left\langle \boldsymbol{W}(k_\epsilon), \frac{\boldsymbol{W}^{\mathrm{mm}}}{\|\boldsymbol{W}^{\mathrm{mm}}\|_F} \right\rangle - (1-\epsilon)\|\boldsymbol{W}(k_\epsilon)\|_F - 2\eta(1-\epsilon)(\mathcal{L}(\boldsymbol{W}(k_\epsilon)) - \mathcal{L}_\star).$$

Consequently, as $k \to \infty$

$$\liminf_{k\to\infty} \left\langle \frac{\boldsymbol{W}(k)}{\|\boldsymbol{W}(k)\|_F}, \frac{\boldsymbol{W}^{\mathrm{mm}}}{\|\boldsymbol{W}^{\mathrm{mm}}\|_F} \right\rangle \geq 1 - \epsilon, \qquad \boldsymbol{W}(k) \in \mathcal{C}_{\mu, R_\epsilon}(\boldsymbol{W}^{\mathrm{mm}}).$$

Since $\epsilon \in (0, \mu/2)$ is arbitrary, we get $\boldsymbol{W}(k)/\|\boldsymbol{W}(k)\|_F \to \boldsymbol{W}^{\mathrm{mm}}/\|\boldsymbol{W}^{\mathrm{mm}}\|_F$. ∎

B.5 Convergence of Regularization Path for Sequence-to-Sequence Setting

In this section, we provide proofs for the regularization path analysis. We first provide a more general formulation of the optimization problem that allows for regressing multiple token outputs. To distinguish from (W-ERM)&(KQ-ERM), let us call this more general version Sequence Empirical Risk Minimization (SERM).

Problem definition: Rather than a single input sequence $\boldsymbol{X}$, let us allow for two input sequences $\boldsymbol{X} \in \mathbb{R}^{T\times d}$ and $\boldsymbol{Z} \in \mathbb{R}^{K\times d}$ with $(\boldsymbol{x}_t)_{t=1}^T$ and $(\boldsymbol{z}_k)_{k=1}^K$. The cross-attention admits $\boldsymbol{X}, \boldsymbol{Z}$ and outputs K tokens. We will also allow for K separate prediction heads $(h_k)_{k=1}^K$ for individual cross attention outputs which strictly generalizes the setting where we used single prediction head $h(\cdot)$. Denote the training labels associated to each token as $\boldsymbol{y} = (y_{ik})_{i=1}^K$. Given n samples $(\boldsymbol{y}_i, \boldsymbol{X}_i, \boldsymbol{Z}_i)_{i=1}^n$, for a decreasing loss function $\ell(\cdot)$, minimize the empirical risk by the prediction of first attention output either univariate ($\boldsymbol{W} \in \mathbb{R}^{d\times d}$) or bivariate ($\boldsymbol{W}_k, \boldsymbol{W}_q \in \mathbb{R}^{d\times m}$) fashion:

$$\mathcal{L}(\boldsymbol{W}) = \frac{1}{n}\sum_{i=1}^n \sum_{k=1}^K \ell(y_{ik} \cdot h_k(\boldsymbol{X}_i^\top \mathbb{S}(\boldsymbol{X}_i \boldsymbol{W} \boldsymbol{z}_{ik}))), \tag{SERM-W}$$

$$\mathcal{L}(\boldsymbol{W}_k, \boldsymbol{W}_q) = \frac{1}{n}\sum_{i=1}^n \ell(y_{ik} \cdot h_k(\boldsymbol{X}_i^\top \mathbb{S}(\boldsymbol{X}_i \boldsymbol{W}_k \boldsymbol{W}_q^\top \boldsymbol{z}_{ik}))). \tag{SERM-KQ}$$

In order to recover the single-output self-attention model, we can simply set $K = 1$ and

$\boldsymbol{z}_{i1} \leftarrow \boldsymbol{x}_{i1}$ and $h_k \leftarrow h$. To proceed, we introduce the more general version of the (W-SVM) problem, which we refer to as *Sequential Cross-Attention SVM* to preserve consistent phrasing. Suppose $\boldsymbol{W}_k, \boldsymbol{W}_q \in \mathbb{R}^{d\times m}$ with $m \le d$ and let $\mathcal{R}_m$ denote the set of rank-m matrices in $\mathbb{R}^{d\times d}$. Given indices $\boldsymbol{\alpha} = (\alpha_{ik})_{ik=(1,1)}^{(n,K)}$, consider the SVM with $\diamond$-norm constraint

$$\boldsymbol{W}_{\diamond,\boldsymbol{\alpha}}^{\mathrm{mm}} \in \arg\min_{\boldsymbol{W}\in\mathcal{R}_m} \|\boldsymbol{W}\|_\diamond \quad \text{s.t.} \quad \min_{t\neq\alpha_{ik}} (\boldsymbol{x}_{i\alpha_{ik}} - \boldsymbol{x}_{it})^\top \boldsymbol{W}\boldsymbol{z}_{ik} \ge 1 \quad \forall \quad i\in[n], k\in[K]. \tag{SAtt-SVM}$$

When solution is non-unique, we denote the solution set by $\mathcal{W}^{\mathrm{mm}}(\boldsymbol{\alpha})$. In what follows, we denote $\boldsymbol{F}_{ikt} := \boldsymbol{x}_{it}\boldsymbol{z}_{ik}^\top$ and given $\boldsymbol{\alpha}$, we denote $\boldsymbol{x}_{ik}^{\mathrm{att}} = \boldsymbol{x}_{i\alpha_{ik}}$ and $\boldsymbol{F}_{ik}^{\boldsymbol{\alpha}} := \boldsymbol{x}_{i\alpha_{ik}}\boldsymbol{z}_{ik}^\top$. With this notation, we can equivalently write

$$\boldsymbol{W}_{\diamond,\boldsymbol{\alpha}}^{\mathrm{mm}} \in \arg\min_{\boldsymbol{W}\in\mathcal{R}_m} \|\boldsymbol{W}\|_\diamond \quad \text{s.t.} \quad \min_{t\neq\alpha_{ik}} \langle \boldsymbol{F}_{ik}^{\boldsymbol{\alpha}} - \boldsymbol{F}_{ikt}, \boldsymbol{W}\rangle \ge 1 \quad \forall\, i\in[n], k\in[K]. \tag{SAtt-SVM'}$$

Definition B.1 (Support Indices and Locally-Optimal Indices) *Fix token indices $\boldsymbol{\alpha} = (\alpha_{ik})_{ik=(1,1)}^{(n,k)}$ for which* (SAtt-SVM) *is feasible to obtain $\boldsymbol{W}_{\boldsymbol{\alpha}}^{mm} := \boldsymbol{W}_{\diamond,\boldsymbol{\alpha}}^{mm}$. Define token scores as*

$$\gamma_{ikt} = y_{ik}\cdot h_k(\boldsymbol{x}_{it}), \quad \boldsymbol{\gamma}_{ik}^{\boldsymbol{\alpha}} := \gamma_{ik\alpha_{ik}} = y_{ik}\cdot h_k(\boldsymbol{x}_{ik}^{att}).$$

Consider tokens $\mathcal{T}_{ik}\subset[T]$ such that $\langle \boldsymbol{F}_{ik}^{\boldsymbol{\alpha}} - \boldsymbol{F}_{ikt}, \boldsymbol{W}_{\boldsymbol{\alpha}}^{mm}\rangle = 1$ for all $t\in\mathcal{T}_{ik}$. $\mathcal{T}_{ik}$ is allowed to be an empty set. We refer to $\mathcal{T}_{ik}$ as support indices of $\boldsymbol{F}_{ik}^{\boldsymbol{\alpha}} = \boldsymbol{x}_{ik}^{att}\boldsymbol{z}_{ik}^\top$ and define its complement $\bar{\mathcal{T}}_{ik} = [T] - \mathcal{T}_{ik} - \{\alpha_{ik}\}$. Additionally, token indices $\boldsymbol{\alpha} = (\alpha_{ik})_{ik=(1,1)}^{(n,k)}$ are called locally-optimal if for all $i\in[n], k\in[K]$ and $t\in\mathcal{T}_{ik}$, token scores obey $\boldsymbol{\gamma}_{ik}^{\boldsymbol{\alpha}} > \gamma_{ikt}$. Associated $\boldsymbol{W}_{\boldsymbol{\alpha}}^{mm}$ is called a locally-optimal direction. Finally, let $opt_{ik} \in \arg\max_{t\in[T]}\gamma_{ikt}$ be the optimal indices and define the associated $\boldsymbol{W}^{mm}(opt)$ to be a globally-optimal direction.

Lemma B.9 (Mapping regularization path of $(\boldsymbol{W}_k, \boldsymbol{W}_q)$ to $\boldsymbol{W}$) *Let $\boldsymbol{W}_k, \boldsymbol{W}_q \in \mathbb{R}^{d\times m}$ and consider regularization path solutions of* (SERM-W) *and* (SERM-KQ)

$$\bar{\boldsymbol{W}}(R) \in \operatorname*{arg\,min}_{\boldsymbol{W}\in\mathcal{R}_m:\|\boldsymbol{W}\|_\star\le R} \mathcal{L}(\boldsymbol{W}) \tag{B.121}$$

$$\bar{\boldsymbol{W}}_k(R), \bar{\boldsymbol{W}}_q(R) \in \operatorname*{arg\,min}_{\|\boldsymbol{W}_k\|_F^2+\|\boldsymbol{W}_q\|_F^2\le 2R} \mathcal{L}(\boldsymbol{W}_k, \boldsymbol{W}_q). \tag{B.122}$$

For all $R\ge 0$, there is a one-to-one map between the set of solutions $\bar{\boldsymbol{W}}(R)$ of (B.121) *and $\bar{\boldsymbol{W}}_k(R)\bar{\boldsymbol{W}}_q(R)^\top$ of* (B.122).

Proof. To prove the mapping, first fix a $\bar{\boldsymbol{W}}(R)$ solution with rank m, set $\mathcal{L}_F = \mathcal{L}(\bar{\boldsymbol{W}}(R))$ and show the existence of $\boldsymbol{W}_k, \boldsymbol{W}_q$ with $\boldsymbol{W}_k\boldsymbol{W}_q^\top = \bar{\boldsymbol{W}}(R)$ feasible for (B.122) and

$\mathcal{L}(\boldsymbol{W}_k, \boldsymbol{W}_q) \leq \mathcal{L}_F$. Use the singular value decomposition $\bar{\boldsymbol{W}}(R) = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^\top$ with $\boldsymbol{\Sigma} \in \mathbb{R}^{m\times m}$ being diagonal matrix of singular values. Set $\boldsymbol{W}_k = \boldsymbol{U}\sqrt{\boldsymbol{\Sigma}}$ and $\boldsymbol{W}_q = \boldsymbol{V}\sqrt{\boldsymbol{\Sigma}}$. Observe that $\boldsymbol{W}_k\boldsymbol{W}_q^\top = \boldsymbol{W}$ and

$$\|\boldsymbol{W}_k\|_F^2 = \|\boldsymbol{W}_q\|_F^2 = \sum_{i=1}^{m} \sqrt{\boldsymbol{\Sigma}_{ii}}^2 = \|\bar{\boldsymbol{W}}(R)\|_\star \leq R.$$

Thus, $\boldsymbol{W}_k, \boldsymbol{W}_q$ achieves $\mathcal{L}(\boldsymbol{W}_k, \boldsymbol{W}_q) = \mathcal{L}_F$. Conversely, given $\bar{\boldsymbol{W}}_k(R), \bar{\boldsymbol{W}}_q(R)$ with $\mathcal{L}_\star = \mathcal{L}(\bar{\boldsymbol{W}}_k(R), \bar{\boldsymbol{W}}_q(R))$, $\boldsymbol{W} = \bar{\boldsymbol{W}}_k(R)\bar{\boldsymbol{W}}_q(R)^\top$ obeys $\mathcal{L}(\boldsymbol{W}) = \mathcal{L}_\star$ and, using the standard nuclear norm inequality, we have

$$\|\boldsymbol{W}\|_\star = \|\bar{\boldsymbol{W}}_k(R)\bar{\boldsymbol{W}}_q(R)^\top\|_\star \leq \frac{1}{2}(\|\bar{\boldsymbol{W}}_k(R)\|_F^2 + \|\bar{\boldsymbol{W}}_q(R)\|_F^2) = R.$$

This shows $\boldsymbol{W}$ is feasible for (B.121). Combining the two findings above, we find that optimal costs are equal ($\mathcal{L}_\star = \mathcal{L}_F$) and for any $(\bar{\boldsymbol{W}}_k(R), \bar{\boldsymbol{W}}_q(R))$ solution there exists a $\bar{\boldsymbol{W}}(R)$ solution and vice versa. ■

To proceed with our analysis, let us define the set of optimal solutions $\boldsymbol{W}_{\boldsymbol{\alpha}}^{\mathrm{mm}} := \boldsymbol{W}_{\diamond,\boldsymbol{\alpha}}^{\mathrm{mm}}$ to (SAtt-SVM). Let us denote this set by $\mathcal{W}_{\boldsymbol{\alpha}}^{\mathrm{mm}} := \mathcal{W}_{\diamond}^{\mathrm{mm}}(\boldsymbol{\alpha})$. Note that, if the $\diamond$-norm is not strongly-convex, $\mathcal{W}_{\boldsymbol{\alpha}}^{\mathrm{mm}}$ may not be singleton.

B.5.1 Local regularization path and proof of Theorem 3.7

We first recall *local regularization path* which solves the $\diamond$-norm-constrained problem over a conic slice $\mathcal{C}$, namely $\bar{\boldsymbol{W}}(R) = \min_{\|\boldsymbol{W}\|_\diamond \leq R, \boldsymbol{W}\in\mathcal{C}} \mathcal{L}(\boldsymbol{W})$. We will show that proper local RP directionally converge to locally-optimal directions. Setting the cone to be the rank-m manifold $\mathcal{R}_m$ (and more specifically the set of all matrices $\mathbb{R}^{d\times d}$), this will also establish the convergence of global RP to the globally-optimal direction.

Our cone definition $\texttt{cone}_\epsilon(\boldsymbol{\alpha})$ is induced by a token selection $\boldsymbol{\alpha} = (\alpha_{ik})_{ik=(1,1)}^{(n,k)}$ and has a simple interpretation: It prioritizes tokens with lower score than $\boldsymbol{\alpha}$ over tokens with high-score than $\boldsymbol{\alpha}$. This way lower score tokens create a barrier for $\boldsymbol{\alpha}$ and prevents optimization to move towards higher score tokens.

Definition B.2 (Low&High Score Tokens and Separating Cone) *Given $\alpha \in [T]$, input sequence $\boldsymbol{X}$ with label y, $h(\cdot) : \mathbb{R}^d \to \mathbb{R}$, and score $\gamma_t = y \cdot h(\boldsymbol{x}_t)$ for all $t \in [T]$, define the low and high score tokens as*

$$\mathtt{low}^\alpha(\boldsymbol{X}) = \left\{t \in [T] \;\middle|\; \gamma_t < \gamma_\alpha\right\}, \quad \mathtt{high}^\alpha(\boldsymbol{X}) = \left\{t \in [T] - \{\alpha\} \;\middle|\; \gamma_t \geq \gamma_\alpha\right\}.$$

For input $\boldsymbol{X}_{ik}$ and index α_{ik}, we use the shorthand notations $\texttt{low}^{\alpha}_{ik}, \texttt{high}^{\alpha}_{ik}$. Finally define $\texttt{cone}_{\epsilon}(\boldsymbol{\alpha})$ as

$$\texttt{cone}_{\epsilon}(\boldsymbol{\alpha}) = \left\{ \boldsymbol{W} \in \mathcal{R}_m \;\middle|\; \min_{i\in[n]} \max_{t\in \texttt{low}^{\alpha}_{ik}} \min_{\tau\in \texttt{high}^{\alpha}_{ik}} \langle \boldsymbol{F}_{ikt} - \boldsymbol{F}_{ik\tau}, \boldsymbol{W}\rangle \geq \epsilon \|\boldsymbol{W}\|_F \right\}. \tag{B.123}$$

Lemma B.10 *Consider the cone definition of* (B.123) *and suppose an SVM solution $\boldsymbol{W}^{mm}_{\boldsymbol{\alpha}}$ exists. If indices $\boldsymbol{\alpha}$ are locally-optimal, $\boldsymbol{W}^{mm}_{\boldsymbol{\alpha}} \in \texttt{cone}_{\epsilon}(\boldsymbol{\alpha})$ for all sufficiently small $\epsilon > 0$. Otherwise, $\boldsymbol{W}^{mm}_{\boldsymbol{\alpha}} \notin \texttt{cone}_{\epsilon}(\boldsymbol{\alpha})$ for all $\epsilon > 0$. Additionally, suppose optimal indices $\text{opt}_{ik} \in \arg\max_{t\in[T]} \gamma_{ikt}$ are unique. Then, $\texttt{cone}_{\epsilon}(opt) = \mathcal{R}_m$.*

Proof. Suppose $\boldsymbol{\alpha}$ is locally optimal. Observe that, thanks to local optimality, $\boldsymbol{W}^{\text{mm}}_{\boldsymbol{\alpha}}$ obeys

$$\min_{t\in\mathcal{T}_{ik}} \langle \boldsymbol{F}_{ikt}, \boldsymbol{W}^{\text{mm}}_{\boldsymbol{\alpha}}\rangle > \max_{\tau\notin\mathcal{T}_{ik}\cup\{\alpha_{ik}\}} \langle \boldsymbol{F}_{ik\tau}, \boldsymbol{W}^{\text{mm}}_{\boldsymbol{\alpha}}\rangle,$$

for all $i \in [n]$. Next, observe that $\mathcal{T}_{ik} \subseteq \texttt{low}^{\alpha}_{ik}$ and $\texttt{high}^{\alpha}_{ik} \subseteq \bar{\mathcal{T}}_{ik} = [T] - \mathcal{T}_{ik} - \{\alpha_{ik}\}$. Thus, the inequality (B.123) holds for small enough $\epsilon > 0$.

Conversely, suppose $\boldsymbol{\alpha}$ is not locally-optimal. Fix support index $t \in \mathcal{T}_{ik}$ with $t \in \texttt{high}^{\alpha}_{ik}$. Since $t \in \mathcal{T}_{ik}$, observe that

$$\langle \boldsymbol{F}_{ikt}, \boldsymbol{W}^{\text{mm}}_{\boldsymbol{\alpha}}\rangle \geq \max_{\tau\neq\alpha_{ik}} \langle \boldsymbol{F}_{ik\tau}, \boldsymbol{W}^{\text{mm}}_{\boldsymbol{\alpha}}\rangle.$$

In other words, for this $i \in [n]$, we found

$$\max_{\tau\in\texttt{low}^{\alpha}_{ik}} \langle \boldsymbol{F}_{ik\tau} - \boldsymbol{F}_{ikt}, \boldsymbol{W}^{\text{mm}}_{\boldsymbol{\alpha}}\rangle \leq 0,$$

violating (B.123) definition for any $\epsilon > 0$. To show the final claim, observe that, setting $\boldsymbol{\alpha} := \text{opt}$, we have that $\texttt{high}^{\alpha}_{ik} = \emptyset$ for all $i \in [n]$ as opt_{ik} are unique optimal indices. Thus, there is no constraint enforced on the cone definition in (B.123) making it equal to the rank-m manifold $\mathcal{R}_m$. ∎

Our main assumption regarding prediction head is a monotonicity condition which is a strict generalization of linearity: We ask for h to preserve the order of token scores under convex combinations.

Assumption B.1 (h preserves the top score) *The functions $(h_k)_{k=1}^K$ are L_h-Lipschitz in Euclidean distance. Given $\boldsymbol{\alpha} = (\alpha_{ik})^{(n,k)}_{ik=(1,1)}$, there exists a scalar $c := c_{\boldsymbol{\alpha}} > 0$ such*

that for all $i \in [n], k \in [K]$ the following holds: Consider any convex combination

$$\boldsymbol{x}(\boldsymbol{s}) = \sum_{t \in \texttt{low}_{ik}^{\alpha} \cup \{\alpha_{ik}\}} s_t \cdot \boldsymbol{x}_{it} \quad \textit{where} \quad \sum_{t \in \texttt{low}_{ik}^{\alpha} \cup \{\alpha_{ik}\}} s_t = 1,\ s_t \geq 0.$$

We have that $y_{ik} \cdot h_k(\boldsymbol{x}(\boldsymbol{s})) \leq \boldsymbol{\gamma}_{ik}^{\boldsymbol{\alpha}} - c(1 - s_{\alpha_{ik}})$ where $\boldsymbol{\gamma}_{ik}^{\boldsymbol{\alpha}} = y_{ik} \cdot h_k(\boldsymbol{x}_{ik}^{att})$ is the score of α_{ik}.

This condition states that **convex combinations of tokens with scores lower than α_{ik} cannot achieve a score higher than α_{ik}**. Here, $1 - s_{\alpha_{ik}}$ term denotes the total share of non-optimal tokens. We require this condition to hold over the training dataset rather than the full domain $\mathbb{R}^d$. Crucially, it is a strict generalization of the linearity assumption: Any linear h_k satisfies Assumption B.1 by setting $c_{\boldsymbol{\alpha}} > 0$ to be the difference between the score of α_{ik} and the largest score within $\texttt{low}_{ik}^{\alpha}$ i.e.

$$c_{\boldsymbol{\alpha}} := \min_{i \in [n], k \in [K]} \{\boldsymbol{\gamma}_{ik}^{\boldsymbol{\alpha}} - \max_{t \in \texttt{low}_{ik}^{\alpha}} \gamma_{ikt}\} > 0. \tag{B.124}$$

This can be seen by writing $h_k(\boldsymbol{x}(\boldsymbol{s})) = \sum_{t \in \texttt{low}_{ik}^{\alpha} \cup \{\alpha_{ik}\}} s_t \cdot \gamma_{ikt} = \boldsymbol{\gamma}_{ik}^{\boldsymbol{\alpha}} + \sum_{t \in \texttt{low}_{ik}^{\alpha}} s_t \cdot (\gamma_{ikt} - \boldsymbol{\gamma}_{ik}^{\boldsymbol{\alpha}}) \leq \boldsymbol{\gamma}_{ik}^{\boldsymbol{\alpha}} - c_{\boldsymbol{\alpha}}(1 - s_{\alpha_{ik}})$. To provide a nonlinear example, consider the setting all labels are $y_{ik} = 1$ and h is an arbitrary convex function. Thanks to convexity, we can write $h(\boldsymbol{x}(\boldsymbol{s})) \leq \sum_{t=1}^{T} s_t \cdot h(\boldsymbol{x}_t) = \sum_{t \in \texttt{low}_{ik}^{\alpha} \cup \{\alpha_{ik}\}} s_t \cdot \gamma_{ikt}$. Thus, we can use the same choice of $c_{\boldsymbol{\alpha}}$ in (B.124).

We remark that, in Section 3.6, we derive formulae describing general inductive bias of attention without enforcing any assumption on h. These formulae allow for arbitrary output tokens generated by the transformer model trained by gradient descent. This includes the setting where gradient descent selects and composes multiple tokens from each sequence rather than a single token α_{ik}.

The following result is our main theorem regarding the convergence of regularization path to locally-optimal directions when restricted over the cone $\texttt{cone}_{\epsilon}(\boldsymbol{\alpha})$.

Theorem B.2 (Convergence of Local Regularization Path) *Suppose* (SAtt-SVM) *is feasible and $\boldsymbol{\alpha} = (\alpha_{ik})_{ik=(1,1)}^{(n,k)}$ are locally-optimal token indices. Suppose Assumptions 3.1&B.1 hold. Recall $\texttt{cone}_{\epsilon}(\boldsymbol{\alpha})$ of* (B.123) *and consider the norm-constrained cone*

$$\mathcal{C}_{\epsilon,R_0}^{\diamond} := \texttt{cone}_{\epsilon}(\boldsymbol{\alpha}) \bigcap \{\boldsymbol{W} \mid \|\boldsymbol{W}\|_{\diamond} \geq R_0\}.$$

Define the conic regularization path $\bar{\boldsymbol{W}}(R) = \min_{\mathcal{C}_{\epsilon,R_0}^{\diamond}, \|\boldsymbol{W}\|_{\diamond} \leq R} \mathcal{L}(\boldsymbol{W})$. Let $\mathcal{W}_{\boldsymbol{\alpha}}^{mm}$ be its set of minima and $\Xi_{\diamond} > 0$ be the associated margin i.e. $\Xi_{\diamond} = 1/\|\mathcal{W}_{\boldsymbol{\alpha}}^{mm}\|_{\diamond}$. For any sufficiently small $\epsilon > 0$ and sufficiently large $R_0 = \mathcal{O}(1/\epsilon) > 0$, $\lim_{R \to \infty} \texttt{dist}\left(\frac{\bar{\boldsymbol{W}}(R)}{R\Xi_{\diamond}}, \mathcal{W}_{\boldsymbol{\alpha}}^{mm}\right) = 0$. Additionally,

suppose optimal indices $opt = (opt_{ik})_{ik=(1,1)}^{(n,k)}$ *are unique and set* $\boldsymbol{\alpha} \leftarrow opt$*. Then, the same RP convergence guarantee holds with* $\mathcal{C}^{\diamond}_{\epsilon,R_0} = \mathcal{R}_m$*.*

Proof. We will prove that $\bar{\boldsymbol{W}}(R)$ is the optimal direction and also $\|\bar{\boldsymbol{W}}(R)\|_\diamond \to \infty$. Define the absolute constant

$$c_\diamond = \min_{\|\boldsymbol{W}\|_\diamond = 1} \|\boldsymbol{W}\|_F.$$

This guarantees that for any $\boldsymbol{W}$ we have $\|\boldsymbol{W}\|_F \geq c_\diamond \|\boldsymbol{W}\|_\diamond$. Also denote $\varepsilon_\diamond = c_\diamond \epsilon$. Let us first determine the ϵ parameter: Fix $\boldsymbol{W}^{\mathrm{mm}}_{\boldsymbol{\alpha}} \in \mathcal{W}^{\mathrm{mm}}_{\boldsymbol{\alpha}}$. For general $\boldsymbol{\alpha}$, we can choose any $\epsilon > 0$ that is sufficiently small to guarantee $\boldsymbol{W}^{\mathrm{mm}}_{\boldsymbol{\alpha}} \in \mathtt{cone}_\epsilon(\boldsymbol{\alpha})$ based on Lemma B.10. For $\boldsymbol{\alpha} = \mathrm{opt}$, our analysis will entirely avoid using ϵ, specifically, observe that $\mathtt{cone}_\epsilon(\boldsymbol{\alpha}) = \mathcal{R}_m$ based on Lemma B.10.

Step 1: Let us first prove that $\bar{\boldsymbol{W}}(R)$ achieves the optimal risk as $R \to \infty$ – rather than problem having finite optima. Define norm-normalized $\bar{\boldsymbol{W}}^{\mathrm{mm}} = \Xi_\diamond \boldsymbol{W}^{\mathrm{mm}}_{\boldsymbol{\alpha}}$. Note that $\boldsymbol{W}^{\mathrm{mm}}_{\boldsymbol{\alpha}}$ separates tokens $\boldsymbol{\alpha}$ from rest of the tokens for each $i, k \in [n] \times [K]$. Thus, we have that

$$\lim_{R\to\infty} \mathcal{L}(\bar{\boldsymbol{W}}(R)) \leq \lim_{R\to\infty} \mathcal{L}(R \cdot \bar{\boldsymbol{W}}^{\mathrm{mm}}) := \mathcal{L}_\star = \frac{1}{n}\sum_{i=1}^{n}\sum_{k=1}^{K} \ell(\boldsymbol{\gamma}^{\boldsymbol{\alpha}}_{ik}). \tag{B.125}$$

On the other hand, for any choice of $\boldsymbol{W} \in \mathtt{cone}_\epsilon(\boldsymbol{\alpha})$, set $\boldsymbol{x}^{\boldsymbol{W}}_{ik} = \sum_{t=1}^{T} \mathbb{S}(\boldsymbol{X}_i \boldsymbol{W} \boldsymbol{z}_{ik})_t \boldsymbol{x}_t$. Set softmax probabilities $\boldsymbol{s}^{(ik)} = \mathbb{S}(\boldsymbol{X}_i \boldsymbol{W} \boldsymbol{z}_{ik})$. Recalling $\mathtt{low}^\alpha_{ik}, \mathtt{high}^\alpha_{ik}$ definitions, we can decompose the attention features as

$$\boldsymbol{x}^{\boldsymbol{W}}_{ik} = \boldsymbol{s}^{(ik)}_{\alpha_{ik}} \boldsymbol{x}^{\mathrm{att}}_{ik} + \sum_{t\in\mathtt{low}^\alpha_{ik}} \boldsymbol{s}^{(ik)}_t \boldsymbol{x}_{it} + \sum_{\tau\in\mathtt{high}^\alpha_{ik}} \boldsymbol{s}^{(ik)}_\tau \boldsymbol{x}_{i\tau}. \tag{B.126}$$

When $\boldsymbol{\alpha} = \mathrm{opt}$, note that we simply have $\mathtt{high}^\alpha_{ik} = \emptyset$. This will be important for setting $R_0 = 0$ and $\mathcal{C}^{\diamond}_{\epsilon,R_0} = \mathcal{R}_m$ in the proof for opt indices.

Set $\gamma^{\mathrm{gap}}_{ikt} = \gamma_{ikt} - \boldsymbol{\gamma}^{\boldsymbol{\alpha}}_{ik} = y_{ik} \cdot (h_k(\boldsymbol{x}_{it}) - h_k(\boldsymbol{x}^{\mathrm{att}}_{ik}))$. Building on L_h-Lipschitzness of the prediction head $h_k(\cdot)$, we define

$$B := \max_{i\in[n],k\in[K]} \max_{t,\tau\in[T]} L_h \cdot \|\boldsymbol{x}_{it} - \boldsymbol{x}_{i\tau}\| \geq |\gamma^{\mathrm{gap}}_{ikt}|. \tag{B.127}$$

Define $P^{ik} := \sum_{t\in\mathtt{low}^\alpha_{ik}} \boldsymbol{s}^{(ik)}_t$, $Q^{ik} := \sum_{t\in\mathtt{high}^\alpha_{ik}} \boldsymbol{s}^{(ik)}_t$, and $\gamma^{\boldsymbol{W}}_{ik} = y_{ik} \cdot h_k(\boldsymbol{x}^{\boldsymbol{W}}_{ik})$. Also set temporary variables $\boldsymbol{x}' = (\boldsymbol{s}^{(ik)}_{\alpha_{ik}} + Q^{ik})\boldsymbol{x}^{\mathrm{att}}_{ik} + \sum_{t\in\mathtt{low}^\alpha_{ik}} \boldsymbol{s}^{(ik)}_t \boldsymbol{x}_{it}$ and $\gamma' = y_{ik} \cdot h_k(\boldsymbol{x}^{\boldsymbol{W}}_{ik})$. Using Assumption B.1 on $\boldsymbol{x}'$ and noticing $P^{ik} = 1 - \boldsymbol{s}^{(ik)}_{\alpha_{ik}} - Q^{ik}$, observe that

$$|\gamma^{\boldsymbol{W}}_{ik} - \gamma'| \leq BQ^{ik} \quad \text{and} \quad \gamma' \leq \boldsymbol{\gamma}^{\boldsymbol{\alpha}}_{ik} - c_{\boldsymbol{\alpha}} P^{ik}.$$

Recall from (B.124) that, when h_k are linear functions, $c_{\boldsymbol{\alpha}}$ can be chosen as

$$c_{\boldsymbol{\alpha}} := \min_{i\in[n],k\in[K]} \min_{t\in \mathtt{low}_{ik}^{\alpha}} -\gamma_{ikt}^{\mathrm{gap}} > 0.$$

To summarize, applying Assumption B.1, we obtain the following score inequalities

$$\gamma_{ik}^{\boldsymbol{W}} \le \boldsymbol{\gamma}_{ik}^{\boldsymbol{\alpha}} - c_{\boldsymbol{\alpha}} P^{ik} + BQ^{ik}, \tag{B.128}$$

$$|\gamma_{ik}^{\boldsymbol{W}} - \boldsymbol{\gamma}_{ik}^{\boldsymbol{\alpha}}| \le L_h \|\boldsymbol{x}_{ik}^{\boldsymbol{W}} - \boldsymbol{x}_{ik}^{\mathrm{att}}\| \le L_h \sum_{t\neq\alpha_{ik}} \boldsymbol{s}_t^{(ik)} \|\boldsymbol{x}_{ikt} - \boldsymbol{x}_{ik}^{\mathrm{att}}\| \le B(1 - \boldsymbol{s}_{\alpha_{ik}}^{(ik)}). \tag{B.129}$$

We will use the $\gamma_{ik}^{\boldsymbol{W}} - \boldsymbol{\gamma}_{ik}^{\boldsymbol{\alpha}}$ term in (B.128) to evaluate $\boldsymbol{W}$ against the reference loss (B.125). Let $\boldsymbol{a}^{(ik)} = \boldsymbol{X}_i \boldsymbol{W} \boldsymbol{z}_{ik}$. Now since $\boldsymbol{W} \in \mathtt{cone}_{\epsilon}(\boldsymbol{\alpha})$, there exists $t \in \mathtt{low}_{ik}^{\alpha}$ obeying $\boldsymbol{a}_t^{(ik)} - \max_{\tau\in\mathtt{high}_{ik}^{\alpha}} \boldsymbol{a}_\tau^{(ik)} \ge \epsilon\|\boldsymbol{W}\|_F \ge \varepsilon_\diamond \|\boldsymbol{W}\|_\diamond$. Denote $D^{ik} := (\sum_{t\in[T]} e^{\boldsymbol{a}_t^{(ik)}})^{-1}$ to be the softmax denominator i.e. sum of exponentials. We find that,

$$Q^{ik} = \sum_{\tau\in\mathtt{high}_{ik}^{\alpha}} \boldsymbol{s}_\tau^{(ik)} = D^{ik} \sum_{\tau\in\mathtt{high}_{ik}^{\alpha}} e^{\boldsymbol{a}_\tau^{(ik)}} \le D^{ik} T e^{\boldsymbol{a}_t^{(ik)} - \epsilon\|\boldsymbol{W}\|_F} \le T e^{-\varepsilon_\diamond \|\boldsymbol{W}\|_\diamond} P^{ik}. \tag{B.130}$$

Consequently, the score difference obeys

$$\gamma_{ik}^{\boldsymbol{W}} - \boldsymbol{\gamma}_{ik}^{\boldsymbol{\alpha}} \le BQ^{ik} - c_{\boldsymbol{\alpha}} P^{ik} \le (BTe^{-\varepsilon_\diamond\|\boldsymbol{W}\|_\diamond} - c_{\boldsymbol{\alpha}}) P^{ik}.$$

Above, the right hand side is strictly negative as soon as $\|\boldsymbol{W}\|_\diamond \ge R_0 := \frac{1}{\varepsilon_\diamond} \log \frac{BT}{c_{\boldsymbol{\alpha}}}$. Note that, this condition applies to all $(i,k) \in [n]\times[K]$ pairs uniformly for the same R_0. Consequently, for any $\|\boldsymbol{W}\|_\diamond \ge R_0$, for all i,k and $\boldsymbol{W} \in \mathtt{cone}_{\epsilon}(\boldsymbol{\alpha})$, we have that $\gamma_{ik}^{\boldsymbol{W}} < \boldsymbol{\gamma}_{ik}^{\boldsymbol{\alpha}}$. Additionally, when $\boldsymbol{\alpha} = \mathrm{opt}$, note that $Q^{ik} = 0$ since $\mathtt{high}_{ik}^{\alpha} = \emptyset$. Thus, $R_0 = 0$ suffices to ensure $\gamma_{ik}^{\boldsymbol{W}} < \boldsymbol{\gamma}_{ik}^{\boldsymbol{\alpha}}$. Using the strictly-decreasing nature of ℓ, we conclude with the fact that for all (finite) $\boldsymbol{W} \in \mathtt{cone}_{\epsilon}(\boldsymbol{\alpha})$,

$$\mathcal{L}(\boldsymbol{W}) = \frac{1}{n}\sum_{i=1}^{n}\sum_{k=1}^{K} \ell(\gamma_{ik}^{\boldsymbol{W}}) > \mathcal{L}_\star = \frac{1}{n}\sum_{i=1}^{n}\sum_{k=1}^{K} \ell(\boldsymbol{\gamma}_{ik}^{\boldsymbol{\alpha}}),$$

which implies $\|\bar{\boldsymbol{W}}(R)\|_\diamond \to \infty$.

Step 2: To proceed, we show that $\bar{\boldsymbol{W}}(R)$ converges in direction to $\mathcal{W}_{\boldsymbol{\alpha}}^{\mathrm{mm}}$. Suppose this is not the case i.e. convergence fails. We will obtain a contradiction by showing that $\bar{\boldsymbol{W}}^{\mathrm{mm}}(R) = R \cdot \bar{\boldsymbol{W}}^{\mathrm{mm}}$ achieves a strictly superior loss compared to $\bar{\boldsymbol{W}}(R)$. Also define the normalized parameter $\bar{\boldsymbol{W}}_0(R) = \frac{\bar{\boldsymbol{W}}(R)}{R\Xi_\diamond}$ and $\boldsymbol{W}' = \frac{\bar{\boldsymbol{W}}(R)}{\|\bar{\boldsymbol{W}}(R)\|_\diamond \Xi_\diamond}$. Note that $\bar{\boldsymbol{W}}_0(R)$ is obtained by scaling down $\boldsymbol{W}'$ since $\|\bar{\boldsymbol{W}}(R)\|_\diamond \le R$ and $\boldsymbol{W}'$ obeys $\|\boldsymbol{W}'\|_\diamond = \|\boldsymbol{W}_{\boldsymbol{\alpha}}^{\mathrm{mm}}\|_\diamond$.

Since $\bar{\boldsymbol{W}}_0(R)$ fails to converge to $\mathcal{W}_{\boldsymbol{\alpha}}^{\mathrm{mm}}$, for some $\delta > 0$, there exists arbitrarily large $R > 0$ such that $\mathtt{dist}\left(\bar{\boldsymbol{W}}_0(R), \mathcal{W}_{\boldsymbol{\alpha}}^{\mathrm{mm}}\right) \geq \delta$. This translates to the suboptimality in terms of margin constraints as follows: First, distance with respect to the $\diamond$-norm obeys $\mathtt{dist}_\diamond\left(\bar{\boldsymbol{W}}_0(R), \mathcal{W}_{\boldsymbol{\alpha}}^{\mathrm{mm}}\right) \geq \delta$ for some updated $\delta \leftarrow c_\diamond \delta$. Secondly, using triangle inequality,

$$\text{This implies that either} \quad \|\bar{\boldsymbol{W}}_0(R)\|_\diamond \leq \|\boldsymbol{W}_{\boldsymbol{\alpha}}^{\mathrm{mm}}\|_\diamond - \delta/2 \quad \text{or} \quad \mathtt{dist}_\diamond\left(\boldsymbol{W}', \mathcal{W}_{\boldsymbol{\alpha}}^{\mathrm{mm}}\right) \geq \delta/2.$$

In either scenario, $\bar{\boldsymbol{W}}_0(R)$ strictly violates one of the margin constraints of (SAtt-SVM): If $\|\bar{\boldsymbol{W}}_0(R)\|_\diamond \leq \|\boldsymbol{W}_{\boldsymbol{\alpha}}^{\mathrm{mm}}\|_\diamond - \delta/2$, then, since the optimal SVM objective is $\|\boldsymbol{W}_{\boldsymbol{\alpha}}^{\mathrm{mm}}\|_\diamond$, there exists a constraint (i,k) for which $\left\langle \boldsymbol{F}_{ik}^{\boldsymbol{\alpha}} - \boldsymbol{F}_{ikt}, \bar{\boldsymbol{W}}_0(R)\right\rangle \leq 1 - \frac{\delta}{2\|\boldsymbol{W}_{\boldsymbol{\alpha}}^{\mathrm{mm}}\|_\diamond}$. If $\mathtt{dist}_\diamond\left(\boldsymbol{W}', \mathcal{W}_{\boldsymbol{\alpha}}^{\mathrm{mm}}\right) \geq \delta/2$, then, $\boldsymbol{W}'$ has same SVM objective but it is strictly bounded away from the solution set. Thus, for some $\epsilon := \epsilon(\delta) > 0$, $\boldsymbol{W}'$ and its scaled down version $\bar{\boldsymbol{W}}_0(R)$ strictly violate an SVM constraint achieving margin $\leq 1 - \epsilon$. Without losing generality, suppose $\bar{\boldsymbol{W}}_0(R)$ violates the first constraint. Thus, for a properly updated $\delta > 0$ (that is function of the initial $\delta > 0$) and for $(i,k) = (1,1)$ and some support index $\tau \in \mathcal{T}_{11}$,

$$\left\langle \boldsymbol{F}_{11}^{\boldsymbol{\alpha}} - \boldsymbol{F}_{11t}, \bar{\boldsymbol{W}}_0(R)\right\rangle \leq 1 - \delta. \tag{B.131}$$

Now, we will argue that this will lead to a contradiction by proving $\mathcal{L}(\bar{\boldsymbol{W}}^{\mathrm{mm}}(R)) < \mathcal{L}(\bar{\boldsymbol{W}}(R))$ for sufficiently large R.

To obtain the result, we establish a refined softmax probability control as in Step 1 by studying distance to $\mathcal{L}_\star$. Following (B.128), denote the score function at $\bar{\boldsymbol{W}}(R)$ via $\gamma_{ik}^R = \gamma_{ik}^{\bar{\boldsymbol{W}}(R)}$ as shorthand notation. Similarly, let $\boldsymbol{s}_{ik}^R = \mathbb{S}(\boldsymbol{a}_{ik}^R)$ with $\boldsymbol{a}_{ik}^R = \boldsymbol{X}_i \bar{\boldsymbol{W}}(R) \boldsymbol{z}_{ik}$. Set the corresponding notation for the reference parameter $\bar{\boldsymbol{W}}^{\mathrm{mm}}(R)$ as $\gamma_{ik}^\star, \boldsymbol{s}_{ik}^\star, \boldsymbol{a}_{ik}^\star$.

Critically, recall the above inequalities (B.130) that applies to both $\boldsymbol{W} \in \{\bar{\boldsymbol{W}}(R), \bar{\boldsymbol{W}}^{\mathrm{mm}}(R)\} \subset \mathtt{cone}_\epsilon(\boldsymbol{\alpha})$ for an index (i,k) and support index $t \in \mathcal{T}_{ik}$

$$\begin{aligned} Q^{ik} &= \sum_{\tau \in \mathtt{high}_{ik}^\alpha} \boldsymbol{s}_{ik\tau} = D^{ik} \sum_{\tau \in \mathtt{high}_{ik}^\alpha} e^{\boldsymbol{a}_{ik\tau}} \\ &\leq D^{ik} T e^{\boldsymbol{a}_{ikt} - \varepsilon_\diamond \|\boldsymbol{W}\|_\diamond} \leq T e^{-\varepsilon_\diamond \|\boldsymbol{W}\|_\diamond} P^{ik} \leq T e^{-\varepsilon_\diamond \|\boldsymbol{W}\|_\diamond} (1 - \boldsymbol{s}_{ik\alpha_{ik}}), \end{aligned} \tag{B.132}$$

where $P^{ik} = \sum_{\tau \in \mathtt{low}_{ik}^\alpha} \boldsymbol{s}_{ik\tau}$ and $P^{ik} + Q^{ik} = 1 - \boldsymbol{s}_{ik\alpha_{ik}}$.

Note that, setting $R_0 \geq \mathcal{O}(1/\varepsilon_\diamond) = \mathcal{O}(1/\epsilon)$, we guarantee that, for any $(i,k) \in [n] \times [K]$

$$P^{ik} \geq Q^{ik} \implies P^{ik} \geq 0.5(1 - \boldsymbol{s}_{ik\alpha_{ik}}). \tag{B.133}$$

Additionally, when $\boldsymbol{\alpha} = \mathrm{opt}$, note that $Q^{ik} = 0$ since $\mathtt{high}_{ik}^\alpha = \emptyset$. Thus, $R_0 = 0$ suffices to

ensure (B.133).

To proceed, recall that $R \geq \|\bar{\boldsymbol{W}}(R)\|_\diamond \geq R_0$ by definition since $\bar{\boldsymbol{W}}(R) \in \mathcal{C}^\diamond_{\epsilon,R_0}$ and recall $\Xi_\diamond := 1/\|\boldsymbol{W}^{\mathrm{mm}}_{\boldsymbol{\alpha}}\|_\diamond$. Equipped with these, we note the following softmax inequalities on the selected tokens α_{ik}

$$\begin{aligned} \boldsymbol{s}^\star_{ik\alpha_{ik}} &\geq \frac{1}{1+Te^{-R\Xi_\diamond}} \geq 1 - Te^{-R\Xi_\diamond} \quad \text{for all} \quad (i,k) \in [n]\times[K], \\ s^R_{ik\alpha_{ik}} &\leq \frac{1}{1+e^{-(1-\delta)\|\bar{\boldsymbol{W}}(R)\|_\diamond \Xi_\diamond}} \leq \frac{1}{1+e^{-(1-\delta)R\Xi_\diamond}} \quad \text{for} \quad (i,k)=(1,1). \end{aligned} \tag{B.134}$$

The former inequality is thanks to $\boldsymbol{W}^{\mathrm{mm}}_{\boldsymbol{\alpha}}$ achieving ≥ 1 margins on all tokens $[T]-\alpha_{ik}$ and the latter arises from the δ-margin violation of $\bar{\boldsymbol{W}}(R)$ at $(i,k)=(1,1)$ i.e. Eq. (B.131). Since ℓ is strictly decreasing with Lipschitz gradient and the scores are upper/lower bounded by an absolute constant (as tokens are bounded, $(h_k)_{k=1}^K$ are Lipschitz, and both are fixed), we know that $c_{\mathtt{up}} \geq -\ell'(\gamma^{\boldsymbol{W}}_{ik}) \geq c_{\mathtt{dn}}$ for some constants $c_{\mathtt{up}} > c_{\mathtt{dn}} > 0$. Thus, following Eq. (B.127) and the score decomposition (B.128), and using (B.132),(B.133),(B.134) we can write

$$\begin{aligned} \mathcal{L}(\bar{\boldsymbol{W}}(R)) - \mathcal{L}_\star &\geq \frac{1}{n}[\ell(\gamma^{\bar{\boldsymbol{W}}(R)}_{11}) - \ell(\boldsymbol{\gamma}^{\boldsymbol{\alpha}}_{11})] \geq \frac{c_{\mathtt{dn}}}{n}(\boldsymbol{\gamma}^{\boldsymbol{\alpha}}_{11} - \gamma^{\bar{\boldsymbol{W}}(R)}_{11}) \\ &\geq \frac{c_{\mathtt{dn}}}{n}(c_{\boldsymbol{\alpha}} P^{11}_{\bar{\boldsymbol{W}}(R)} - BQ^{11}_{\bar{\boldsymbol{W}}(R)}) \\ &\geq \frac{c_{\mathtt{dn}}}{n}(1 - \boldsymbol{s}^R_{11\alpha_{11}})(0.5c_{\boldsymbol{\alpha}} - BTe^{-\varepsilon_\diamond \|\bar{\boldsymbol{W}}(R)\|_\diamond}) \\ &\geq \frac{c_{\mathtt{dn}}}{n}\frac{1}{1+e^{(1-\delta)R\Xi_\diamond}}(0.5c_{\boldsymbol{\alpha}} - BTe^{-\varepsilon_\diamond R_0}). \end{aligned} \tag{B.135}$$

Above, recalling the choice $R_0 \geq \mathcal{O}(1/\varepsilon_\diamond) = \mathcal{O}(1/\epsilon)$, $R \geq R_0$ implies $BTe^{-\varepsilon_\diamond R_0} \leq c_{\boldsymbol{\alpha}}/4$ to obtain

$$\mathcal{L}(\bar{\boldsymbol{W}}(R)) - \mathcal{L}_\star \geq \frac{c_{\mathtt{dn}} \cdot c_{\boldsymbol{\alpha}}}{4n}\frac{1}{1+e^{(1-\delta)R\Xi_\diamond}}. \tag{B.136}$$

Additionally when $\boldsymbol{\alpha} = \mathrm{opt}$, since $Q^{11}_{\bar{\boldsymbol{W}}(R)} = 0$ in (B.135), the bound above holds with $R_0 = 0$ by directly using (B.135).

Conversely, we upper bound the difference between $\mathcal{L}(\bar{\boldsymbol{W}}^{\mathrm{mm}}(R))$ and $\mathcal{L}_\star$ as follows. Define the worst-case loss difference for $\bar{\boldsymbol{W}}(R)$ as $(i',k') = \arg\max_{i\in[n],k\in[K]}[\ell(\gamma^\star_{ik}) - \ell(\boldsymbol{\gamma}^{\boldsymbol{\alpha}}_{ik})]$. Using

(B.129)&(B.134), we write

$$\begin{aligned}\mathcal{L}(\bar{\boldsymbol{W}}^{\mathrm{mm}}(R)) - \mathcal{L}_\star &\leq \max_{i\in[n],k\in[K]} [\ell(\gamma_{ik}^\star) - \ell(\boldsymbol{\gamma}_{ik}^{\boldsymbol{\alpha}})] \leq c_{\mathtt{up}} \cdot (\boldsymbol{\gamma}_{i'k'}^{\boldsymbol{\alpha}} - \gamma_{i'k'}^\star) \\ &\leq c_{\mathtt{up}} \cdot (1 - \boldsymbol{s}_{i'k'\alpha_{i'k'}}^\star) B \\ &\leq c_{\mathtt{up}} \cdot T e^{-R\Xi_\diamond} B.\end{aligned} \tag{B.137}$$

Combining the last inequality and (B.136), we conclude that $\mathcal{L}(\bar{\boldsymbol{W}}^{\mathrm{mm}}(R)) < \mathcal{L}(\bar{\boldsymbol{W}}(R))$ whenever

$$c_{\mathtt{up}} T \cdot e^{-R\Xi_\diamond} B < \frac{c_{\mathtt{dn}} \cdot c_{\boldsymbol{\alpha}}}{4n} \frac{1}{1 + e^{(1-\delta)R\Xi_\diamond}} \iff \frac{e^{R\Xi_\diamond}}{1 + e^{(1-\delta)R\Xi_\diamond}} > \frac{4 c_{\mathtt{up}} T n B}{c_{\mathtt{dn}} c_{\boldsymbol{\alpha}}}.$$

The left hand-side inequality holds for all sufficiently large R: Specifically, as soon as R obeys $R > \frac{1}{\delta\Xi_\diamond} \log(\frac{8 c_{\mathtt{up}} T n B}{c_{\mathtt{dn}} c_{\boldsymbol{\alpha}}})$. This completes the proof of the theorem via contradiction as we obtained $\mathcal{L}(\bar{\boldsymbol{W}}(R)) > \mathcal{L}(\bar{\boldsymbol{W}}^{\mathrm{mm}}(R))$. ∎

B.5.2 Global regularization path

The following result is a direct corollary of Theorem B.2. Namely, we simply restate the final line of this theorem that applies to optimal tokens.

Corollary B.1 (Global Convergence of Regularization Path) *Suppose Assumptions 3.1&B.1 hold and the optimal indices* $opt_{ik} = \arg\max_{t\in[T]} \gamma_{ikt}$ *are unique. Consider the global regularization path* $\bar{\boldsymbol{W}}_{\diamond,R} = \min_{\boldsymbol{W}\in\mathcal{R}_m, \|\boldsymbol{W}\|_\diamond \leq R} \mathcal{L}(\boldsymbol{W})$. *Let* $\mathcal{W}_\diamond^{mm}$ *be the non-empty solution set of* (SAtt-SVM) *with* $\boldsymbol{\alpha} \leftarrow$ *opt normalized to have unit* $\diamond$*-norm. Then*

$$\lim_{R\to\infty} \mathtt{dist}\left(\frac{\bar{\boldsymbol{W}}_{\diamond,R}}{R}, \mathcal{W}_\diamond^{mm}\right)$$

The next corollary directly targets application to (SERM-W) and (SERM-KQ). This corollary is also a strict generalization of Theorem 3.2. Specifically, we immediately recover Theorem 3.2 by specializing this to the single-output setting $K \leftarrow 1$ and full-dimensional parameterization $m \leftarrow d$.

Corollary B.2 *Suppose Assumptions 3.1&B.1 hold and the optimal indices* $opt_{ik} = \arg\max_{t\in[T]} \gamma_{ikt}$ *are unique. Consider the regularization paths associated to* (SERM-W) *and*

(SERM-KQ)*:*

$$\bar{\boldsymbol{W}}(R) = \underset{\boldsymbol{W}\in\mathcal{R}_m, \|\boldsymbol{W}\|_F\leq R}{\arg\min} \mathcal{L}(\boldsymbol{W}) \quad \textit{and} \quad \bar{\boldsymbol{W}}_k(R), \bar{\boldsymbol{W}}_q(R) = \underset{\|\boldsymbol{W}_k\|_F^2+\|\boldsymbol{W}_q\|_F^2\leq 2R}{\arg\min} \mathcal{L}(\boldsymbol{W}_k, \boldsymbol{W}_q) \tag{B.138}$$

Suppose (SAtt-SVM) *is feasible for* $\boldsymbol{\alpha} \leftarrow$ *opt. Let* $\boldsymbol{W}^{mm}$ *be the unique solution of* (SAtt-SVM) *with Frobenius norm and* $\mathcal{W}_\star^{mm}$ *be the solution set of* (SAtt-SVM) *with nuclear norm and cost function* $\|\mathcal{W}_\star^{mm}\|_\star$*. We have that*

$$\lim_{R\to\infty} \frac{\bar{\boldsymbol{W}}(R)}{R} = \frac{\boldsymbol{W}^{mm}}{\|\boldsymbol{W}^{mm}\|_F}, \quad \lim_{R\to\infty} \texttt{dist}\left(\frac{\bar{\boldsymbol{W}}_q(R)\bar{\boldsymbol{W}}_k(R)^\top}{R}, \frac{\mathcal{W}_\star^{mm}}{\|\mathcal{W}_\star^{mm}\|_\star}\right) = 0.$$

Proof. We directly apply Corollary B.1 with $\diamond = F$ and $\diamond = \star$ respectively. To obtain the result on $\bar{\boldsymbol{W}}(R)$, we note that $\boldsymbol{W}^{\rm mm}$ is unique because Frobenius norm-squared is strongly convex. To obtain the result on $(\bar{\boldsymbol{W}}_q(R), \bar{\boldsymbol{W}}_k(R))$, we use Lemma B.9 and observe that

$$\bar{\boldsymbol{W}}_{\star,R} := \bar{\boldsymbol{W}}_q(R)\bar{\boldsymbol{W}}_k(R)^\top \in \underset{\boldsymbol{W}\in\mathcal{R}_m, \|\boldsymbol{W}\|_\star\leq R}{\arg\min} \mathcal{L}(\boldsymbol{W}).$$

We then apply Corollary B.1 with $\diamond = \star$ to conclude with the convergence of the path $\bar{\boldsymbol{W}}_{\star,R}$. ■

B.5.2.1 Proof of Theorem 3.2

Corollary B.2 already proves Theorem 3.2 through the more general Theorem B.2. Below, we provide a self contained proof of Theorem 3.2 for clarity.

Proof. Throughout $\diamond$ denotes either Frobenius norm or nuclear norm. We will prove that $\bar{\boldsymbol{W}}(R)$ asymptotically aligns with the set of globally-optimal directions and also $\|\bar{\boldsymbol{W}}(R)\|_\diamond \to \infty$. $\mathcal{R}_m \subseteq \mathbb{R}^{d\times d}$ denote the manifold of rank $\leq m$ matrices.

Step 1: Let us first prove that $\bar{\boldsymbol{W}}(R)$ achieves the optimal risk as $R \to \infty$ – rather than problem having finite optima. Define $\Xi_\diamond = 1/\|\boldsymbol{W}^{\rm mm}\|_\diamond$ and norm-normalized $\bar{\boldsymbol{W}}^{\rm mm} = \Xi_\diamond \boldsymbol{W}^{\rm mm}$. Note that $\boldsymbol{W}^{\rm mm}$ separates tokens opt from rest of the tokens for each $i \in [n]$. Thus, we have that

$$\lim_{R\to\infty} \mathcal{L}(\bar{\boldsymbol{W}}(R)) \leq \lim_{R\to\infty} \mathcal{L}(R\cdot\bar{\boldsymbol{W}}^{\rm mm}) := \mathcal{L}_\star = \frac{1}{n}\sum_{i=1}^n \ell(\gamma_i^{\rm opt}). \tag{B.139}$$

On the other hand, for any $\boldsymbol{W} \in \mathcal{R}_m$, define the softmax probabilities $\boldsymbol{s}^{(i)} = \mathbb{S}(\boldsymbol{X}_i\boldsymbol{W}\boldsymbol{z}_i)$ and attention features $\boldsymbol{x}_i^{\boldsymbol{W}} = \sum_{t=1}^T \boldsymbol{s}_t^{(i)}\boldsymbol{x}_t$. Decompose $\boldsymbol{x}_i^{\boldsymbol{W}}$ as $\boldsymbol{x}_i^{\boldsymbol{W}} = \boldsymbol{s}_{{\rm opt}_i}^{(i)}\boldsymbol{x}_{i{\rm opt}_i} + \sum_{t\neq{\rm opt}_i} \boldsymbol{s}_t^{(i)}\boldsymbol{x}_{it}$.

Set $\gamma_{it}^{\rm gap} = \gamma_i^{\rm opt} - \gamma_{it} = y_i \cdot \boldsymbol{v}^\top(\boldsymbol{x}_{i{\rm opt}_i} - \boldsymbol{x}_{it}) > 0$, and define

$$B := \max_{i\in[n]} \max_{t,\tau\in[T]} \|\boldsymbol{v}\| \cdot \|\boldsymbol{x}_{it} - \boldsymbol{x}_{i\tau}\| \geq \gamma_{it}^{\rm gap}. \tag{B.140}$$

Define $c_{\rm opt} = \min_{i\in[n],t\neq{\rm opt}_i} \gamma_{it}^{\rm gap} > 0$ and $\gamma_i^{\boldsymbol{W}} = y_i \cdot \boldsymbol{v}^\top \boldsymbol{x}_i^{\boldsymbol{W}}$. We obtain the following score inequalities

$$\begin{aligned} &\gamma_i^{\boldsymbol{W}} \leq \gamma_i^{\rm opt} - c_{\rm opt}(1 - \boldsymbol{s}_{{\rm opt}_i}^{(i)}) < \gamma_i^{\rm opt}, \\ &|\gamma_i^{\boldsymbol{W}} - \gamma_i^{\rm opt}| \leq \|\boldsymbol{v}\| \cdot \|\boldsymbol{x}_i^{\boldsymbol{W}} - \boldsymbol{x}_i^{\rm att}\| \leq \|\boldsymbol{v}\| \sum_{t\neq{\rm opt}_i} \boldsymbol{s}_t^{(i)} \|\boldsymbol{x}_{it} - \boldsymbol{x}_i^{\rm att}\| \leq B(1 - \boldsymbol{s}_{{\rm opt}_i}^{(i)}). \end{aligned} \tag{B.141}$$

We will use the $\gamma_i^{\boldsymbol{W}} - \gamma_i^{\rm opt}$ term in (B.141) to evaluate $\boldsymbol{W}$ against the reference loss $\mathcal{L}_\star$ of (B.139). Using the strictly-decreasing nature of ℓ, we conclude with the fact that for all (finite) $\boldsymbol{W} \in \mathcal{R}_m$,

$$\mathcal{L}(\boldsymbol{W}) = \frac{1}{n}\sum_{i=1}^n \ell(\gamma_i^{\boldsymbol{W}}) > \mathcal{L}_\star = \frac{1}{n}\sum_{i=1}^n \ell(\gamma_i^{\rm opt}),$$

which implies $\|\bar{\boldsymbol{W}}(R)\|_\diamond \to \infty$ together with (B.139).

Step 2: To proceed, we show that $\bar{\boldsymbol{W}}(R)$ converges in direction to $\mathcal{W}^{\rm mm}$, which denotes the set of SVM minima. Suppose this is not the case and convergence fails. We will obtain a contradiction by showing that $\bar{\boldsymbol{W}}_R^{\rm mm} = R \cdot \bar{\boldsymbol{W}}^{\rm mm}$ achieves a strictly superior loss compared to $\bar{\boldsymbol{W}}(R)$. Let us introduce the normalized parameters $\bar{\boldsymbol{W}}_0(R) = \frac{\bar{\boldsymbol{W}}(R)}{R\Xi_\diamond}$ and $\boldsymbol{W}' = \frac{\bar{\boldsymbol{W}}(R)}{\|\bar{\boldsymbol{W}}(R)\|_\diamond \Xi_\diamond}$. Note that $\bar{\boldsymbol{W}}_0(R)$ is obtained by scaling down $\boldsymbol{W}'$ since $\|\bar{\boldsymbol{W}}(R)\|_\diamond \leq R$ and $\boldsymbol{W}'$ obeys $\|\boldsymbol{W}'\|_\diamond = \|\boldsymbol{W}^{\rm mm}\|_\diamond$. Since $\bar{\boldsymbol{W}}_0(R)$ fails to converge to $\mathcal{W}^{\rm mm}$, for some $\delta > 0$, there exists arbitrarily large $R > 0$ such that $\texttt{dist}\left(\bar{\boldsymbol{W}}_0(R), \mathcal{W}^{\rm mm}\right) \geq \delta$. This translates to the suboptimality in terms of the margin constraints as follows: First, since nuclear norm dominates Frobenius, distance with respect to the $\diamond$-norm obeys $\texttt{dist}_\diamond\left(\bar{\boldsymbol{W}}_0(R), \mathcal{W}^{\rm mm}\right) \geq \delta$. Secondly, using triangle inequality,

$$\text{this implies that either} \quad \|\bar{\boldsymbol{W}}_0(R)\|_\diamond \leq \|\boldsymbol{W}^{\rm mm}\|_\diamond - \delta/2 \quad \text{or} \quad \texttt{dist}_\diamond\left(\boldsymbol{W}', \mathcal{W}^{\rm mm}\right) \geq \delta/2.$$

In either scenario, $\bar{\boldsymbol{W}}_0(R)$ strictly violates one of the margin constraints of (SAtt-SVM): If $\|\bar{\boldsymbol{W}}_0(R)\|_\diamond \leq \|\boldsymbol{W}^{\rm mm}\|_\diamond - \delta/2$, then, since the optimal SVM objective is $\|\boldsymbol{W}^{\rm mm}\|_\diamond$, there exists a constraint $i, t \neq {\rm opt}_i$ for which $\left\langle (\boldsymbol{x}_i^{\rm opt} - \boldsymbol{x}_{it})\boldsymbol{z}_i^\top, \bar{\boldsymbol{W}}_0(R)\right\rangle \leq 1 - \frac{\delta}{2\|\boldsymbol{W}^{\rm mm}\|_\diamond}$. If $\texttt{dist}_\diamond\left(\boldsymbol{W}', \mathcal{W}^{\rm mm}\right) \geq \delta/2$, then, $\boldsymbol{W}'$ has the same SVM objective but it is strictly bounded away from the solution set. Thus, for some $\epsilon := \epsilon(\delta) > 0$, $\boldsymbol{W}'$ and its scaled down version $\bar{\boldsymbol{W}}_0(R)$ strictly violate an SVM constraint achieving margin $\leq 1 - \epsilon$. Without losing generality, suppose $\bar{\boldsymbol{W}}_0(R)$ violates the first constraint $i = 1$. Thus, for a properly updated $\delta > 0$

(that is function of the initial $\delta > 0$) and for $i = 1$ and some support index $\tau \in \mathcal{T}_1$,

$$\left\langle (\boldsymbol{x}_1^{\rm opt} - \boldsymbol{x}_{1t})\boldsymbol{z}_1^\top, \bar{\boldsymbol{W}}_0(R) \right\rangle \le 1 - \delta. \tag{B.142}$$

Now, we will argue that this leads to a contradiction by proving $\mathcal{L}(\bar{\boldsymbol{W}}_R^{\rm mm}) < \mathcal{L}(\bar{\boldsymbol{W}}(R))$ for sufficiently large R.

To obtain the result, we establish a refined softmax probability control as in Step 1 by studying distance to $\mathcal{L}_\star$. Following (B.141), denote the score function at $\bar{\boldsymbol{W}}(R)$ via $\gamma_i^R := \gamma_i^{\bar{\boldsymbol{W}}(R)}$. Similarly, let $\boldsymbol{s}_i^R = \mathbb{S}(\boldsymbol{a}_i^R)$ with $\boldsymbol{a}_i^R = \boldsymbol{X}_i \bar{\boldsymbol{W}}(R)\boldsymbol{z}_i$. Set the corresponding notation for the reference parameter $\bar{\boldsymbol{W}}_R^{\rm mm}$ as $\gamma_i^\star, \boldsymbol{s}_i^\star, \boldsymbol{a}_i^\star$. Recall that $R \ge \|\bar{\boldsymbol{W}}(R)\|_\diamond$ and $\Xi_\diamond := 1/\|\boldsymbol{W}^{\rm mm}\|_\diamond$. We note the following softmax inequalities

$$\begin{aligned} \boldsymbol{s}^\star_{i{\rm opt}_i} &\ge \frac{1}{1 + Te^{-R\Xi_\diamond}} \ge 1 - Te^{-R\Xi_\diamond} \quad \text{for all} \quad i \in [n], \\ s^R_{i{\rm opt}_i} &\le \frac{1}{1 + e^{-(1-\delta)\|\bar{\boldsymbol{W}}(R)\|_\diamond \Xi_\diamond}} \le \frac{1}{1 + e^{-(1-\delta)R\Xi_\diamond}} \quad \text{for} \quad i = 1. \end{aligned} \tag{B.143}$$

The former inequality is thanks to $\boldsymbol{W}^{\rm mm}$ achieving ≥ 1 margins on all tokens $[T] - {\rm opt}_i$ and the latter arises from the δ-margin violation of $\bar{\boldsymbol{W}}(R)$ at $i = 1$ i.e. Eq. (B.142). Since ℓ is strictly decreasing with Lipschitz derivative and the scores are upper/lower bounded by an absolute constant (as tokens are bounded and fixed), we have that $c_{\tt up} \ge -\ell'(\gamma_i^{\boldsymbol{W}}) \ge c_{\tt dn}$ for some constants $c_{\tt up} > c_{\tt dn} > 0$. Thus, following Eq. (B.140), the score decomposition (B.141), and (B.143) we can write

$$\begin{aligned} \mathcal{L}(\bar{\boldsymbol{W}}(R)) - \mathcal{L}_\star &\ge \frac{1}{n}[\ell(\gamma_1^{\bar{\boldsymbol{W}}(R)}) - \ell(\gamma_1^{\rm opt})] \ge \frac{c_{\tt dn}}{n}(\gamma_1^{\rm opt} - \gamma_1^{\bar{\boldsymbol{W}}(R)}) \\ &\ge \frac{c_{\tt dn}}{n} c_{\rm opt}(1 - \boldsymbol{s}^R_{1{\rm opt}_1}). \\ &\ge \frac{c_{\tt dn} c_{\rm opt}}{n} \frac{1}{1 + e^{(1-\delta)R\Xi_\diamond}}. \end{aligned} \tag{B.144}$$

Conversely, we upper bound the difference between $\mathcal{L}(\bar{\boldsymbol{W}}_R^{\rm mm})$ and $\mathcal{L}_\star$ as follows. Define the worst-case loss difference for $\bar{\boldsymbol{W}}(R)$ as $j = \arg\max_{i\in[n]}[\ell(\gamma_i^\star) - \ell(\gamma_i^{\rm opt})]$. Using (B.141)&(B.143), we write

$$\begin{aligned} \mathcal{L}(\bar{\boldsymbol{W}}_R^{\rm mm}) - \mathcal{L}_\star &\le \max_{i\in[n]}[\ell(\gamma_i^\star) - \ell(\gamma_i^{\rm opt})] \le c_{\tt up} \cdot (\gamma_j^{\rm opt} - \gamma_j^\star) \\ &\le c_{\tt up} \cdot (1 - \boldsymbol{s}^\star_{j{\rm opt}_j})B \\ &\le c_{\tt up} \cdot Te^{-R\Xi_\diamond} B. \end{aligned}$$

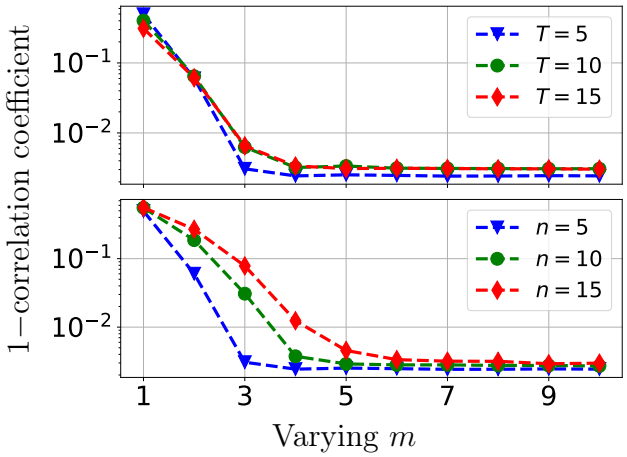


Figure B.1: Convergence behavior of GD when training attention weights $(\boldsymbol{W}_k, \boldsymbol{W}_q) \in \mathbb{R}^{d\times m}$ with random data and varying m. The misalignment between attention SVM and GD, $1-\texttt{corr\_coef}(\boldsymbol{W}_{\star,\boldsymbol{\alpha}}^{\mathrm{mm}}, \boldsymbol{W}_k\boldsymbol{W}_q^\top)$, is studied. $\boldsymbol{W}_{\star,\boldsymbol{\alpha}}^{\mathrm{mm}}$ is from (W-SVM$_\star$) with GD tokens $\boldsymbol{\alpha}$ and $m = d$. Subfigures with fixed $n = 5$ and $T = 5$ show that as m approaches or exceeds n, $\boldsymbol{W}_k\boldsymbol{W}_q^\top$ aligns more with $\boldsymbol{W}_{\star,\boldsymbol{\alpha}}^{\mathrm{mm}}$.

Combining the last inequality and (B.144), we conclude that $\mathcal{L}(\bar{\boldsymbol{W}}_R^{\mathrm{mm}}) < \mathcal{L}(\bar{\boldsymbol{W}}(R))$ whenever

$$c_{\mathtt{up}} T \cdot e^{-R\Xi_\diamond} B < \frac{c_{\mathtt{dn}} \cdot c_{\mathrm{opt}}}{n} \frac{1}{1+e^{(1-\delta)R\Xi_\diamond}} \iff \frac{e^{R\Xi_\diamond}}{1+e^{(1-\delta)R\Xi_\diamond}} > \frac{c_{\mathtt{up}} TnB}{c_{\mathtt{dn}} c_{\mathrm{opt}}}.$$

The left hand-side inequality holds for all sufficiently large R: Specifically, as soon as R obeys $R > \frac{1}{\delta\Xi_\diamond} \log(\frac{2c_{\mathtt{up}} TnB}{c_{\mathtt{dn}} c_{\mathrm{opt}}})$. This completes the proof of the theorem by contradiction since we obtained $\mathcal{L}(\bar{\boldsymbol{W}}(R)) > \mathcal{L}(\bar{\boldsymbol{W}}_R^{\mathrm{mm}})$. ∎

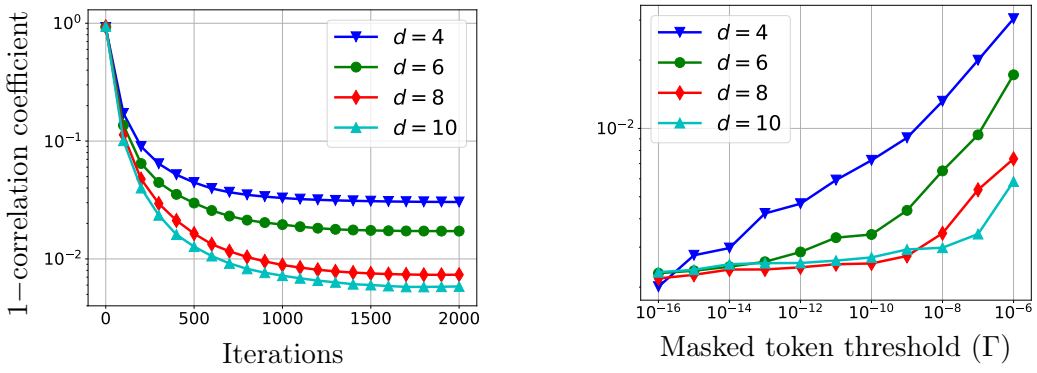
B.6 Supporting Experiments

In this section, we introduce implementation details and additional experiments. Code is available at

https://github.com/umich-sota/TF-as-SVM

We create a 1-layer self-attention using `PyTorch`, training it with the SGD optimizer and a learning rate of $\eta = 0.1$. We apply normalized gradient descent to ensure divergence of attention weights. The attention weight $\boldsymbol{W}$ is then updated through

$$\boldsymbol{W}(k+1) = \boldsymbol{W}(k) - \eta \frac{\nabla\mathcal{L}(\boldsymbol{W}(k))}{\|\nabla\mathcal{L}(\boldsymbol{W}(k))\|_F}.$$



(a) Evolution of correlation under varying d (b) Γ vs correlation coefficient

Figure B.2: Behavior of GD with nonlinear nonconvex prediction head and multi-token compositions. **(a)**: Blue, green, red and teal curves represent the evolution of $1 - \texttt{corr\_coef}(\boldsymbol{W}, \boldsymbol{W}^{\texttt{SVMeq}})$ for $d = 4, 6, 8$ and 10 respectively, which have been displayed in Figure 3.9(upper). **(b)**: Over the 500 random instances as discussed in Figure 3.9, we filter different instances by constructing masked set with tokens whose softmax output $< \Gamma$ and vary Γ from 10^{-16} to 10^{-6}. The corresponding results of $1 - \texttt{corr\_coef}(\boldsymbol{W}, \boldsymbol{W}^{\texttt{SVMeq}})$ are displayed in blue, green, red and teal curves.

In the setting of $(\boldsymbol{W}_k, \boldsymbol{W}_q)$-parameterization, we noted that with extended training iterations, the norm of the combined parameter $\boldsymbol{W}_k \boldsymbol{W}_q^\top$ consistently rises, despite the gradient being treated as zero due to computational limitations. To tackle this issue, we introduce a minor regularization penalty to the loss function, ensuring that the norms of $\boldsymbol{W}_k$ and $\boldsymbol{W}_q$ remain within reasonable bounds. This adjustment involves

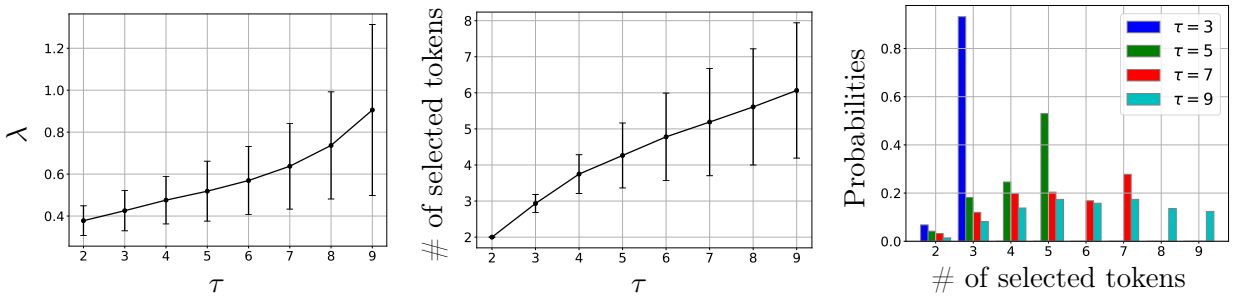
$$\widetilde{\mathcal{L}}(\boldsymbol{W}_k, \boldsymbol{W}_q) = \mathcal{L}(\boldsymbol{W}_k, \boldsymbol{W}_q) + \lambda(\|\boldsymbol{W}_k\|_F^2 + \|\boldsymbol{W}_q\|_F^2).$$

Here, we set λ to be the the smallest representable number, e.g. computed as $1 + \lambda \neq 1$ in `Python`, which is around 2.22×10^{-16}. Therefore, $\boldsymbol{W}_k, \boldsymbol{W}_q$ parameters are updated as follows.

$$\boldsymbol{W}_k(k+1) = \boldsymbol{W}_k(k) - \eta \frac{\nabla \widetilde{\mathcal{L}}_{\boldsymbol{W}_k}(\boldsymbol{W}_k(k), \boldsymbol{W}_q(k))}{\|\nabla \widetilde{\mathcal{L}}_{\boldsymbol{W}_k}(\boldsymbol{W}_k(k), \boldsymbol{W}_q(k))\|_F}, \quad \boldsymbol{W}_q(k+1) = \boldsymbol{W}_q(k) - \eta \frac{\nabla \widetilde{\mathcal{L}}_{\boldsymbol{W}_q}(\boldsymbol{W}_k(k), \boldsymbol{W}_q(k))}{\|\nabla \widetilde{\mathcal{L}}_{\boldsymbol{W}_q}(\boldsymbol{W}_k(k), \boldsymbol{W}_q(k))\|_F}.$$

• As observed in previous work [186], and due to the exponential expression of softmax nonlinearity and computation limitation, `PyTorch` has no guarantee to select optimal tokens when the score gap is too small. Therefore in Figures 3.4 and 3.5, we generate random tokens making sure that $\min_{i \in [n], t \neq \text{opt}_i} \gamma_{i\text{opt}_i} - \gamma_{it} \geq \underline{\gamma}$ and we choose $\underline{\gamma} = 0.1$ in our experiments.

Rank sensitivity of $(\boldsymbol{W}_k, \boldsymbol{W}_q)$-parameterization (Figure B.1). In Figure 3.3 and Lemma 3.1, we have both theoretically and empirically established that the rank of the SVM



(a) τ *vs* λ (b) τ *vs* # selected tokens (c) Distribution of # selected

Figure B.3: Behavior of GD when selecting multiple tokens.

solution, denoted as $\boldsymbol{W}^{\mathrm{mm}}$ in (W-SVM) or $\boldsymbol{W}_{\star}^{\mathrm{mm}}$ in (W-SVM$_\star$), is at most rank $\max(n, d)$. Now, moving to Figure B.1, we delve into GD performance across various dimensions of $\boldsymbol{W}_k, \boldsymbol{W}_q \in \mathbb{R}^{d\times m}$ while keeping $d = 20$ fixed and varying m from 1 to 10. In the upper subfigure, we maintain a constant $n = 5$ and vary T within $\{5, 10, 15\}$, while in the lower subfigure, T is fixed at 5 and n changes within $\{5, 10, 15\}$. Results are depicted using blue, green, and red dashed curves, with both y-axes representing $1 - \texttt{corr\_coef}(\boldsymbol{W}, \boldsymbol{W}_{\star,\boldsymbol{\alpha}}^{\mathrm{mm}})$, where $\boldsymbol{W}$ represents the GD solution and $\boldsymbol{W}_{\star,\boldsymbol{\alpha}}^{\mathrm{mm}}$ is obtained from (W-SVM$_\star$) by employing token indices $\boldsymbol{\alpha}$ selected via GD and setting the rank limit to $m = d$. Observing both subfigures, we note that a larger n necessitates a larger m for attention weights $\boldsymbol{W}_k\boldsymbol{W}_q^\top$ to accurately converge to the SVM solution (Figure B.1(lower)). Meanwhile, performances remain consistent across varying T values (Figure B.1(upper)). This observation further validates Lemma 3.1. Furthermore, the results demonstrate that $\boldsymbol{W}$ converges directionally towards $\boldsymbol{W}_{\star,\boldsymbol{\alpha}}^{\mathrm{mm}}$ as long as $m \gtrsim n$, thereby confirming the assertion in our Theorem 3.5.

Behavior of GD with nonlinear nonconvex prediction head and multi-token compositions (Figure B.2). To better investigate how correlation changes with data dimension d, we collect the solid curves in Figure 3.9(upper) and construct as Figure B.2a. Moreover, Figure B.2b displays the average correlation of instances (refer to scatters in Figure 3.9 (lower)), considering masked tokens with softmax probability $< \Gamma$. Both findings highlight that higher d enhances alignment. For $d \geq 8$ or $\Gamma \leq 10^{-9}$, the GD solution $\boldsymbol{W}$ achieves a correlation of > 0.99 with the SVM-equivalence $\boldsymbol{W}^{\texttt{SVMeq}}$, defined in Section 3.6.

Investigation of Lemma 3.6 over different τ selections (Figure B.3). Consider the setting of Section 3.6.1 and Lemma 3.6. Figure 3.10 explores the influence of λ on the count of tokens selected by GD-derived attention weights. As λ increases, the likelihood of selecting

more tokens also increases. Shifting focus to Figure B.3, we examine the effect of τ. For each outcome, we generate random λ values, retaining pairs $(\lambda, \boldsymbol{X})$ satisfying τ constraints, with averages derived from 100 successful trials. The results indicate a positive correlation among τ, λ, and the number of selected tokens. Moreover, Figure B.3c provides a precise distribution of selected token counts across various τ values (specifically $\tau \in \{3, 5, 7, 9\}$). The findings confirm that the number of selected tokens remains within the limit of τ, thus validating the assertion made in Lemma 3.6.

APPENDIX C

Appendix for Chapter 4

C.1 Auxiliary Results

C.1.1 Soft-composition Component

In Section 4.3, we have theoretically shown that when training a single-layer self-attention model with gradient descent and log-loss function, once $\boldsymbol{W}^{\mathrm{mm}} \neq 0$, the composed attention weight $\boldsymbol{W}(\tau)$ will diverge in Frobenius norm and $\boldsymbol{W}(\tau)$ converges towards direction $\boldsymbol{W}^{\mathrm{mm}}/\|\boldsymbol{W}^{\mathrm{mm}}\|_F$; while in the subspace of $\mathcal{S}_{\mathrm{fin}}$, $\boldsymbol{\Pi}_{\mathcal{S}_{\mathrm{fin}}}(\boldsymbol{W}(\tau))$ converges to a finite $\boldsymbol{W}^{\mathrm{fin}}$ which is the unique solution minimizing the training loss over subspace $\mathcal{S}_{\mathrm{fin}}$ as described in Theorem 4.2. Here, $\boldsymbol{W}^{\mathrm{mm}}$ follows the solution of (Graph-SVM) and plays a role in separating tokens from different SCCs within the same TPG. Specifically, nodes satisfy $(i \Rightarrow j) \in \mathcal{G}^{(k)}$.

As for the nodes contained within the same SCC (e.g., $(i \asymp j)$), to ensure that i and j will not suppress each other, (Graph-SVM) solves the SVM problem with the constraint $(\boldsymbol{e}_i - \boldsymbol{e}_j)^\top \boldsymbol{W} \boldsymbol{e}_k = 0$. This essentially disregards the influence of distinct tokens within the same SCC. Consequently, $\boldsymbol{W}^{\mathrm{mm}}$ does not truly capture the essence of the ERM solution. In the following, we introduce cyclic subdataset and the so-called *cyclic-component*, and an equivalence between the cyclic term and $\boldsymbol{W}^{\mathrm{fin}}$ can be established under mild assumptions.

Definition C.1 (Cyclic subdataset) *Given any training sample $(\boldsymbol{X}, y) \in \mathit{DSET}$, we obtain the corresponding sample $(\boldsymbol{X}', y) \in \overline{\mathit{DSET}}$ by removing all tokens in $\boldsymbol{X}$ that satisfy $(y \Rightarrow x)$ in the corresponding TPG.*

In short, cyclic subdataset focuses on the input tokens that are part of the same SCC as the label token. Fig. C.1(Right) presents the cyclic subdataset $\overline{\texttt{DSET}}$ of DSET given in Fig. C.1(Left), which is the same as Fig. 4.3. In $\mathcal{G}^{(1)}$, all nodes are separated into different SCCs, and therefore, none of them is present in $\overline{\texttt{DSET}}$; while in $\mathcal{G}^{(2)}$, token $\boldsymbol{e}_1, \boldsymbol{e}_2$ and $\boldsymbol{e}_3$ are reachable from each other, and then are utilized to construct $\overline{\texttt{DSET}}$ while $\boldsymbol{e}_4$ is removed from

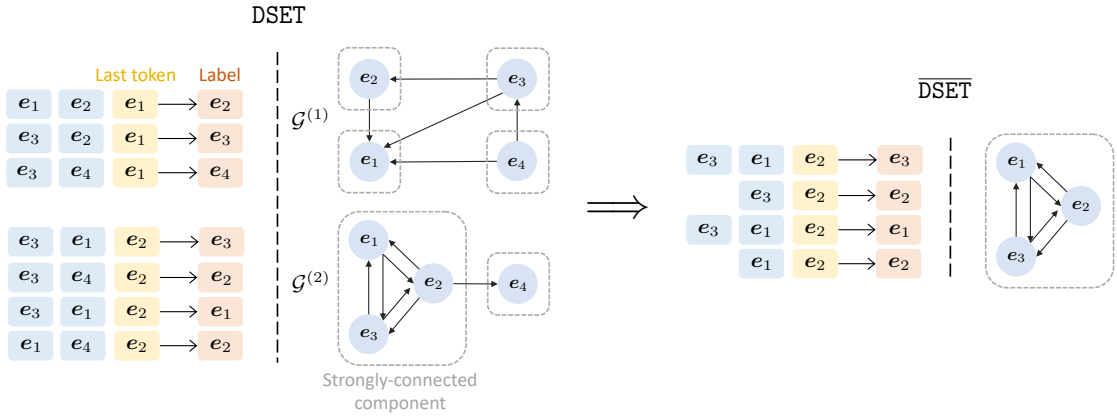


Figure C.1: Illustration of token-priority graph (TPG). Given the input sequences and labels (next tokens), we construct the TPGs $\{\mathcal{G}^{(k)}\}_{k=1}^{K}$ according to the last token. **Left hand side:** Two TPGs $\mathcal{G}^{(1)}$ and $\mathcal{G}^{(2)}$ are constructed using the samples with $\boldsymbol{e}_1$ and $\boldsymbol{e}_2$ as the last tokens, respectively. In each graph, directed edges (label→input token) are added, and based on the token relations, each graph can be partitioned into different strongly-connected components (SCCs, highlighted as dashed grey rectangles). **Right hand side:** We consider a cyclic subdataset $\overline{\texttt{DSET}}$ by removing all the singleton SCCs, as well as their corresponding edges (see Definition C.1). Then, $\overline{\texttt{DSET}}$ contains non-singleton SCCs only as shown on the right.

the dataset. Note that $\overline{\texttt{DSET}}$ provides a self-contained sub-problem that solely focuses on intra-SCC edges.

Definition C.2 (Cyclic component) *$\mathcal{W}^{fin}$ is obtained as the solution set of the ERM problem over the cyclic subdataset $\overline{\texttt{DSET}}$ per Definition C.1. Concretely,*

$$\mathcal{W}^{fin} = \arg\min_{\boldsymbol{W}\in\mathcal{S}_{fin}} \bar{\mathcal{L}}(\boldsymbol{W})$$
$$\textit{where} \quad \bar{\mathcal{L}}(\boldsymbol{W}) = \frac{1}{n}\sum_{(\boldsymbol{X},y)\in\overline{\texttt{DSET}}} \ell(\boldsymbol{c}_y^\top \boldsymbol{X}^\top \mathbb{S}(\boldsymbol{X}\boldsymbol{W}\bar{\boldsymbol{x}})).$$

Lemma C.1 *Consider a dataset $\texttt{DSET}$ and let $\boldsymbol{W}^{mm}$ be the corresponding SVM solution of* (Graph-SVM) *with $\boldsymbol{W}^{mm} \neq 0$. Then we have $\boldsymbol{W}^{mm} \perp \mathcal{S}_{fin}$, and for any $\bar{\boldsymbol{W}}^{fin} \neq 0 \in \mathcal{W}^{fin}$, $\bar{\boldsymbol{W}}^{fin}$ and $\boldsymbol{W}^{mm}$ are orthogonal.*

Lemma C.2 *Let $\boldsymbol{W} \in \mathbb{R}^{d\times d}$ be an arbitrary matrix, then we have $\bar{\mathcal{L}}(\boldsymbol{W}) = \bar{\mathcal{L}}(\boldsymbol{\Pi}_{\mathcal{S}_{fin}}(\boldsymbol{W}))$.*

Lemma C.3 *Suppose Assumptions 4.1 and 4.2 hold, and loss function $\ell(u) = -\log(u)$, then for any finite $\boldsymbol{W}$, $\tilde{\mathcal{L}}(\boldsymbol{W}) = \bar{\mathcal{L}}(\boldsymbol{W})$ where $\tilde{\mathcal{L}}(\boldsymbol{W})$ and $\bar{\mathcal{L}}(\boldsymbol{W})$ are defined in Theorem 4.2 and Definition C.2, respectively.*

C.1.2 Useful Notations

In this section, we introduce additional notations used in the subsequent proofs.

• **Token index sets $\mathcal{O}_i$, $\bar{\mathcal{O}}_i$, $\mathcal{R}_i$, $\bar{\mathcal{R}}_i$, $i \in [n]$.** Consider dataset $\texttt{DSET}$. Throughout, for any sample $(\boldsymbol{X}_i, y_i) \in \texttt{DSET}$, $i \in [n]$, we define

$$\mathcal{O}_i := \Big\{ t \;\Big|\; x_{it} = y_i, \forall t \in [T_i] \Big\} \quad \text{and} \quad \bar{\mathcal{O}}_i = [T_i] - \mathcal{O}_i, \tag{C.1a}$$

$$\mathcal{R}_i := \mathcal{O}_i \bigcup \Big\{ t \;\Big|\; (x_{it} \asymp y_i) \in \mathcal{G}^{(\bar{x}_i)}, \forall t \in [T_i] \Big\} \quad \text{and} \quad \bar{\mathcal{R}}_i = [T_i] - \mathcal{R}_i \tag{C.1b}$$

where x_{it} is the token ID of $\boldsymbol{x}_{it}$, T_i is the number of tokens in the input sequence $\boldsymbol{X}_i$ and $\mathcal{G}^{(\bar{x}_i)}$ is the corresponding token-priority graph (TPG) associated with the last/query token of $\boldsymbol{X}_i$. Concretely, $\mathcal{O}_i$ returns the token indices of i-th input that have the same token ID as label y_i, while $\mathcal{R}_i$ returns the token indices of i-th input that are included in the same strongly-connected component (SCC) as label y_i in the corresponding TPG. Then for any $t \in \bar{\mathcal{R}}_i$, we have $(y_i \Rightarrow x_{it}) \in \mathcal{G}^{(\bar{x}_i)}$. Take the last input sequence in Figure C.1(left) as an example, where $\mathcal{O} = \{3\}$ and $\mathcal{R} = \{1, 3\}$.

• **Datasets $\texttt{DSET}$, $\overline{\texttt{DSET}}$ and sample index set $\mathcal{I}$, $\bar{\mathcal{I}}$.** Recap the training dataset $\texttt{DSET} = (\boldsymbol{X}_i, y_i)_{i=1}^n$. Based on the relationships between input tokens and label token, following instructions in Section 4.2.1 we can construct the TPGs of dataset $\texttt{DSET}$. Then, let $\mathcal{I} \subseteq [n]$ be the sample index set such that for any $i \in \mathcal{I}$, $\boldsymbol{X}_i$ contains distinct tokens from the same SCC as label y_i in their corresponding TPG. Or equivalently,

$$\mathcal{I} = \Big\{ i \;\Big|\; \mathcal{R}_i - \mathcal{O}_i \neq \emptyset, i \in [n] \Big\} \quad \text{and} \quad \bar{\mathcal{I}} = [n] - \mathcal{I}. \tag{C.2}$$

Then the cyclic subset defined in Definition C.1 can be written by

$$\overline{\texttt{DSET}} = (\bar{\boldsymbol{X}}_i, y_i)_{i \in \mathcal{I}}, \tag{C.3}$$

where $\bar{\boldsymbol{X}}_i$ is obtained by removing all input tokens of $\boldsymbol{X}_i$ that are in the different SCCs from the label token y_i, or equivalently, removing $\boldsymbol{x}_{it}$, $t \in \bar{\mathcal{R}}_i$. Hence, for all $i \in \bar{\mathcal{I}}$, $\boldsymbol{X}_i$ only contains input tokens (ignoring the ones with the same token ID as label) that have strictly lower priority than its label token, i.e., $(y_i \Rightarrow x_{it}) \in \mathcal{G}^{(\bar{x}_i)}$ for $t \in \bar{\mathcal{O}}_i$. In Figure C.1(left), we have $\mathcal{I} = \{4, 5, 6, 7\}$ and $\bar{\mathcal{I}} = \{1, 2, 3\}$.

• **Token scores $\gamma_i, i \in [n]$ and loss $\mathcal{L}(\boldsymbol{W})$ under Assumption 4.1.** Let $\gamma_i = \boldsymbol{X}_i \boldsymbol{c}_{y_i}$ be

the token score vectors. Then under Assumption 4.1, we have

$$\gamma_{it} = \begin{cases} 1, & t \in \mathcal{O}_i \\ 0, & t \in \bar{\mathcal{O}}_i \end{cases} \qquad \text{for all } i \in [n]. \tag{C.4}$$

Additionally, letting $\boldsymbol{s}_i^{\boldsymbol{W}} = \mathbb{S}(\boldsymbol{X}_i \boldsymbol{W} \bar{\boldsymbol{x}}_i)$, we can rewrite the training risk as follows:

$$\mathcal{L}(\boldsymbol{W}) = \frac{1}{n} \sum_{i=1}^{n} \ell \left(\sum_{t \in \mathcal{O}_i} s_{it}^{\boldsymbol{W}} \right). \tag{C.5}$$

C.1.3 Proof of Lemma 4.1

Proof. Recap the constraints in (Graph-SVM) problem:

$$(\boldsymbol{e}_i - \boldsymbol{e}_j)^\top \boldsymbol{W} \boldsymbol{e}_k \begin{cases} = 0 & \forall (i \asymp j) \in \mathcal{G}^{(k)} \\ \geq 1 & \forall (i \Rightarrow j) \in \mathcal{G}^{(k)} \end{cases} \quad \text{for all} \quad k \in [K]. \tag{C.6}$$

Since $\boldsymbol{E} = [\boldsymbol{e}_1 \ \boldsymbol{e}_2 \ \cdots \ \boldsymbol{e}_K]^\top \in \mathbb{R}^{K \times d}$ is full row rank, then $K \leq d$ and $\boldsymbol{e}_k, k \in [K]$ are linearly independent. Let $\bar{\boldsymbol{E}} \in \mathbb{R}^{K \times d}$ satisfying $\bar{\boldsymbol{E}} \boldsymbol{E}^\top = \boldsymbol{I}$. Then for any $\bar{\boldsymbol{W}} \in \mathbb{R}^{K \times K}$, we get

$$(\boldsymbol{e}_i - \boldsymbol{e}_j)^\top \bar{\boldsymbol{E}}^\top \bar{\boldsymbol{W}} \bar{\boldsymbol{E}} \boldsymbol{e}_k \begin{cases} = 0 & \forall (i \asymp j) \in \mathcal{G}^{(k)} \\ \geq 1 & \forall (i \Rightarrow j) \in \mathcal{G}^{(k)} \end{cases} \quad \text{for all} \quad k \in [K] \tag{C.7}$$

and feasibility of (C.7) implies $\boldsymbol{W} \in \mathbb{R}^{d \times d}$ in (C.6) is feasible. Since we can set $\boldsymbol{W} = \bar{\boldsymbol{E}}^\top \bar{\boldsymbol{W}} \bar{\boldsymbol{E}}$. Next let $\boldsymbol{u}_i = \bar{\boldsymbol{E}} \boldsymbol{e}_i$ $i \in [K]$ be K-dimensional one-hot vectors. Then it remains to show that there exists $\bar{\boldsymbol{W}} \in \mathbb{R}^{K \times K}$ such that

$$(\boldsymbol{u}_i - \boldsymbol{u}_j)^\top \bar{\boldsymbol{W}} \boldsymbol{u}_k \begin{cases} = 0 & \forall (i \asymp j) \in \mathcal{G}^{(k)} \\ \geq 1 & \forall (i \Rightarrow j) \in \mathcal{G}^{(k)} \end{cases} \quad \text{for all} \quad k \in [K] \tag{C.8}$$

is feasible. Additionally, it is equivalent with showing that for any $k \in [K]$, there exists $\boldsymbol{w} \in \mathbb{R}^K$, such that

$$(\boldsymbol{u}_i - \boldsymbol{u}_j)^\top \boldsymbol{w} \begin{cases} = 0 & \forall (i \asymp j) \in \mathcal{G}^{(k)} \\ \geq 1 & \forall (i \Rightarrow j) \in \mathcal{G}^{(k)}. \end{cases} \tag{C.9}$$

To start with, we first derive the priority order of each graph (referring to the topological sorting of directed graph). Let M_i be the order of $\boldsymbol{u}_i$ where M_i's $i \in [K]$ are positive integers.

Then if $(i \asymp j)$, $M_i = M_j$; if $(i \Rightarrow j)$, $M_i > M_j$. Then let $\boldsymbol{w} = \sum_{i\in[K]} M_i \boldsymbol{u}_i$. We obtain that for any $k \in [K]$,

$$\forall (i \asymp j) \in \mathcal{G}^{(k)},\ (\boldsymbol{u}_i - \boldsymbol{u}_j)^\top \boldsymbol{w} = (\boldsymbol{u}_i - \boldsymbol{u}_j)^\top (M_i \boldsymbol{u}_i + M_j \boldsymbol{u}_j) = M_i - M_j = 0$$
$$\forall (i \Rightarrow j) \in \mathcal{G}^{(k)},\ (\boldsymbol{u}_i - \boldsymbol{u}_j)^\top \boldsymbol{w} = (\boldsymbol{u}_i - \boldsymbol{u}_j)^\top (M_i \boldsymbol{u}_i + M_j \boldsymbol{u}_j) = M_i - M_j \geq 1$$

which indicates that (C.9) is feasible for any $k \in [K]$ and it completes the proof. ∎

C.1.4 Proof of Lemma C.1

Proof. Recall from Definition 4.1 that $\mathcal{S}_{\text{fin}}$ is the span of all matrices $(\boldsymbol{e}_i - \boldsymbol{e}_j)\boldsymbol{e}_k^\top$ for all $(i \asymp j) \in \mathcal{G}^{(k)}$ and $k \in [K]$. Then for any matrix $\boldsymbol{W} \in \mathcal{S}_{\text{fin}}$, there exist a_{ijk}'s satisfying

$$\boldsymbol{W} = \sum_{i,j,k} a_{ijk} (\boldsymbol{e}_i - \boldsymbol{e}_j)\boldsymbol{e}_k^\top$$

where $(i \asymp j) \in \mathcal{G}^{(k)}$ and $k \in [K]$. Since $\boldsymbol{W}^{\text{mm}}$ is the solution of (Graph-SVM) that satisfies the all "$= 0$" constraints, for any matrix $\boldsymbol{W} \in \mathcal{S}_{\text{fin}}$, we have

$$\langle \boldsymbol{W}^{\text{mm}}, \boldsymbol{W} \rangle = \sum_{i,j,k} a_{ijk} \left\langle \boldsymbol{W}^{\text{mm}}, (\boldsymbol{e}_i - \boldsymbol{e}_j)\boldsymbol{e}_k^\top \right\rangle = 0.$$

Therefore, $\boldsymbol{W}^{\text{mm}} \perp \mathcal{S}_{\text{fin}}$. ∎

C.1.5 Proof of Lemma C.2

Proof. Recap the definition of $\bar{\mathcal{L}}(\boldsymbol{W})$ from Def. C.2 and $\mathcal{I}$, $\bar{\boldsymbol{X}}_i$ from (C.2), (C.3). Then

$$\bar{\mathcal{L}}(\boldsymbol{W}) = \frac{1}{n} \sum_{i\in\mathcal{I}} \ell(\boldsymbol{c}_{y_i}^\top \bar{\boldsymbol{X}}_i^\top \mathbb{S}(\bar{\boldsymbol{X}}_i \boldsymbol{W} \bar{\boldsymbol{x}}_i)).$$

Let $\boldsymbol{W}^\perp = \boldsymbol{\Pi}_{\mathcal{S}_{\text{fin}}^\perp}(\boldsymbol{W})$ and $\boldsymbol{W}^\parallel = \boldsymbol{\Pi}_{\mathcal{S}_{\text{fin}}}(\boldsymbol{W})$. Then it remains to show that for any $(\bar{\boldsymbol{X}}, y) \in \overline{\texttt{DSET}}$, $\mathbb{S}(\bar{\boldsymbol{X}} \boldsymbol{W} \bar{\boldsymbol{x}}) = \mathbb{S}(\bar{\boldsymbol{X}} \boldsymbol{W}^\parallel \bar{\boldsymbol{x}})$.

For simplification, let $\bar{\boldsymbol{x}} = \boldsymbol{e}_k$, and following the definition of TPG, SCC and $\overline{\texttt{DSET}}$, we have that all tokens $\boldsymbol{x} \in \bar{\boldsymbol{X}}$ are in the same SCC and denote the token set as $\mathcal{C}^{(k)}$. Then $\mathcal{S}_{\text{fin}}$ spans the matrices $(\boldsymbol{e}_i - \boldsymbol{e}_j)\boldsymbol{e}_k^\top$ for $i, j \in \mathcal{C}^{(k)}$. For any $i \in \mathcal{C}^{(k)}$, we get

$$\boldsymbol{e}_i^\top \boldsymbol{W} \boldsymbol{e}_k = \boldsymbol{e}_i^\top \boldsymbol{W}^\parallel \boldsymbol{e}_k + \boldsymbol{e}_i^\top \boldsymbol{W}^\perp \boldsymbol{e}_k.$$

Next, let $a_{ik} = \boldsymbol{e}_i^\top \boldsymbol{W}^\perp \boldsymbol{e}_k$, $i \in \mathcal{C}^{(k)}$. Since $\boldsymbol{W}^\perp \perp \mathcal{S}_{\text{fin}}$, and $(\boldsymbol{e}_i - \boldsymbol{e}_j)\boldsymbol{e}_k^\top \in \mathcal{S}_{\text{fin}}$, we obtain

$$\begin{aligned} & (\boldsymbol{e}_i - \boldsymbol{e}_j)^\top \boldsymbol{W}^\perp \boldsymbol{e}_k = 0 \\ \implies & \boldsymbol{e}_i^\top \boldsymbol{W}^\perp \boldsymbol{e}_k - \boldsymbol{e}_j^\top \boldsymbol{W}^\perp \boldsymbol{e}_k = 0 \\ \implies & a_{ik} - a_{jk} = 0 \\ \implies & a_{ik} = a_{jk} =: \bar{a}_k. \end{aligned} \tag{C.10}$$

Then we have that for any $\boldsymbol{x} \in \bar{\boldsymbol{X}}$, $\boldsymbol{x}^\top \boldsymbol{W}^\perp \bar{\boldsymbol{x}} = \bar{a}_k$ where $\bar{a}_k$ is associated with the last/query token $\bar{\boldsymbol{x}}$ and hence

$$\begin{aligned} \bar{\boldsymbol{X}} \boldsymbol{W} \bar{\boldsymbol{x}} &= \bar{\boldsymbol{X}} \boldsymbol{W}^\parallel \bar{\boldsymbol{x}} + \bar{\boldsymbol{X}} \boldsymbol{W}^\perp \bar{\boldsymbol{x}} = \bar{\boldsymbol{X}} \boldsymbol{W}^\parallel \bar{\boldsymbol{x}} + \bar{a}_k \mathbf{1} \\ \mathbb{S}(\bar{\boldsymbol{X}} \boldsymbol{W} \bar{\boldsymbol{x}}) &= \mathbb{S}(\bar{\boldsymbol{X}} \boldsymbol{W}^\parallel \bar{\boldsymbol{x}} + \bar{a}_k \mathbf{1}) = \mathbb{S}(\bar{\boldsymbol{X}} \boldsymbol{W}^\parallel \bar{\boldsymbol{x}}), \end{aligned}$$

which completes the proof. ■

C.1.6 Proof of Lemma C.3

In the following, we present an additional lemma that incorporates Lemma C.3.

Lemma C.4 *Suppose Assumptions 4.1 and 4.2 hold and loss function $\ell : \mathbb{R} \to \mathbb{R}$ is strictly decreasing. For any finite $\boldsymbol{W}$, once $\overline{\texttt{DSET}} \neq \texttt{DSET}$, we have that*

$$\mathcal{L}(\boldsymbol{W}) > \frac{|\bar{\mathcal{I}}|}{n} \ell(1) + \bar{\mathcal{L}}(\boldsymbol{W}). \tag{C.11}$$

Additionally, we have

$$\min_{\boldsymbol{W}' \in \mathcal{S}_{fin}^\perp} \mathcal{L}(\boldsymbol{W}' + \boldsymbol{W}) = \tilde{\mathcal{L}}(\boldsymbol{W}) = \frac{|\bar{\mathcal{I}}|}{n} \ell(1) + \bar{\mathcal{L}}(\boldsymbol{W}), \tag{C.12a}$$

$$\min_{\boldsymbol{W}} \mathcal{L}(\boldsymbol{W}) = \min_{\boldsymbol{W}} \tilde{\mathcal{L}}(\boldsymbol{W}) = \frac{|\bar{\mathcal{I}}|}{n} \ell(1) + \min_{\boldsymbol{W}} \bar{\mathcal{L}}(\boldsymbol{W}). \tag{C.12b}$$

Proof. We start with proving that for any finite $\boldsymbol{W}$, $\mathcal{L}(\boldsymbol{W}) \geq \frac{|\bar{\mathcal{I}}|}{n} \ell(1) + \bar{\mathcal{L}}(\boldsymbol{W})$. Let $\boldsymbol{W}^\perp = \boldsymbol{\Pi}_{\mathcal{S}_{\text{fin}}^\perp}(\boldsymbol{W})$, $\boldsymbol{W}^\parallel = \boldsymbol{\Pi}_{\mathcal{S}_{\text{fin}}}(\boldsymbol{W})$ where we have $\boldsymbol{W} = \boldsymbol{W}^\perp + \boldsymbol{W}^\parallel$. Let $\boldsymbol{a}_i = \boldsymbol{X}_i \boldsymbol{W}^\perp \bar{\boldsymbol{x}}_i$, $\boldsymbol{b}_i = \boldsymbol{X}_i \boldsymbol{W}^\parallel \bar{\boldsymbol{x}}_i$ and $\boldsymbol{s}_i = \mathbb{S}(\boldsymbol{X}_i \boldsymbol{W} \bar{\boldsymbol{x}}_i) = \mathbb{S}(\boldsymbol{a}_i + \boldsymbol{b}_i)$. Following (C.5) and the definition of $\bar{\mathcal{L}}(\boldsymbol{W})$, training losses obey

$$\mathcal{L}(\boldsymbol{W}) = \frac{1}{n} \sum_{i=1}^{n} \ell \left(\sum_{t \in \mathcal{O}_i} s_{it} \right) \quad \text{and} \quad \bar{\mathcal{L}}(\boldsymbol{W}) = \frac{1}{n} \sum_{i \in \mathcal{I}} \ell \left(\frac{\sum_{t \in \mathcal{O}_i} s_{it}}{\sum_{t \in \mathcal{R}_i} s_{it}} \right).$$

From proof of Lemma C.2 (more specifically (C.10)), we have that for any $t \in \mathcal{R}_i$,

$$\boldsymbol{x}_{it}^\top \boldsymbol{W}^\perp \bar{\boldsymbol{x}}_i = \bar{a}_i \implies a_{it} = \bar{a}_i, \ \forall t \in \mathcal{R}_i$$

where $\bar{a}_i$ is some constant associated with $\boldsymbol{W}^\perp$. Then for any $i \in [n]$, we get

$$\sum_{t\in\mathcal{O}_i} s_{it} = \frac{\sum_{t\in\mathcal{O}_i} e^{\bar{a}_i+b_{it}}}{\sum_{t\in\mathcal{R}_i} e^{\bar{a}_i+b_{it}} + \sum_{t\in\bar{\mathcal{R}}_i} e^{a_{it}+b_{it}}} = \frac{\sum_{t\in\mathcal{O}_i} e^{b_{it}}}{\sum_{t\in\mathcal{R}_i} e^{b_{it}} + \sum_{t\in\bar{\mathcal{R}}_i} e^{b_{it}+a_{it}-\bar{a}_i}} \le \frac{\sum_{t\in\mathcal{O}_i} e^{b_{it}}}{\sum_{t\in\mathcal{R}_i} e^{b_{it}}},$$
$$\frac{\sum_{t\in\mathcal{O}_i} s_{it}}{\sum_{t\in\mathcal{R}_i} s_{it}} = \frac{\sum_{t\in\mathcal{O}_i} e^{\bar{a}_i+b_{it}}}{\sum_{t\in\mathcal{R}_i} e^{\bar{a}_i+b_{it}}} = \frac{\sum_{t\in\mathcal{O}_i} e^{b_{it}}}{\sum_{t\in\mathcal{R}_i} e^{b_{it}}}.$$

Next following (C.2) we have that for $i \in \bar{\mathcal{I}}$, $\mathcal{R}_i = \mathcal{O}_i$ and therefore

$$\frac{\sum_{t\in\mathcal{O}_i} s_{it}}{\sum_{t\in\mathcal{R}_i} s_{it}} = 1 \quad \text{for all} \quad i \in \bar{\mathcal{I}}.$$

Note that since a_{it}, b_{it} are finite and $\overline{\texttt{DSET}} \neq \texttt{DSET}$, there exists $i \in [n]$ such that $\sum_{t\in\mathcal{O}_i} s_{it} < \frac{\sum_{t\in\mathcal{O}_i} s_{it}}{\sum_{t\in\mathcal{R}_i} s_{it}}$. Given strictly decreasing loss function ℓ and any finite $\boldsymbol{W}$, the training risks obey

$$\mathcal{L}(\boldsymbol{W}) = \frac{1}{n}\sum_{i=1}^n \ell\left(\sum_{i\in\mathcal{O}_i} s_{it}\right) > \frac{1}{n}\sum_{i\in\bar{\mathcal{I}}} \ell(1) + \frac{1}{n}\sum_{i\in\mathcal{I}} \ell\left(\frac{\sum_{t\in\mathcal{O}_i} s_{it}}{\sum_{t\in\mathcal{R}_i} s_{it}}\right) = \frac{|\bar{\mathcal{I}}|}{n}\ell(1) + \bar{\mathcal{L}}(\boldsymbol{W}).$$

It completes the proof of (C.11).

We next show that

$$\min_{\boldsymbol{W}'\in\mathcal{S}_{\text{fin}}^\perp} \mathcal{L}(\boldsymbol{W}' + \boldsymbol{W}) = \tilde{\mathcal{L}}(\boldsymbol{W}) = \frac{|\bar{\mathcal{I}}|}{n}\ell(1) + \bar{\mathcal{L}}(\boldsymbol{W}).$$

Recap from Theorem 4.2 that $\tilde{\mathcal{L}}(\boldsymbol{W}) = \lim_{R\to\infty} \mathcal{L}(\boldsymbol{W} + R\cdot\boldsymbol{W}^{\text{mm}})$. Let $\boldsymbol{W}^\perp = \boldsymbol{\Pi}_{\mathcal{S}_{\text{fin}}^\perp}(\boldsymbol{W})$, $\boldsymbol{W}^\parallel = \boldsymbol{\Pi}_{\mathcal{S}_{\text{fin}}}(\boldsymbol{W})$ where we have $\boldsymbol{W} = \boldsymbol{W}^\perp + \boldsymbol{W}^\parallel$. Let $\boldsymbol{a}_i = \boldsymbol{X}_i\boldsymbol{W}^\perp\bar{\boldsymbol{x}}_i$, $\boldsymbol{b}_i = \boldsymbol{X}_i\boldsymbol{W}^\parallel\bar{\boldsymbol{x}}_i$, $\boldsymbol{c}_i = \boldsymbol{X}_i\boldsymbol{W}^{\text{mm}}\bar{\boldsymbol{x}}_i$ and $\boldsymbol{s}_i^R = \mathbb{S}(\boldsymbol{X}_i(R\cdot\boldsymbol{W}^{\text{mm}} + \boldsymbol{W})\bar{\boldsymbol{x}}_i) = \mathbb{S}(\boldsymbol{a}_i + \boldsymbol{b}_i + R\cdot\boldsymbol{c}_i)$. Similarly, for any $t \in \mathcal{R}_i$,

$$\boldsymbol{x}_{it}^\top \boldsymbol{W}^\perp \bar{\boldsymbol{x}}_i = \bar{a}_i \implies a_{it} = \bar{a}_i, \ \forall t \in \mathcal{R}_i$$

where $\bar{a}_i$ is some constant associated with $\boldsymbol{W}^\perp$. Additionally, since $\boldsymbol{W}^{\text{mm}}$ follows

(Graph-SVM), we have

$$(\boldsymbol{x}_{i\tau} - \boldsymbol{x}_{it})^\top \boldsymbol{W}^{\mathrm{mm}} \bar{\boldsymbol{x}}_i \begin{cases} = 0 & \forall t \in \mathcal{R}_i \\ \geq 1 & \forall t \in \bar{\mathcal{R}}_i \end{cases}, \quad \text{for all } \tau \in \mathcal{R}_i \quad \Longrightarrow \quad \begin{cases} c_{it} = \bar{c}_i, & \forall t \in \mathcal{R}_i, \\ c_{it} \leq \bar{c}_i - 1, & \forall t \in \bar{\mathcal{R}}_i. \end{cases} \tag{C.13}$$

Then for any $i \in [n]$, we get

$$\begin{aligned} \sum_{t \in \mathcal{O}_i} s_{it}^R &= \frac{\sum_{t \in \mathcal{O}_i} e^{\bar{a}_i + b_{it} + R\bar{c}_i}}{\sum_{t \in \mathcal{R}_i} e^{\bar{a}_i + b_{it} + R\bar{c}_i} + \sum_{t \in \bar{\mathcal{R}}_i} e^{a_{it} + b_{it} + Rc_{it}}} \\ &= \frac{\sum_{t \in \mathcal{O}_i} e^{b_{it}}}{\sum_{t \in \mathcal{R}_i} e^{b_{it}} + \sum_{t \in \bar{\mathcal{R}}_i} e^{b_{it} + a_{it} - \bar{a}_i + R(c_{it} - \bar{c}_i)}} \\ &\leq \frac{\sum_{t \in \mathcal{O}_i} e^{b_{it}}}{\sum_{t \in \mathcal{R}_i} e^{b_{it}}}. \end{aligned}$$

Case 1: $\boldsymbol{W}^{\mathrm{mm}} = 0$. Then for all $i \in [n]$, $\bar{\mathcal{R}}_i = \emptyset$ and the equality holds for all $i \in [n]$.
Case 2: $\boldsymbol{W}^{\mathrm{mm}} \neq 0$. Since a_{it}, b_{it} are finite and $c_{it} - \bar{c}_i \leq -1$ for $t \in \bar{\mathcal{R}}_i$ following (C.13), the equality holds when $R \to \infty$, and therefore we have for any $i \in [n]$,

$$\lim_{R \to \infty} \sum_{t \in \mathcal{O}_i} s_{it}^R = \frac{\sum_{t \in \mathcal{O}_i} e^{b_{it}}}{\sum_{t \in \mathcal{R}_i} e^{b_{it}}}$$

and

$$\begin{aligned} \tilde{\mathcal{L}}(\boldsymbol{W}) = \lim_{R \to \infty} \mathcal{L}(R \cdot \boldsymbol{W}^{\mathrm{mm}} + \boldsymbol{W}) &= \lim_{R \to \infty} \frac{1}{n} \sum_{i=1}^n \ell\left(\sum_{i \in \mathcal{O}_i} s_{it}^R \right) \\ &= \frac{|\bar{\mathcal{I}}|}{n} \ell(1) + \frac{1}{n} \sum_{i \in \mathcal{I}} \ell\left(\frac{\sum_{t \in \mathcal{O}_i} e^{b_{it}}}{\sum_{t \in \mathcal{R}_i} e^{b_{it}}} \right) \\ &= \frac{|\bar{\mathcal{I}}|}{n} \ell(1) + \bar{\mathcal{L}}(\boldsymbol{W}). \end{aligned} \tag{C.14}$$

Additionally, we have for any $\boldsymbol{W}' \in \mathcal{S}_{\mathrm{fin}}^\perp$,

$$\mathcal{L}(\boldsymbol{W}' + \boldsymbol{W}) \geq \frac{|\bar{\mathcal{I}}|}{n} \ell(1) + \bar{\mathcal{L}}(\boldsymbol{W}' + \boldsymbol{W}) = \frac{|\bar{\mathcal{I}}|}{n} \ell(1) + \bar{\mathcal{L}}(\boldsymbol{W})$$

where the inequality uses (C.11) and the equality comes from Lemma C.2. Since bound is achievable (by choosing $\boldsymbol{W}' = \lim_{R \to \infty} R \cdot \boldsymbol{W}^{\mathrm{mm}}$ as in (C.14)), then combining it with (C.14) completes the proof of (C.12a). (C.12b) is directly obtained from (C.12a).

■

C.2 Global Convergence of Gradient Descent

C.2.1 Supporting Results under the Setting of Theorem 4.2

In this section, we introduce results useful for the main proof. Recap the setting of Theorem 4.2 where $\ell(u) = -\log(u)$. Therefore loss defined in (C.5) is

$$\mathcal{L}(\boldsymbol{W}) = -\frac{1}{n}\sum_{i=1}^{n}\log\left(\sum_{t\in\mathcal{O}_i} s_{it}^{\boldsymbol{W}}\right) \tag{C.15}$$

where $\boldsymbol{s}_i^{\boldsymbol{W}} = \mathbb{S}(\boldsymbol{X}_i\boldsymbol{W}\bar{\boldsymbol{x}}_i)$ and $\mathcal{O}_i$'s follow (C.1).

• **$\nabla\mathcal{L}(\boldsymbol{W})$ under the setting of Theorem 4.2.** For any $\boldsymbol{W}\in\mathbb{R}^{d\times d}$, let $\boldsymbol{h}_i = \boldsymbol{X}_i\boldsymbol{W}\bar{\boldsymbol{x}}_i, \boldsymbol{s}_i = \mathbb{S}(\boldsymbol{h}_i), \gamma_i = \boldsymbol{X}_i\boldsymbol{c}_{y_i}$.

$$\begin{aligned}\nabla\mathcal{L}(\boldsymbol{W}) &= \frac{1}{n}\sum_{i=1}^{n}\ell'(\gamma_i^\top\boldsymbol{s}_i)\boldsymbol{X}_i^\top\mathbb{S}'(\boldsymbol{h}_i)\gamma_i\bar{\boldsymbol{x}}_i^\top\\ &= \frac{1}{n}\sum_{i=1}^{n}-\frac{1}{\gamma_i^\top\boldsymbol{s}_i}\boldsymbol{X}_i^\top(\mathrm{diag}(\boldsymbol{s}_i)-\boldsymbol{s}_i\boldsymbol{s}_i^\top)\gamma_i\bar{\boldsymbol{x}}_i^\top\\ &= \frac{1}{n}\sum_{i=1}^{n}\sum_{t\in\bar{\mathcal{O}}_i}s_{it}(\boldsymbol{x}_{it}-\boldsymbol{e}_{y_i})\bar{\boldsymbol{x}}_i^\top\end{aligned} \tag{C.16}$$

where the last equation uses the fact that for any example $(\boldsymbol{X}_i, y_i)\in\texttt{DSET}$, $i\in[n]$,

$$\begin{aligned}\frac{\boldsymbol{X}_i^\top(\mathrm{diag}(\boldsymbol{s}_i)-\boldsymbol{s}_i\boldsymbol{s}_i^\top)\gamma_i}{\gamma_i^\top\boldsymbol{s}_i} &= \frac{\boldsymbol{X}_i^\top\mathrm{diag}(\boldsymbol{s}_i)\gamma_i}{\gamma_i^\top\boldsymbol{s}_i}-\boldsymbol{X}_i^\top\boldsymbol{s}_i\\ &= \frac{\sum_{t\in\mathcal{O}_i}s_{it}\boldsymbol{e}_{y_i}}{\sum_{t\in\mathcal{O}_i}s_{it}}-\boldsymbol{X}_i^\top\boldsymbol{s}_i\\ &= \boldsymbol{e}_{y_i}-\boldsymbol{X}_i^\top\boldsymbol{s}_i\\ &= \sum_{t\in\bar{\mathcal{O}}_i}s_{it}(\boldsymbol{e}_{y_i}-\boldsymbol{x}_{it}).\end{aligned}$$

Here, the second equality comes from (C.4).

• **Lipschitzness of $\nabla\mathcal{L}(\boldsymbol{W})$ in** (C.16)**.** For any $\boldsymbol{W}, \dot{\boldsymbol{W}}\in\mathbb{R}^{d\times d}$, let $\boldsymbol{s}_i = \mathbb{S}(\boldsymbol{X}_i\boldsymbol{W}\bar{\boldsymbol{x}}_i)$ and $\dot{\boldsymbol{s}}_i = \mathbb{S}(\boldsymbol{X}_i\dot{\boldsymbol{W}}\bar{\boldsymbol{x}}_i)$. Consider bounded tokens and let $M := \max_{k\in[K]}\|\boldsymbol{e}_k\|$. Following (C.16),

we have:

$$\begin{aligned}\|\nabla\mathcal{L}(\boldsymbol{W})-\nabla\mathcal{L}(\dot{\boldsymbol{W}})\|_F &= \|\frac{1}{n}\sum_{i=1}^{n}\sum_{t\in\bar{\mathcal{O}}_i}(s_{it}-\dot{s}_{it})(\boldsymbol{x}_{it}-\boldsymbol{e}_{y_i})\bar{\boldsymbol{x}}_i^\top\|_F \\ &\le \frac{1}{n}\sum_{i=1}^{n}\sum_{t\in\bar{\mathcal{O}}_i}|s_{it}-\dot{s}_{it}|\ \|(\boldsymbol{x}_{it}-\boldsymbol{e}_{y_i})\bar{\boldsymbol{x}}_i^\top\|_F \\ &\le \frac{2M^2}{n}\sum_{i=1}^{n}\sum_{t\in\bar{\mathcal{O}}_i}|s_{it}-\dot{s}_{it}| \\ &\le \frac{2M^2}{n}\sum_{i=1}^{n}\|\boldsymbol{s}_i-\dot{\boldsymbol{s}}_i\|_1 \\ &\le \frac{2M^2}{n}\sum_{i=1}^{n}\sqrt{T_i}\cdot\|\boldsymbol{s}_i-\dot{\boldsymbol{s}}_i\|.\end{aligned} \tag{C.17}$$

Next for any $\boldsymbol{s},\dot{\boldsymbol{s}}$, we get

$$\begin{aligned}\|\boldsymbol{s}-\dot{\boldsymbol{s}}\| &= \|\mathbb{S}(\boldsymbol{X}\boldsymbol{W}\bar{\boldsymbol{x}})-\mathbb{S}(\boldsymbol{X}\dot{\boldsymbol{W}}\bar{\boldsymbol{x}})\| \\ &\le \|\boldsymbol{X}\boldsymbol{W}\bar{\boldsymbol{x}}-\boldsymbol{X}\dot{\boldsymbol{W}}\bar{\boldsymbol{x}}\| \\ &\le M^2\|\boldsymbol{W}-\dot{\boldsymbol{W}}\|_F.\end{aligned} \tag{C.18}$$

Combining results in that

$$\|\nabla\mathcal{L}(\boldsymbol{W})-\nabla\mathcal{L}(\dot{\boldsymbol{W}})\|_F \le 2M^4\sqrt{T_{\max}}\cdot\|\boldsymbol{W}-\dot{\boldsymbol{W}}\|_F \tag{C.19}$$

where $T_{\max}:=\max_{i\in[n]}T_i$. Then let

$$L := 2M^4\sqrt{T_{\max}} \tag{C.20}$$

and $\nabla\mathcal{L}(\boldsymbol{W})$ is L-Lipschitz continuous.

C.2.2 Proof of Lemma 4.2

In this subsection, we provide and prove a general version of Lemma 4.2. To to that, we first introduce the following new subspaces.

Definition C.3 *Define the subspace $\mathcal{S}_{active}$ as the span of all matrices $(\boldsymbol{e}_i-\boldsymbol{e}_j)\boldsymbol{e}_k^\top$ for all $(i\to j)\in\mathcal{G}^{(k)}$ and $k\in[K]$.*

Definition C.4 (Cyclic subspace (Restated)) *Define cyclic subspace $\mathcal{S}_{fin}$ as the span of all matrices $(\boldsymbol{e}_i-\boldsymbol{e}_j)\boldsymbol{e}_k^\top$ for all $(i\asymp j)\in\mathcal{G}^{(k)}$ and $k\in[K]$.*

Observe that $\mathcal{S}_{\text{fin}}$ is a subspace of $\mathcal{S}_{\text{active}}$. This is because if two nodes $i, j \in \mathcal{G}^{(k)}$ and $i \asymp j$, this also implies that $i \to j$ and $j \to i$.

Definition C.5 (SVM subspace) *Define svm subspace $\mathcal{S}_{svm}$ as the orthogonal complement of the subspace $\mathcal{S}_{fin}$ inside the subspace $\mathcal{S}_{active}$.*

Lemma C.5 *We have the following:*

(i) Recall the definition of $\boldsymbol{W}^{mm}$ in (Graph-SVM). *$\boldsymbol{W}^{mm} \in \mathcal{S}_{svm}$.*

(ii) Let $\mathcal{S}_{active}^{\perp}$ be the orthogonal complement of $\mathcal{S}_{active}$ inside $\mathbb{R}^{d\times d}$. Then,

$$\|\boldsymbol{\Pi}_{\mathcal{S}_{active}^{\perp}}(\nabla\mathcal{L}(\boldsymbol{W}))\|_F = 0, \quad \forall \boldsymbol{W} \in \mathbb{R}^{d\times d}.$$

Proof.

• **(i):** Recall the definition of $\boldsymbol{W}^{\text{mm}}$:

$$\begin{aligned}\boldsymbol{W}^{\text{mm}} &= \arg\min_{\boldsymbol{W}} \|\boldsymbol{W}\|_F \\ \text{s.t. } (\boldsymbol{e}_i - \boldsymbol{e}_j)^\top \boldsymbol{W} \boldsymbol{e}_k &\begin{cases} = 0 & \forall (i \asymp j) \in \mathcal{G}^{(k)} \\ \geq 1 & \forall (i \Rightarrow j) \in \mathcal{G}^{(k)} \end{cases} \quad \text{for all} \quad k \in [K].\end{aligned}$$

Assume that the statement is not correct. Then, either $\|\boldsymbol{\Pi}_{\mathcal{S}_{\text{fin}}}(\boldsymbol{W}^{\text{mm}})\|_F > 0$ or $\|\boldsymbol{\Pi}_{\mathcal{S}_{\text{active}}^{\perp}}(\boldsymbol{W}^{\text{mm}})\|_F > 0$.

By definition, $\|\boldsymbol{\Pi}_{\mathcal{S}_{\text{fin}}}(\boldsymbol{W}^{\text{mm}})\|_F = 0$ since for all $(i \asymp j) \in \mathcal{G}^{(k)}$, $(\boldsymbol{e}_i - \boldsymbol{e}_j)^\top \boldsymbol{W}^{\text{mm}} \boldsymbol{e}_k = 0$.

On the other hand, if $\|\boldsymbol{\Pi}_{\mathcal{S}_{\text{active}}^{\perp}}(\boldsymbol{W}^{\text{mm}})\|_F > 0$, then $\boldsymbol{W}^{\text{mm}} - \boldsymbol{\Pi}_{\mathcal{S}_{\text{active}}^{\perp}}(\boldsymbol{W}^{\text{mm}})$ also satisfies all of the constraints of (Graph-SVM), and

$$\|\boldsymbol{W}^{\text{mm}} - \boldsymbol{\Pi}_{\mathcal{S}_{\text{active}}^{\perp}}(\boldsymbol{W}^{\text{mm}})\|_F < \|\boldsymbol{W}^{\text{mm}}\|_F,$$

which is a contradiction. Therefore, $\boldsymbol{W}^{\text{mm}} \in \mathcal{S}_{\text{svm}}$.

• **(ii):** From (C.16), we know that

$$\nabla\mathcal{L}(\boldsymbol{W}) = \frac{1}{n}\sum_{i=1}^{n}\sum_{t\in\bar{\mathcal{O}}_i} s_{it}(\boldsymbol{x}_{it} - \boldsymbol{e}_{y_i})\bar{\boldsymbol{x}}_i^\top$$

where $\bar{\mathcal{O}}_i$ is given by (C.1). By definition of $\mathcal{S}_{\text{active}}$, $\|\boldsymbol{\Pi}_{\mathcal{S}_{\text{active}}^{\perp}}(\boldsymbol{x}_{it} - \boldsymbol{e}_{y_i})\bar{\boldsymbol{x}}_i^\top\|_F = 0$ for any $i \in [n]$ and $t \in \bar{\mathcal{O}}_i$. As $\nabla\mathcal{L}(\boldsymbol{W})$ is the summation of these terms, the advertised result is proved. ∎

Now, we are ready to prove a stronger version of Lemma 4.2.

Lemma C.6 (Stronger version of Lemma 4.2) *Suppose Assumptions 4.1 and 4.2 hold and consider the log-loss $\ell(u) = -\log(u)$, then $\mathcal{L}(\boldsymbol{W})$ is convex. Furthermore, $\mathcal{L}(\boldsymbol{W})$ is strictly convex on $\mathcal{S}_{active}$.*

Proof. Let $\mathcal{S}_K$ be the span of all $\boldsymbol{e}_i\boldsymbol{e}_j^\top$ where $i, j \in [K]$.
• **First Case: $\boldsymbol{W} \in \mathcal{S}_K$.** Let $g : \mathcal{S}_K \to \mathbb{R}^{K\times K}$ such that $g(\boldsymbol{W}) = \boldsymbol{E}\boldsymbol{W}\boldsymbol{E}^\top$. By definition, this function is linear. In addition to that, this function g is invertible by Assumption 4.1 and the domain of the function is $\mathcal{S}_K$. Note that Assumption 4.1 ensures $\operatorname{rank}(\boldsymbol{E}) = K$.

Let $\boldsymbol{E}' = \boldsymbol{C}' = \boldsymbol{I}_k$, $(\boldsymbol{X}_i', y_i')$ be a `DSET` such that $y_i' = y_i$ and $\boldsymbol{X}_i' = \boldsymbol{X}_i\boldsymbol{E}^\dagger$. Then, for any $\boldsymbol{W}' \in \mathbb{R}^{K\times K}$, we have the following:

$$\mathcal{L} \circ g^{-1}(\boldsymbol{W}') = \frac{1}{n}\sum_{i=1}^{n} \ell\left((\boldsymbol{c}_{y_i}')^\top (\boldsymbol{X}_i')^\top \mathbb{S}(\boldsymbol{X}_i'\boldsymbol{W}'\bar{\boldsymbol{x}}_i')\right)$$

Using Lemma C.7 and C.8, we know that $\mathcal{L}\circ g^{-1}(\boldsymbol{W}')$ is convex on $\mathbb{R}^{K\times K}$ and strictly convex on $g(\mathcal{S}_{\text{active}})$. Using these two facts and Lemma C.7, we have $\mathcal{L}(\boldsymbol{W})$ is convex on $\mathcal{S}_K$ and strictly convex on $\mathcal{S}_{\text{active}} \cap \mathcal{S}_K = \mathcal{S}_{\text{active}}$.
• **Second Case: $\boldsymbol{W} \notin \mathcal{S}_K$.** By definition of loss function in (ERM), we have

$$\mathcal{L}(\boldsymbol{W}) = \frac{1}{n}\sum_{i=1}^{n} \ell(\boldsymbol{c}_{y_i}^\top \boldsymbol{X}_i^\top \mathbb{S}(\boldsymbol{X}_i\boldsymbol{W}\bar{\boldsymbol{x}}_i)) = \frac{1}{n}\sum_{i=1}^{n} \ell(\boldsymbol{c}_{y_i}^\top \boldsymbol{X}_i^\top \mathbb{S}(\boldsymbol{X}_i\boldsymbol{\Pi}_{\mathcal{S}_K}(\boldsymbol{W})\bar{\boldsymbol{x}}_i)) = \mathcal{L}(\boldsymbol{\Pi}_{\mathcal{S}_K}(\boldsymbol{W})) \tag{C.21}$$

Let $\boldsymbol{W}_1, \boldsymbol{W}_2 \in \mathbb{R}^{d\times d}$ be arbitrary variables. For any $0 < \lambda < 1$, we have the following:

$$\mathcal{L}(\lambda\boldsymbol{W}_1 + (1-\lambda)\boldsymbol{W}_2) = \mathcal{L}(\lambda\boldsymbol{\Pi}_{\mathcal{S}_K}(\boldsymbol{W}_1) + \lambda\boldsymbol{\Pi}_{\mathcal{S}_K}(\boldsymbol{W}_2)) \tag{C.22}$$

Then, using (C.21) and (C.22), we have the following:

$$\begin{aligned}\lambda\mathcal{L}(\boldsymbol{W}_1) + (1-\lambda)\mathcal{L}(\boldsymbol{W}_2) &= \lambda\mathcal{L}(\boldsymbol{\Pi}_{\mathcal{S}_K}(\boldsymbol{W}_1)) + (1-\lambda)\mathcal{L}(\boldsymbol{\Pi}_{\mathcal{S}_K}(\boldsymbol{W}_2)) \\ &\overset{(a)}{\geq} \mathcal{L}(\lambda\boldsymbol{\Pi}_{\mathcal{S}_K}(\boldsymbol{W}_1) + \lambda\boldsymbol{\Pi}_{\mathcal{S}_K}(\boldsymbol{W}_2)) = \mathcal{L}(\lambda\boldsymbol{W}_1 + (1-\lambda)\boldsymbol{W}_2)\end{aligned}$$

where (a) follows from the convexity of $\mathcal{L}(\boldsymbol{W})$ inside $\mathcal{S}_K$. This implies that $\mathcal{L}(\boldsymbol{W})$ is convex when $\boldsymbol{W} \notin \mathcal{S}_K$. Note that $\mathcal{S}_{\text{active}} \subset \mathcal{S}_K$, therefore we do not look at the strict convexity in this case. ■

Lemma C.7 *Let $T : \mathcal{X} \to \mathcal{Y}$ be an invertible linear map. If a function $f : \mathcal{Y} \to \mathbb{R}$ is convex/strictly convex on $\mathcal{Y}$, then $f \circ T(x)$ is a convex/strictly convex function on $\mathcal{X}$.*

Proof. Let $x_1 \neq x_2 \in \mathcal{X}$ be arbitrary variables. Let $y_1 = T(x_1)$ and $y_2 = T(x_2)$. Since T is an invertible map, $y_1 \neq y_2$. Since T is a linear map, $T(\lambda x_1 + (1-\lambda)x_2) = \lambda y_1 + (1-\lambda)y_2$ for $0 < \lambda < 1$. Then, we obtain the following

$$\begin{aligned}\lambda(f \circ T(x_1)) + (1-\lambda)(f \circ T(x_2)) &= \lambda f(y_1) + (1-\lambda) f(y_2) \\ &\overset{(a)}{>} f(\lambda y_1 + (1-\lambda) y_2) \\ &= f \circ T(\lambda x_1 + (1-\lambda) x_2)\end{aligned}$$

where (a) follows from the strict convexity of the function f. This implies that $f \circ T(x)$ is a strictly convex function on $\mathcal{X}$. Note that if $y_1 = y_2$, then we cannot achieve (a). Additionally, if f is convex instead of strictly convex, then $>$ in (a) is changed to $\geq$, and $f \circ T(x)$ is convex. ■

Lemma C.8 *Suppose that Assumption 4.2 holds and $\boldsymbol{E} = \boldsymbol{I}_d$. Let $f : \mathbb{R}^{d \times d} \to \mathbb{R}^{d^2}$ be a linear transformation defined as $f(\boldsymbol{W}) = \boldsymbol{v}$ where $v_{i \times d + j} = \boldsymbol{e}_i^\top \boldsymbol{W} \boldsymbol{e}_j$. Then, $\mathcal{L} \circ f^{-1}(\boldsymbol{v})$ is convex. Furthermore, $\mathcal{L} \circ f^{-1}(\boldsymbol{v})$ is strictly convex on $f(\mathcal{S}_{active})$, where $\mathcal{S}_{active}$ is defined in Definition C.3.*

Proof. • **We first prove that $\mathcal{L} \circ f^{-1}(\boldsymbol{v})$ is convex.** Let $\bar{\ell} : \mathbb{R}^{d^2} \times \mathbb{R}^{T \times d} \times \mathbb{R} \to \mathbb{R}$ be defined as follows:

$$\bar{\ell}(\boldsymbol{v}, \boldsymbol{X}, y) := \ell\left(\boldsymbol{c}_y^\top \boldsymbol{X}^\top \mathbb{S}(\boldsymbol{X}\big(f^{-1}(\boldsymbol{v})\big)\bar{\boldsymbol{x}})\right).$$

Then, using (ERM), we have the following:

$$\mathcal{L} \circ f^{-1}(\boldsymbol{v}) = \frac{1}{n}\sum_{i=1}^{n} \ell\left(\boldsymbol{c}_{y_i}^\top \boldsymbol{X}_i^\top \mathbb{S}(\boldsymbol{X}_i\big(f^{-1}(\boldsymbol{v})\big)\bar{\boldsymbol{x}}_i)\right) = \frac{1}{n}\sum_{i=1}^{n} \bar{\ell}(\boldsymbol{v}, \boldsymbol{X}_i, y_i). \tag{C.23}$$

Note that the summation of convex functions is convex. Therefore, it is sufficient to prove the convexity of $\mathcal{L} \circ f^{-1}(\boldsymbol{v})$ by proving the convexity of $\bar{\ell}(\boldsymbol{v}, \boldsymbol{X}, y)$ for an arbitrary pair of input sequence and label $(\boldsymbol{X}, y)$. For the simplicity of notation, we use $\bar{\ell}(\boldsymbol{v})$ instead of $\bar{\ell}(\boldsymbol{v}, \boldsymbol{X}, y)$. Let m_j be the number of token ID j inside input sequence $\boldsymbol{X}$ for $j \in [K]$. Let k be the last token of $\boldsymbol{X}$. By Assumption 4.1 and log-loss, we know that

$$\bar{\ell}(\boldsymbol{v}) := \bar{\ell}(\boldsymbol{v}, \boldsymbol{X}, y) = -\log\left(\frac{m_y \cdot e^{\boldsymbol{v}_{y \times d + k}}}{\sum_{j \in [K]} m_j \cdot e^{\boldsymbol{v}_{j \times d + k}}}\right) = \log\left(\sum_{j \in [K]} m_j \cdot e^{\boldsymbol{v}_{j \times d + k}}\right) - \log(m_y \cdot e^{v_{y \times d + k}}).$$

Let $\boldsymbol{z} \in \mathbb{R}^{d^2}$ be a vector such that the $(j \times d + k)^{\text{th}}$ element of $\boldsymbol{z}$ is $z_{j \times d + k} = m_j \cdot e^{v_{j \times d + k}}$ for

$k \in [K]$, otherwise $z_i = 0$. Then, the Hessian matrix of $\bar{\ell}(\boldsymbol{v})$ is

$$\nabla^2 \bar{\ell}(\boldsymbol{v}) = \frac{1}{(\mathbf{1}^\top \boldsymbol{z})^2} \left((\mathbf{1}^\top \boldsymbol{z}) \mathrm{diag}(\boldsymbol{z}) - \boldsymbol{z}\boldsymbol{z}^\top \right)$$

For any $\boldsymbol{u} \in \mathbb{R}^{d^2}$, we obtain that

$$\boldsymbol{u}^\top \nabla^2 \bar{\ell}(\boldsymbol{v}) \boldsymbol{u} = \frac{1}{(\mathbf{1}^\top \boldsymbol{z})^2} \left(\left(\sum_{j=1}^{d^2} z_j \right) \left(\sum_{j=1}^{d^2} u_j^2 z_j \right) - \left(\sum_{j=1}^{d^2} u_j z_j \right)^2 \right) \geq 0. \tag{C.24}$$

Since $\boldsymbol{z}_i \geq 0$, $i \in [d^2]$, (C.24) follows from the Cauchy-Schwarz inequality $(\boldsymbol{\alpha}^\top \boldsymbol{\alpha})(\boldsymbol{\beta}^\top \boldsymbol{\beta}) \geq (\boldsymbol{\alpha}^\top \boldsymbol{\beta})^2$ applied to the vectors with $\alpha_i = u_i \sqrt{z_i}$ and $\beta_i = \sqrt{z_i}$. The equality condition holds $k\boldsymbol{\alpha} = \boldsymbol{\beta}$ for $k \neq 0$. This means that $\bar{\ell}(\boldsymbol{v})$ is convex.

• **Next, we will show that $\mathcal{L} \circ f^{-1}(\boldsymbol{v})$ is strictly convex on $f(\mathcal{S}_{\mathbf{active}})$.** Assume that $\mathcal{L} \circ f^{-1}(\boldsymbol{v})$ is not strictly convex on $f(\mathcal{S}_{\mathrm{active}})$. Using the convexity of $\mathcal{L} \circ f^{-1}(\boldsymbol{v})$, this implies that there exist $\boldsymbol{u}, \boldsymbol{v} \in f(\mathcal{S}_{\mathrm{active}})$, $\|\boldsymbol{u}\|_2 > 0$ such that

$$\boldsymbol{u}^\top \left(\nabla^2 \mathcal{L} \circ f^{-1}(\boldsymbol{v}) \right) \boldsymbol{u} = 0$$

Using the convexity of $\bar{\ell}(\boldsymbol{v})$ and (C.23), we have the following:

$$\boldsymbol{u}^\top \left(\nabla^2 \bar{\ell}(\boldsymbol{v}, \boldsymbol{X}_i, \boldsymbol{y}_i) \right) \boldsymbol{u} = 0 \qquad \forall i \in [n] \tag{C.25}$$

Now, we are going to prove that $\|\boldsymbol{u}\|_2 = 0$ if (C.25) holds. As $\boldsymbol{u} \in f(\mathcal{S}_{\mathrm{active}})$, there exists $\boldsymbol{W} \in \mathcal{S}_{\mathrm{active}}$ such that $f(\boldsymbol{W}) = \boldsymbol{u}$. As the function f preserves the norm, $\|\boldsymbol{W}\|_F > 0$. By definition of $\mathcal{S}_{\mathrm{active}}$, there exist $\bar{i}, \bar{j}, \bar{k} \in [K]$ and $(\boldsymbol{X}_{\bar{n}}, y_{\bar{n}}) \in \texttt{DSET}$ such that $\langle (\boldsymbol{e}_{\bar{i}} - \boldsymbol{e}_{\bar{j}}) \boldsymbol{e}_{\bar{k}}^\top, \boldsymbol{W} \rangle > 0$, $\boldsymbol{X}_{\bar{n}}$ includes the $\bar{j}^{\mathrm{th}}$ token, the last token of $\boldsymbol{X}_{\bar{n}}$ is the $\bar{k}^{\mathrm{th}}$ token, and $y_{\bar{n}} = \bar{i}$. On the other hand, by Assumption 4.2, $z_{\bar{i} \times d + \bar{k}}$ and $z_{\bar{j} \times d + \bar{k}}$ in (C.24) are non-zero for this input sequence $\boldsymbol{X}_{\bar{n}}$. Using the equality condition of Cauchy-Schwartz Inequality in (C.24), we obtain that $u_{\bar{i} \times d + \bar{k}} - u_{\bar{j} \times d + \bar{k}} = 0$. This implies that

$$\begin{aligned} 0 &= u_{\bar{i} \times d + \bar{k}} - u_{\bar{j} \times d + \bar{k}} \\ &= \boldsymbol{e}_{\bar{i}}^\top \boldsymbol{W} \boldsymbol{e}_{\bar{k}} - \boldsymbol{e}_{\bar{j}}^\top \boldsymbol{W} \boldsymbol{e}_{\bar{k}} \\ &= (\boldsymbol{e}_{\bar{i}} - \boldsymbol{e}_{\bar{j}})^\top \boldsymbol{W} \boldsymbol{e}_{\bar{k}} = \langle (\boldsymbol{e}_{\bar{i}} - \boldsymbol{e}_{\bar{j}}) \boldsymbol{e}_{\bar{k}}^\top, \boldsymbol{W} \rangle \end{aligned}$$

which contradicts with the fact that $\|\boldsymbol{u}\|_2 > 0$. This completes the proof. ∎

C.2.3 Divergence of $\|\mathbf{\Pi}_{\mathcal{S}_{\mathbf{svm}}}(\mathbf{W}(\tau))\|_F$

We first introduce the following lemmas establishing the descent property of gradient descent for $\mathcal{L}(\boldsymbol{W})$ (Lemma C.9) and the correlation between $\nabla\mathcal{L}(\boldsymbol{W})$ and the solution of (Graph-SVM) $\boldsymbol{W}^{\mathrm{mm}}$ (Lemma C.10) under the setting of Theorem 4.2. The proofs in this section follow Appendix B.1 of [185].

Lemma C.9 (Descent Lemma) *Consider the loss in* (C.15) *and choose step size* $\eta \le 1/L$ *where* L *is the Lipschitzness of* $\nabla\mathcal{L}(\boldsymbol{W})$ *defined in* (C.20)*. Then from any initialization* $\boldsymbol{W}(0)$*, Algorithm Algo-GD satisfies:*

$$\mathcal{L}(\boldsymbol{W}(\tau+1)) - \mathcal{L}(\boldsymbol{W}(\tau)) \le -\frac{\eta}{2}\|\nabla\mathcal{L}(\boldsymbol{W}(\tau))\|_F^2$$

for all $\tau \ge 0$*. Additionally, it holds that* $\sum_{\tau=0}^{\infty}\|\nabla\mathcal{L}(\boldsymbol{W}(\tau))\|_F^2 < \infty$*, and* $\lim_{\tau\to\infty}\|\nabla\mathcal{L}(\boldsymbol{W}(\tau))\|_F^2 = 0$

Proof. From (Algo-GD), for $\tau \ge 0$, we have that $\boldsymbol{W}(\tau+1) = \boldsymbol{W}(\tau) - \eta\nabla\mathcal{L}(\boldsymbol{W}(\tau))$. Since $\mathcal{L}(\boldsymbol{W})$ is L-smooth with L defined in (C.20), we get

$$\begin{aligned}
\mathcal{L}(\boldsymbol{W}(\tau+1)) &\le \mathcal{L}(\boldsymbol{W}(\tau)) + \langle\nabla\mathcal{L}(\boldsymbol{W}(\tau)), \boldsymbol{W}(\tau+1) - \boldsymbol{W}(\tau)\rangle + \frac{L}{2}\|\boldsymbol{W}(\tau+1) - \boldsymbol{W}(\tau)\|_F^2 \\
&= \mathcal{L}(\boldsymbol{W}(\tau)) - \eta\cdot\|\nabla\mathcal{L}(\boldsymbol{W}(\tau))\|_F^2 + \frac{L\eta^2}{2}\|\nabla\mathcal{L}(\boldsymbol{W}(\tau))\|_F^2 \\
&= \mathcal{L}(\boldsymbol{W}(\tau)) - \eta\left(1 - \frac{L\eta}{2}\right)\|\nabla\mathcal{L}(\boldsymbol{W}(\tau))\|_F^2 \\
&\le \mathcal{L}(\boldsymbol{W}(\tau)) - \frac{\eta}{2}\|\nabla\mathcal{L}(\boldsymbol{W}(\tau))\|_F^2.
\end{aligned}$$

The inequality above also indicates that

$$\sum_{\tau=0}^{\infty}\|\nabla\mathcal{L}(\boldsymbol{W}(\tau))\|_F^2 \le \frac{2}{\eta}(\mathcal{L}(\boldsymbol{W}(0)) - \mathcal{L}^*) < \infty, \quad \text{and} \quad \lim_{\tau\to\infty}\|\nabla\mathcal{L}(\boldsymbol{W}(\tau))\|_F^2 = 0.$$

■

Lemma C.10 *Let* $\boldsymbol{W}^{mm}$ *be the SVM solution of* (Graph-SVM) *and suppose* $\boldsymbol{W}^{mm} \ne 0$*. For any* $\boldsymbol{W}$ *with* $\|\boldsymbol{W}\|_F < \infty$*, the training loss* $\mathcal{L}(\boldsymbol{W})$ *in* (C.15) *obeys* $\langle\nabla\mathcal{L}(\boldsymbol{W}), \boldsymbol{W}^{mm}\rangle < 0$*. Equivalently,* $\langle\mathbf{\Pi}_{\mathcal{S}_{svm}}(\nabla\mathcal{L}(\boldsymbol{W})), \boldsymbol{W}^{mm}\rangle < 0$*.*

Proof. Recap $\mathcal{O}_i, \bar{\mathcal{O}}_i, \mathcal{R}_i, \bar{\mathcal{R}}_i, i \in [n]$ in (C.1). From (C.16), for any $\boldsymbol{W} \in \mathbb{R}^{d\times d}$, we obtain the gradient

$$\nabla\mathcal{L}(\boldsymbol{W}) = \frac{1}{n}\sum_{i=1}^{n}\sum_{t\in\bar{\mathcal{O}}_i} s_{it}(\boldsymbol{x}_{it} - \boldsymbol{e}_{y_i})\bar{\boldsymbol{x}}_i^\top.$$

Then

$$\begin{aligned}\langle \nabla \mathcal{L}(\boldsymbol{W}), \boldsymbol{W}^{\mathrm{mm}} \rangle &= \frac{1}{n} \sum_{i \in [n]} \sum_{t \in \bar{\mathcal{O}}_i} \left\langle s_{it} (\boldsymbol{x}_{it} - \boldsymbol{e}_{y_i}) \bar{\boldsymbol{x}}_i^\top, \boldsymbol{W}^{\mathrm{mm}} \right\rangle \\ &= \frac{1}{n} \sum_{i \in [n]} \sum_{t \in \bar{\mathcal{O}}_i} s_{it} \cdot \operatorname{trace}\left((\boldsymbol{W}^{\mathrm{mm}})^\top (\boldsymbol{x}_{it} - \boldsymbol{e}_{y_i}) \bar{\boldsymbol{x}}_i^\top \right) \\ &= \frac{1}{n} \sum_{i \in [n]} \sum_{t \in \bar{\mathcal{O}}_i} s_{it} \cdot (\boldsymbol{x}_{it} - \boldsymbol{e}_{y_i})^\top \boldsymbol{W}^{\mathrm{mm}} \bar{\boldsymbol{x}}_i.\end{aligned}$$

From the (Graph-SVM) formulation, we have that $(\boldsymbol{x}_{it} - \boldsymbol{e}_{y_i})^\top \boldsymbol{W}^{\mathrm{mm}} \bar{\boldsymbol{x}}_i = 0$ for $t \in \mathcal{R}_i$ and $(\boldsymbol{x}_{it} - \boldsymbol{e}_{y_i})^\top \boldsymbol{W}^{\mathrm{mm}} \bar{\boldsymbol{x}}_i \leq -1$ for $t \in \bar{\mathcal{R}}_i$. Then $\boldsymbol{W}^{\mathrm{mm}} \neq 0$ ensures that there exists $i \in [n]$ such that $\bar{\mathcal{R}}_i \neq \emptyset$, which implies that

$$\langle \nabla \mathcal{L}(\boldsymbol{W}), \boldsymbol{W}^{\mathrm{mm}} \rangle < 0.$$

Using the fact that $\boldsymbol{W}^{\mathrm{mm}} \in \mathcal{S}_{\mathrm{svm}}$ (Lemma C.1) completes the proof. ∎

The next theorem proves the divergence of norm of the iterates $\boldsymbol{W}(\tau)$.

Theorem C.1 *Consider the same setting as in Theorem 4.2, then there is no finite $\boldsymbol{W} \in \mathbb{R}^{d \times d}$ satisfying $\nabla \mathcal{L}(\boldsymbol{W}) = 0$. Furthermore, Algorithm Algo-GD with the step size $\eta \leq 1/L$ where L is the Lipschitzness of $\nabla \mathcal{L}(\boldsymbol{W})$ defined in* (C.20) *and any starting point $\boldsymbol{W}(0)$ satisfies $\lim_{\tau \to \infty} \|\boldsymbol{\Pi}_{\mathcal{S}_{svm}}(\boldsymbol{W}(\tau))\|_F = \infty$.*

Proof. Following Lemma C.9, when using log-loss $\ell(u) = -\log(u)$, for any starting point $\boldsymbol{W}(0)$, the Algorithm Algo-GD satisfies $\lim_{\tau \to \infty} \|\nabla \mathcal{L}(\boldsymbol{W}(\tau))\|_F^2 = 0$. Moreover, assume that the first claim is wrong and that there is a finite critical point $\boldsymbol{W}$ that satisfies $\nabla \mathcal{L}(\boldsymbol{W}) = 0$. We then have $\langle \boldsymbol{\Pi}_{\mathcal{S}_{\mathrm{svm}}}(\nabla \mathcal{L}(\boldsymbol{W})), \boldsymbol{W}^{\mathrm{mm}} \rangle = 0$. This leads to a contradiction with Lemma C.10 which says that for any finite $\boldsymbol{W}$, $\langle \boldsymbol{\Pi}_{\mathcal{S}_{\mathrm{svm}}}(\nabla \mathcal{L}(\boldsymbol{W})), \boldsymbol{W}^{\mathrm{mm}} \rangle < 0$. This implies that $\|\boldsymbol{\Pi}_{\mathcal{S}_{\mathrm{svm}}}(\boldsymbol{W}(\tau))\|_F \to \infty$. ∎

C.2.4 Uniqueness and Finiteness of $\mathbf{W}^{\mathrm{fin}}$

Lemma C.11 *Consider the setting of Theorem 4.2. $\boldsymbol{W}^{fin}$ defined in Theorem 4.2 is unique and finite.*

Proof. Following Lemma C.3, it is equivalent to show $\mathcal{W}^{\mathrm{fin}}$ defined in Def. C.1 has unique element $\bar{\boldsymbol{W}}^{\mathrm{fin}}$ and $\bar{\boldsymbol{W}}^{\mathrm{fin}} = \boldsymbol{W}^{\mathrm{fin}}$ is unique and finite. To start with, recap the definition of $\overline{\texttt{DSET}}$ (Definition C.1). Denote $\mathcal{I} \subset [n]$ as in (C.3), and let $\boldsymbol{s}_i = \mathbb{S}(\bar{\boldsymbol{X}}_i \boldsymbol{W} \bar{\boldsymbol{x}}_i)$

where $(\bar{\boldsymbol{X}}_i, y_i) \in \overline{\texttt{DSET}}$. What's more, recap the ERM loss from (C.15) and loss function $\ell(u) = -\log(u)$. Then we have

$$\bar{\boldsymbol{W}}^{\text{fin}} = \arg\min_{\boldsymbol{W} \in \mathcal{S}_{\text{fin}}} \bar{\mathcal{L}}(\boldsymbol{W}) \quad \text{where} \quad \bar{\mathcal{L}}(\boldsymbol{W}) = \frac{1}{n} \sum_{i \in \mathcal{I}} -\log \left(\sum_{t \in \mathcal{O}_i} s_{it} \right). \tag{C.26}$$

Different from (C.1), $\mathcal{O}_i$ for dataset $\overline{\texttt{DSET}}$ is defined as follows:

$$\mathcal{O}_i := \left\{ t \mid x_{it} = y_i,\ \boldsymbol{x}_{it} \in \bar{\boldsymbol{X}}_i,\ t \in [\bar{T}_i] \right\},$$

where $\bar{T}_i$ is the number of tokens – tokens that are in the same SCC as label token y_i within their corresponding TPG – in $\bar{\boldsymbol{X}}_i$ and if recap the notation of $\mathcal{R}_i$ in (C.1), here we have $|\mathcal{R}_i| = \bar{T}_i$.

We will first prove that $\bar{\boldsymbol{W}}^{\text{fin}}$ is finite by contradiction. Specifically, we will show that for any $\boldsymbol{W} \in \mathcal{S}_{\text{fin}}$ with $\|\boldsymbol{W}\|_F \neq 0$, $\lim_{R \to \infty} \bar{\mathcal{L}}(R \cdot \boldsymbol{W}) = \infty$ which implies that the optimal solution $\bar{\boldsymbol{W}}^{\text{fin}}$ has to be finite.

Let $\boldsymbol{W} \in \mathcal{S}_{\text{fin}}$ be arbitrary attention weight. Following the definition of $\mathcal{S}_{\text{fin}}$ as in Def. 4.1, we have that $(\boldsymbol{e}_i - \boldsymbol{e}_j)\boldsymbol{e}_k^\top \in \mathcal{S}_{\text{fin}}$ for all $(i \asymp j) \in \mathcal{G}^{(k)}$ and $k \in [K]$. For any $\bar{\boldsymbol{X}}$, let $\mathcal{O}, \bar{\mathcal{O}}$ correspond to the token index sets. Then we have

$$\sum_{t \in \mathcal{O}} s_t = \frac{|\mathcal{O}| e^{\boldsymbol{e}_y^\top (R \cdot \boldsymbol{W}) \boldsymbol{e}_k}}{|\mathcal{O}| e^{\boldsymbol{e}_y^\top (R \cdot \boldsymbol{W}) \boldsymbol{e}_k} + \sum_{t \in \bar{\mathcal{O}}} e^{\boldsymbol{e}_t^\top (R \cdot \boldsymbol{W}) \boldsymbol{e}_k}} = \frac{1}{1 + \sum_{t \in \bar{\mathcal{O}}} e^{(\boldsymbol{e}_t - \boldsymbol{e}_y)^\top (R \cdot \boldsymbol{W}) \boldsymbol{e}_k} / |\mathcal{O}|}.$$

Given the sample loss $\ell = -\log\left(\sum_{t \in \mathcal{O}} s_t\right)$ and to prevent it from divergence as $R \to \infty$, that is, $\sum_{t \in \mathcal{O}} s_t \not\to 0$, we have

$$\sum_{t \in \bar{\mathcal{O}}} e^{(\boldsymbol{e}_t - \boldsymbol{e}_y)^\top (R \cdot \boldsymbol{W}) \boldsymbol{e}_k} \not\to \infty \quad \Longrightarrow \quad (\boldsymbol{e}_y - \boldsymbol{e}_t)^\top \boldsymbol{W} \boldsymbol{e}_k \geq 0 \text{ for all } t \in \bar{\mathcal{O}}$$

where $\boldsymbol{e}_y$ is the label token and $\boldsymbol{e}_t$ is any other token in $\bar{\mathcal{O}}$. Recap from the construction of TPG in Section 4.2.1, the directed edge $y \to t$ exists in the graph $\mathcal{G}^{(k)}$. Since SCC is bidirectionally reachable, which means there exists route from t to y, e.g., $t \to p_1 \to p_2 \to \cdots p_m \to y$, similarly we have

$$(\boldsymbol{e}_t - \boldsymbol{e}_{p_1})^\top \boldsymbol{W} \boldsymbol{e}_k,\ (\boldsymbol{e}_{p_1} - \boldsymbol{e}_{p_2})^\top \boldsymbol{W} \boldsymbol{e}_k,\ \cdots, (\boldsymbol{e}_{p_m} - \boldsymbol{e}_y)^\top \boldsymbol{W} \boldsymbol{e}_k \geq 0 \Longrightarrow (\boldsymbol{e}_t - \boldsymbol{e}_y)^\top \boldsymbol{W} \boldsymbol{e}_k \geq 0.$$

Combining results in that $(\boldsymbol{e}_t - \boldsymbol{e}_y)^\top \boldsymbol{W} \boldsymbol{e}_k = 0$. This implies that for all $(i \asymp j) \in \mathcal{G}^{(k)}$ and $R \to \infty$, to ensure the training loss $\bar{\mathcal{L}}(R \cdot \boldsymbol{W})$ finite, $(\boldsymbol{e}_i - \boldsymbol{e}_j)^\top \boldsymbol{W} \boldsymbol{e}_k = 0$, which contradicts

the facts that $\boldsymbol{W} \in \mathcal{S}_{\text{fin}}$ and $\boldsymbol{W} \neq 0$.

Next, we prove that there is at most one local minimum for $\bar{\mathcal{L}}(\boldsymbol{W})$ based on Lemma 4.2. Suppose to the contrary that we have two optimal solutions satisfying $\min_{\boldsymbol{W}} \bar{\mathcal{L}}(\boldsymbol{W}) = \bar{\mathcal{L}}(\boldsymbol{W}^{\text{fin}}{}_1) = \bar{\mathcal{L}}(\boldsymbol{W}^{\text{fin}}{}_2), \boldsymbol{W}^{\text{fin}}{}_1 \neq \boldsymbol{W}^{\text{fin}}{}_2$. From Lemma 4.2, since $\mathcal{L}(\boldsymbol{W})$ is strictly convex over subspace $\mathcal{S}_{\text{fin}}$, for any $\boldsymbol{W}_1, \boldsymbol{W}_2 \in \mathcal{S}_{\text{fin}}, \lambda \in (0,1)$, if $\boldsymbol{W}_1 \neq \boldsymbol{W}_2$, we have

$$\bar{\mathcal{L}}((1-\lambda)\boldsymbol{W}_1 + \lambda \boldsymbol{W}_2) < (1-\lambda)\bar{\mathcal{L}}(\boldsymbol{W}_1) + \lambda \bar{\mathcal{L}}(\boldsymbol{W}_2) \tag{C.27}$$

Substitute $\boldsymbol{W}_1 = \boldsymbol{W}^{\text{fin}}{}_1, \boldsymbol{W}_2 = \boldsymbol{W}^{\text{fin}}{}_2$, we get

$$\bar{\mathcal{L}}((1-\lambda)\boldsymbol{W}^{\text{fin}}{}_1 + \lambda \boldsymbol{W}^{\text{fin}}{}_2) < (1-\lambda)\bar{\mathcal{L}}(\boldsymbol{W}^{\text{fin}}{}_1) + \lambda \bar{\mathcal{L}}(\boldsymbol{W}^{\text{fin}}{}_2) = \min_{\boldsymbol{W}} \bar{\mathcal{L}}(\boldsymbol{W}) \tag{C.28}$$

which leads to a contradiction to the assumption that $\boldsymbol{W}^{\text{fin}}{}_1$ and $\boldsymbol{W}^{\text{fin}}{}_2$ are both optimal solutions. Combining this with the fact that $\boldsymbol{W}^{\text{fin}}$ is not attained at infinity, there exists one unique and finite solution with $\bar{\boldsymbol{W}}^{\text{fin}} = \arg\min_{\boldsymbol{W} \in \mathcal{S}_{\text{fin}}} \bar{\mathcal{L}}(\boldsymbol{W})$. ■

C.2.5 Proof of Theorem 4.2

Lemma C.12 *Consider the same setting of Theorem 4.2. Given any $\pi > 0$, there exists $R_\pi > 0$ such that for any $\boldsymbol{W}$ with $\|\boldsymbol{\Pi}_{\mathcal{S}_{fin}}(\boldsymbol{W})\|_F < \infty$ and $\|\boldsymbol{\Pi}_{\mathcal{S}_{fin}^{\perp}}(\boldsymbol{W})\|_F > R_\pi$,*

$$\mathcal{L}(\boldsymbol{W}) \geq \mathcal{L}\left((1+\pi)\frac{\|\boldsymbol{\Pi}_{\mathcal{S}_{fin}^{\perp}}(\boldsymbol{W})\|_F}{\|\boldsymbol{W}^{mm}\|_F}\boldsymbol{W}^{mm} + \boldsymbol{\Pi}_{\mathcal{S}_{fin}}(\boldsymbol{W})\right).$$

Proof. Recap $\mathcal{O}_i, \bar{\mathcal{O}}_i, \mathcal{R}_i, \bar{\mathcal{R}}_i$, $i \in [n]$ from (C.1) and recap from (C.15), we get that for any $\boldsymbol{W}$,

$$\mathcal{L}(\boldsymbol{W}) = -\frac{1}{n}\sum_{i=1}^{n} \log\left(\sum_{t \in \mathcal{O}_i} s_{it}\right)$$

where $\boldsymbol{s}_i = \mathbb{S}(\boldsymbol{X}_i \boldsymbol{W} \bar{\boldsymbol{x}}_i)$.

To obtain the result, we establish a refined softmax probability control by studying the distance to $\bar{\mathcal{L}}(\boldsymbol{W})$ as defined in Definition C.2. Let $\boldsymbol{W}^{\|} = \boldsymbol{\Pi}_{\mathcal{S}_{\text{fin}}}(\boldsymbol{W})$, $\boldsymbol{W}^{\perp} = \boldsymbol{W} - \boldsymbol{W}^{\|}$, $\|\boldsymbol{W}^{\perp}\|_F = R$, and $\Theta = 1/\|\boldsymbol{W}^{\text{mm}}\|_F$. Let $\boldsymbol{b}_i = \boldsymbol{X}_i \boldsymbol{W}^{\|} \bar{\boldsymbol{x}}_i$, $\boldsymbol{a}_i^\star = \boldsymbol{X}_i((1+\pi)R\Theta \cdot \boldsymbol{W}^{\text{mm}})\bar{\boldsymbol{x}}_i$, and $\boldsymbol{s}_i^\star = \mathbb{S}(\boldsymbol{X}_i((1+\pi)R\Theta \cdot \boldsymbol{W}^{\text{mm}} + \boldsymbol{W}^{\|})\bar{\boldsymbol{x}}_i) = \mathbb{S}(\boldsymbol{a}_i^\star + \boldsymbol{b}_i)$. Additionally, let $\boldsymbol{a}_i = \boldsymbol{X}_i \boldsymbol{W}^{\perp} \bar{\boldsymbol{x}}_i$, $\boldsymbol{s}_i = \mathbb{S}(\boldsymbol{X}_i \boldsymbol{W} \bar{\boldsymbol{x}}_i) = \mathbb{S}(\boldsymbol{a}_i + \boldsymbol{b}_i)$, $\gamma_i^\star := \boldsymbol{c}_{y_i}^\top \boldsymbol{X}_i^\top \boldsymbol{s}_i^\star$, and $\gamma_i := \boldsymbol{c}_{y_i}^\top \boldsymbol{X}_i^\top \boldsymbol{s}_i$.

From proof of Lemma C.2 (more specifically (C.10)), we get for all $t, t' \in \mathcal{R}_i$

$$(\boldsymbol{x}_{it} - \boldsymbol{x}_{it'})^\top \boldsymbol{V} \bar{\boldsymbol{x}}_i = 0 \quad \text{for any } \boldsymbol{V} \perp \mathcal{S}_{\text{fin}} \Longrightarrow a_{it}^\star - a_{it'}^\star = a_{it} - a_{it'} = 0.$$

Additionally, since $\frac{\boldsymbol{W}}{\|\boldsymbol{W}\|_F} \neq \Theta \boldsymbol{W}^{\mathrm{mm}}$, there exist $i \in [n], t \in \mathcal{O}_i, t' \in \bar{\mathcal{R}}_i$ such that $(\boldsymbol{x}_{it} - \boldsymbol{x}_{it'})^\top \boldsymbol{W} \bar{\boldsymbol{x}}_i < R\Theta$. Then,

$$\sum_{t\in\mathcal{O}_i} s_{it} = \frac{\sum_{t\in\mathcal{O}_i} e^{a_{it}+b_{it}}}{\sum_{t\in[T_i]} e^{a_{it}+b_{it}}} \leq \frac{\sum_{t\in\mathcal{O}_i} e^{b_{it}}}{\sum_{t\in\mathcal{R}_i} e^{b_{it}} + e^{-R\Theta+b_{it'}}} \leq \frac{c_i}{d_i + e^{-R\Theta-\bar{b}}}, \quad \exists i \in [n]$$
$$\sum_{t\in\mathcal{O}_i} s^\star_{it} = \frac{\sum_{t\in\mathcal{O}_i} e^{a^\star_{it}+b_{it}}}{\sum_{t\in[T_i]} e^{a^\star_{it}+b_{it}}} \geq \frac{\sum_{t\in\mathcal{O}_i} e^{b_{it}}}{\sum_{t\in\mathcal{R}_i} e^{b_{it}} + \sum_{t\in\bar{\mathcal{R}}_i} e^{-(1+\pi)R\Theta+b_{it}}} \geq \frac{c_i}{d_i + Te^{-(1+\pi)R\Theta+\bar{b}}}, \quad \forall i \in [n],$$

where $c_i = \sum_{t\in\mathcal{O}_i} e^{b_{it}}$, $d_i = \sum_{t\in\mathcal{R}_i} e^{b_{it}}$, and $\bar{b} := \max_{t\in\bar{\mathcal{R}}_i, i\in[n]} |b_{it}|$, and we have

$$\bar{\mathcal{L}}(\boldsymbol{W}) = \bar{\mathcal{L}}(\boldsymbol{W}^{\|}) = -\frac{1}{n}\sum_{i\in\mathcal{I}} \log\left(\frac{c_i}{d_i}\right).$$

Then we obtain

$$\begin{aligned}\mathcal{L}(\boldsymbol{W}) - \bar{\mathcal{L}}(\boldsymbol{W}) &\geq -\frac{1}{n}\left(\log\left(\frac{c_i}{d_i + e^{-R\Theta-\bar{b}}}\right) - \log\left(\frac{c_i}{d_i}\right)\right) \\ &\geq \frac{1}{n}\log\left(1 + e^{-R\Theta-\bar{b}}/d_i\right)\end{aligned}$$

and let $j := \arg\max_{i\in[n]} \left(-\log\left(\sum_{t\in\mathcal{O}_i} s^\star_{it}\right) + \log\left(\frac{c_i}{d_i}\right)\right)$. We can upper-bound the loss difference for $(1+\pi)R\Theta \cdot \boldsymbol{W}^{\mathrm{mm}} + \boldsymbol{W}^{\|}$ as follows:

$$\begin{aligned}\mathcal{L}((1+\pi)R\Theta \cdot \boldsymbol{W}^{\mathrm{mm}} + \boldsymbol{W}^{\|}) - \bar{\mathcal{L}}(\boldsymbol{W}) &\leq \max_{i\in[n]} \left(-\log\left(\sum_{t\in\mathcal{O}_i} s^\star_{it}\right) + \log\left(\frac{c_i}{d_i}\right)\right) \\ &= \log\left(1 + Te^{-(1+\pi)R\Theta+\bar{b}}/d_j\right).\end{aligned}$$

Combining them together results in that, $\mathcal{L}(\boldsymbol{W}) > \mathcal{L}((1+\pi)R\Theta \cdot \boldsymbol{W}^{\mathrm{mm}} + \boldsymbol{W}^{\|})$ whenever

$$\frac{1}{n}\log\left(1 + e^{-R\Theta-\bar{b}}/d_i\right) \geq \log\left(1 + Te^{-(1+\pi)R\Theta+\bar{b}}/d_j\right).$$

Given that $x/2 \leq \log(1+x)$ for any $0 \leq x \leq 1$, we get $\mathcal{L}(\boldsymbol{W}) > \mathcal{L}((1+\pi)R\Theta \cdot \boldsymbol{W}^{\mathrm{mm}} + \boldsymbol{W}^{\|})$ whenever

$$\begin{aligned}&\frac{e^{-R\Theta-\bar{b}}}{2nd_i} \geq \frac{Te^{-(1+\pi)R\Theta+\bar{b}}}{d_j} \quad \text{when} \quad R \geq -\frac{\log(d_j)+\bar{b}}{\Theta} \\ \Longrightarrow &R > R_\pi := \max\left\{\frac{1}{\pi\Theta}\log\left(\frac{2nTd_i}{d_j}\right) + \frac{2\bar{b}}{\pi\Theta}, -\frac{\log(d_j)+\bar{b}}{\Theta}\right\}. \qquad \text{(C.29)}\end{aligned}$$

Here, since $\|\boldsymbol{W}^{\parallel}\|_F < \infty$, $d_i, d_j, \bar{b} < \infty$. ∎

Proof of Theorem 4.2. Now gathering all the results so far, we are ready to prove the gradient descent convergence. The divergence of $\|\boldsymbol{W}(\tau)\|_F$ as $\tau \to \infty$ has been proven by Theorem C.1.

• **We first show that $\boldsymbol{\Pi}_{\mathcal{S}_{\text{fin}}}(\boldsymbol{W}(\tau)) \to \boldsymbol{W}^{\text{fin}}$.** Lemma C.3 has established the equivalence between $\bar{\boldsymbol{W}}^{\text{fin}}$ and $\boldsymbol{W}^{\text{fin}}$. Since $\mathcal{L}(\boldsymbol{W})$ is convex following Lemma 4.2, we have that $\mathcal{L}(\boldsymbol{W}(\tau)) \to \mathcal{L}_\star := \min_{\boldsymbol{W}} \mathcal{L}(\boldsymbol{W})$. Additionally, Lemma 4.2 shows that $\mathcal{L}(\boldsymbol{W})$ is strictly convex on $\mathcal{S}_{\text{fin}}$ and Lemma C.11 shows that $\boldsymbol{W}^{\text{fin}}$ is the unique finite solution. Suppose $\boldsymbol{\Pi}_{\mathcal{S}_{\text{fin}}}(\boldsymbol{W}(\tau)) \not\to \boldsymbol{W}^{\text{fin}}$. Let $\boldsymbol{W}$ be any matrix with $\boldsymbol{\Pi}_{\mathcal{S}_{\text{fin}}}(\boldsymbol{W}) \neq \boldsymbol{W}^{\text{fin}}$. Then Lemma C.4 and Lemma C.2 give that $\mathcal{L}(\boldsymbol{W}) \geq \bar{\mathcal{L}}(\boldsymbol{W}) = \bar{\mathcal{L}}(\boldsymbol{\Pi}_{\mathcal{S}_{\text{fin}}}(\boldsymbol{W})) > \mathcal{L}_\star$ where $\mathcal{L}_\star = \min_{\boldsymbol{W}} \mathcal{L}(\boldsymbol{W}) = \min_{\boldsymbol{W}} \bar{\mathcal{L}}(\boldsymbol{W})$. Given that $\mathcal{L}_\star$ is achievable, the proof is done by contradiction and we have that $\boldsymbol{\Pi}_{\mathcal{S}_{\text{fin}}}(\boldsymbol{W}(\tau)) \to \boldsymbol{W}^{\text{fin}}$.

• **We next prove that when $\boldsymbol{W}^{\text{mm}} = 0$, $\boldsymbol{\Pi}_{\mathcal{S}_{\text{fin}}^{\perp}}(\boldsymbol{W}(\tau)) = \boldsymbol{\Pi}_{\mathcal{S}_{\text{fin}}^{\perp}}(\boldsymbol{W}(0))$.** Recap the gradient in (C.16) where

$$\nabla \mathcal{L}(\boldsymbol{W}) = \frac{1}{n} \sum_{i=1}^{n} \sum_{t \in \bar{\mathcal{O}}_i} s_{it} (\boldsymbol{x}_{it} - \boldsymbol{e}_{y_i}) \bar{\boldsymbol{x}}_i^\top.$$

Since $\boldsymbol{W}^{\text{mm}} = 0$ implies that for all $i \in [n]$, $\bar{\mathcal{R}}_i = \emptyset$, then $\bar{\mathcal{O}}_i \subseteq \mathcal{R}_i$. Additionally, since following definition of $\mathcal{S}_{\text{fin}}$ from Def. 4.1, for any $t \in \mathcal{R}_i$, $(\boldsymbol{x}_{it} - \boldsymbol{e}_{y_i}) \bar{\boldsymbol{x}}_i^\top \in \mathcal{S}_{\text{fin}}$. Then we obtain

$$\boldsymbol{\Pi}_{\mathcal{S}_{\text{fin}}}(\nabla \mathcal{L}(\boldsymbol{W})) = \frac{1}{n} \sum_{i=1}^{n} \sum_{t \in \bar{\mathcal{O}}_i} s_{it} \boldsymbol{\Pi}_{\mathcal{S}_{\text{fin}}} \left((\boldsymbol{x}_{it} - \boldsymbol{e}_{y_i}) \bar{\boldsymbol{x}}_i^\top \right) = \frac{1}{n} \sum_{i=1}^{n} \sum_{t \in \bar{\mathcal{O}}_i} s_{it} (\boldsymbol{x}_{it} - \boldsymbol{e}_{y_i}) \bar{\boldsymbol{x}}_i^\top = \nabla \mathcal{L}(\boldsymbol{W}).$$

Therefore, $\boldsymbol{\Pi}_{\mathcal{S}_{\text{fin}}^{\perp}}(\nabla \mathcal{L}(\boldsymbol{W})) = 0$ for any $\boldsymbol{W}$, which completes the proof.

• **Last, we show that when $\boldsymbol{W}^{\text{mm}} \neq 0$, $\boldsymbol{W}(\tau)/\|\boldsymbol{W}(\tau)\|_F \to \boldsymbol{W}^{\text{mm}}/\|\boldsymbol{W}^{\text{mm}}\|_F$.**

Consider any $\boldsymbol{W} \in \mathbb{R}^{d \times d}$, and let $\boldsymbol{W}^{\perp} = \boldsymbol{\Pi}_{\mathcal{S}_{\text{fin}}^{\perp}}(\boldsymbol{W})$, $\boldsymbol{W}^{\parallel} = \boldsymbol{\Pi}_{\mathcal{S}_{\text{fin}}}(\boldsymbol{W})$, $R = \|\boldsymbol{W}^{\perp}\|_F$, and $\Theta = 1/\|\boldsymbol{W}^{\text{mm}}\|_F$. Next, from Lemma C.9, for any $\tau \geq 0$, $\mathcal{L}(\boldsymbol{W}(\tau+1)) \leq \mathcal{L}(\boldsymbol{W}(\tau))$. Let $\boldsymbol{W}^{\perp}(\tau) = \boldsymbol{\Pi}_{\mathcal{S}_{\text{fin}}^{\perp}}(\boldsymbol{W}(\tau))$ and $\boldsymbol{W}^{\parallel}(\tau) = \boldsymbol{\Pi}_{\mathcal{S}_{\text{fin}}}(\boldsymbol{W}(\tau))$. Following Lemma C.4, since loss satisfies $\ell(1) = -\log(1) = 0$, we obtain

$$\mathcal{L}(\boldsymbol{W}(\tau)) \geq \bar{\mathcal{L}}(\boldsymbol{W}^{\parallel}(\tau)).$$

Since following Lemma C.11, the training risk $\bar{\mathcal{L}}(\boldsymbol{W}^{\parallel}(\tau))$ is infinite if $\|\boldsymbol{W}^{\parallel}(\tau)\|_F \to \infty$, which implies $\|\boldsymbol{W}^{\parallel}(\tau)\|_F < \infty$ for any $\tau \geq 0$. Additionally, Theorem C.1 proves the divergence of $\boldsymbol{W}(\tau)$ as $\tau \to \infty$, hence we have $\|\boldsymbol{W}^{\perp}(\tau)\|_F \to \infty$.

Applying Lemma C.12, as well as the fact that $\|\boldsymbol{W}^{\parallel}(\tau)\|_F < \infty$ and $\|\boldsymbol{W}^{\perp}(\tau)\|_F \to \infty$, there exists sufficiently large R_π as defined in (C.29) such that once $\|\boldsymbol{W}^{\perp}(\tau)\|_F = R > R_\pi$,

$\mathcal{L}(\boldsymbol{W}(\tau)) - \mathcal{L}((1+\pi)R\Theta \cdot \boldsymbol{W}^{\mathrm{mm}} + \boldsymbol{W}^{\parallel}(\tau)) \geq 0$. Since $\mathcal{L}(\boldsymbol{W})$ is convex following Lemma 4.2, we have that

$$\begin{aligned}
\mathcal{L}(\boldsymbol{W}) &\leq \mathcal{L}((1+\pi)R\Theta \cdot \boldsymbol{W}^{\mathrm{mm}} + \boldsymbol{W}^{\parallel}) + \left\langle \nabla \mathcal{L}(\boldsymbol{W}), \boldsymbol{W} - \left((1+\pi)R\Theta \cdot \boldsymbol{W}^{\mathrm{mm}} + \boldsymbol{W}^{\parallel}\right) \right\rangle \\
&= \mathcal{L}((1+\pi)R\Theta \cdot \boldsymbol{W}^{\mathrm{mm}} + \boldsymbol{W}^{\parallel}) + \left\langle \nabla \mathcal{L}(\boldsymbol{W}), \left(\boldsymbol{W}^{\perp} - (1+\pi)R\Theta \cdot \boldsymbol{W}^{\mathrm{mm}}\right) \right\rangle \\
&= \mathcal{L}((1+\pi)R\Theta \cdot \boldsymbol{W}^{\mathrm{mm}} + \boldsymbol{W}^{\parallel}) + \left\langle \boldsymbol{\Pi}_{\mathcal{S}_{\mathrm{fin}}^{\perp}}(\nabla \mathcal{L}(\boldsymbol{W})), \left(\boldsymbol{W}^{\perp} - (1+\pi)R\Theta \cdot \boldsymbol{W}^{\mathrm{mm}}\right) \right\rangle .
\end{aligned} \tag{C.30}$$

Here, the first inequality uses the convexity of $\mathcal{L}(\boldsymbol{W})$ and last equation is obtained from the fact that $\boldsymbol{W}^{\mathrm{mm}} \perp \mathcal{S}_{\mathrm{fin}}$. It implies that once $\|\boldsymbol{W}^{\perp}(\tau)\|_F = R > R_{\pi}$,

$$\left\langle \boldsymbol{\Pi}_{\mathcal{S}_{\mathrm{fin}}^{\perp}}(\nabla \mathcal{L}(\boldsymbol{W})), \left(\boldsymbol{W}^{\perp} - (1+\pi)R\Theta \cdot \boldsymbol{W}^{\mathrm{mm}}\right) \right\rangle \geq 0.$$

Now we choose τ_0 such that for all $\tau \geq \tau_0$, $\|\boldsymbol{W}^{\perp}(\tau)\|_F > R_{\pi}$. Then for $\tau > \tau_0$, we get

$$\left\langle \boldsymbol{W}^{\perp}(\tau+1) - \boldsymbol{W}^{\perp}(\tau), \frac{\boldsymbol{W}^{\mathrm{mm}}}{\|\boldsymbol{W}^{\mathrm{mm}}\|_F} \right\rangle \geq \frac{1}{1+\pi} \left\langle \boldsymbol{W}^{\perp}(\tau+1) - \boldsymbol{W}^{\perp}(\tau), \frac{\boldsymbol{W}^{\perp}(\tau)}{\|\boldsymbol{W}^{\perp}(\tau)\|_F} \right\rangle \tag{C.31}$$

where

$$\begin{aligned}
&\left\langle \boldsymbol{W}^{\perp}(\tau+1) - \boldsymbol{W}^{\perp}(\tau), \frac{\boldsymbol{W}^{\perp}(\tau)}{\|\boldsymbol{W}^{\perp}(\tau)\|_F} \right\rangle \\
&= \frac{1}{2\|\boldsymbol{W}^{\perp}(\tau)\|_F} \left(\|\boldsymbol{W}^{\perp}(\tau+1)\|_F^2 - \|\boldsymbol{W}^{\perp}(\tau)\|_F^2 - \|\boldsymbol{W}^{\perp}(\tau+1) - \boldsymbol{W}^{\perp}(\tau)\|_F^2 \right) \\
&\geq \frac{\|\boldsymbol{W}^{\perp}(\tau+1)\|_F^2 - \|\boldsymbol{W}^{\perp}(\tau)\|_F^2}{2\|\boldsymbol{W}^{\perp}(\tau)\|_F} - \|\boldsymbol{W}^{\perp}(\tau+1) - \boldsymbol{W}^{\perp}(\tau)\|_F^2 && \text{(C.32)} \\
&\geq \|\boldsymbol{W}^{\perp}(\tau+1)\|_F - \|\boldsymbol{W}^{\perp}(\tau)\|_F - \|\boldsymbol{W}^{\perp}(\tau+1) - \boldsymbol{W}^{\perp}(\tau)\|_F^2 && \text{(C.33)} \\
&\geq \|\boldsymbol{W}^{\perp}(\tau+1)\|_F - \|\boldsymbol{W}^{\perp}(\tau)\|_F - \|\boldsymbol{W}(\tau+1) - \boldsymbol{W}(\tau)\|_F^2 && \text{(C.34)} \\
&\geq \|\boldsymbol{W}^{\perp}(\tau+1)\|_F - \|\boldsymbol{W}^{\perp}(\tau)\|_F + 2\eta \left(\mathcal{L}(\boldsymbol{W}(\tau+1)) - \mathcal{L}(\boldsymbol{W}(\tau)) \right). && \text{(C.35)}
\end{aligned}$$

Here, (C.31) is obtained from (C.30) and holds for all $\tau > \tau_0$; (C.32) comes from the fact that $\|\boldsymbol{W}^{\perp}(\tau)\|_F > 0.5$; (C.33) follows that for any $a, b > 0$, $(a^2 - b^2)/2b > a - b$; (C.34) follows the projection property that $\|\boldsymbol{W}(\tau+1) - \boldsymbol{W}(\tau)\|_F^2 = \|\boldsymbol{W}^{\perp}(\tau+1) - \boldsymbol{W}^{\perp}(\tau)\|_F^2 + \|\boldsymbol{W}^{\parallel}(\tau+1) - \boldsymbol{W}^{\parallel}(\tau)\|_F^2$; and (C.35) is obtained via Lemma C.9.

Summing the above inequality over $\tau \geq \tau_0$ obtains

$$\left\langle \boldsymbol{W}^{\perp}(\tau) - \boldsymbol{W}^{\perp}(\tau_0), \frac{\boldsymbol{W}^{\mathrm{mm}}}{\|\boldsymbol{W}^{\mathrm{mm}}\|_F} \right\rangle \geq \frac{1}{1+\pi} \left(\|\boldsymbol{W}^{\perp}(\tau)\|_F - \|\boldsymbol{W}^{\perp}(\tau_0)\|_F + 2\eta \left(\mathcal{L}(\boldsymbol{W}(\tau)) - \mathcal{L}(\boldsymbol{W}(\tau_0)) \right) \right)$$

$$\Longrightarrow \left\langle \frac{\boldsymbol{W}^{\perp}(\tau)}{\|\boldsymbol{W}^{\perp}(\tau)\|_F}, \frac{\boldsymbol{W}^{\mathrm{mm}}}{\|\boldsymbol{W}^{\mathrm{mm}}\|_F} \right\rangle \geq \frac{1}{1+\pi} \left(1 + \frac{C}{\|\boldsymbol{W}^{\perp}(\tau)\|_F} \right)$$

where

$$C := \left\langle \boldsymbol{W}^{\perp}(\tau_0), \frac{\boldsymbol{W}^{\mathrm{mm}}}{\|\boldsymbol{W}^{\mathrm{mm}}\|_F} \right\rangle - \|\boldsymbol{W}^{\perp}(\tau_0)\|_F + 2\eta \left(\mathcal{L}(\boldsymbol{W}(\tau)) - \mathcal{L}(\boldsymbol{W}(\tau_0)) \right).$$

Since $\|\boldsymbol{W}^{\perp}(\tau)\|_F \to \infty$ and $0 < \mathcal{L}(\boldsymbol{W}(\tau)) \leq \mathcal{L}(\boldsymbol{W}(0)) < \infty$, we get

$$\lim_{\tau \to \infty} \left\langle \frac{\boldsymbol{W}^{\perp}(\tau)}{\|\boldsymbol{W}^{\perp}(\tau)\|_F}, \frac{\boldsymbol{W}^{\mathrm{mm}}}{\|\boldsymbol{W}^{\mathrm{mm}}\|_F} \right\rangle \geq \frac{1}{1+\pi}. \tag{C.36}$$

Choosing $\pi \to 0$ and combining (C.36) with the fact that $\lim_{\tau \to \infty} \|\boldsymbol{W}^{\|}(\tau)\|_F < \infty$ completes the proof.

■

C.3 Global Convergence of Regularization Path

C.3.1 Proof of Theorem 4.3

Lemma C.13 *Suppose Assumptions 4.1 and 4.2 hold. Additionally, assume loss $\ell : \mathbb{R} \to \mathbb{R}$ is strictly decreasing and $|\ell'|$ is bounded. Define $\bar{\boldsymbol{W}}_R^{\perp} := \bar{\boldsymbol{W}}_R^{\perp}(\boldsymbol{W}^{\|}) \in \mathcal{S}_{\mathit{fin}}^{\perp}$ by*

$$\bar{\boldsymbol{W}}_R^{\perp} := \arg \min_{\boldsymbol{W} \in \mathcal{S}_{\mathit{fin}}^{\perp}, \|\boldsymbol{W}\|_F \leq R} \mathcal{L}(\boldsymbol{W} + \boldsymbol{W}^{\|}). \tag{C.37}$$

Let $\boldsymbol{W}^{mm} \neq 0$ denote the solution of (Graph-SVM)*. Then we have that for any $\boldsymbol{W}^{\|} \in \mathcal{S}_{\mathit{fin}}$ with $\|\boldsymbol{W}^{\|}\|_F < \infty$*

$$\lim_{R \to \infty} \frac{\bar{\boldsymbol{W}}_R^{\perp}}{R} = \frac{\boldsymbol{W}^{mm}}{\|\boldsymbol{W}^{mm}\|_F}.$$

Proof. Recap $\mathcal{O}_i, \bar{\mathcal{O}}_i, \mathcal{R}_i, \bar{\mathcal{R}}_i$, $i \in [n]$ from (C.1). Let $\gamma_i = \boldsymbol{X}_i \boldsymbol{c}_{y_i}$ where following Assumption 4.1 we have

$$\gamma_{it} = \boldsymbol{x}_{it}^{\top} \boldsymbol{c}_{y_i} = \begin{cases} 1, & t \in \mathcal{O}_i \\ 0, & t \in \bar{\mathcal{O}}_i. \end{cases}$$

Recap $\mathcal{I}, \bar{\mathcal{I}}$ from (C.2). To proceed, define $\gamma_i^{\max}$ as follows:

- Consider $i \in \bar{\mathcal{I}}$. Given $\min_{\boldsymbol{W}} \ell(\boldsymbol{c}_{y_i}^\top \boldsymbol{X}_i^\top \mathbb{S}(\boldsymbol{X}_i \boldsymbol{W} \bar{\boldsymbol{x}}_i)) \geq \ell(\boldsymbol{c}_{y_i}^\top \boldsymbol{e}_{y_i})$, we define the maximal score $\gamma_i^{\max} := \boldsymbol{c}_{y_i}^\top \boldsymbol{e}_{y_i} = 1$.
- Consider $i \in \mathcal{I}$.
  1. Assumption 4.1 ensures that all tokens, excluding the ones with token ID $x_{it} = y_i$, return zero score, that is, $\boldsymbol{c}_{y_i}^\top \boldsymbol{e}_k = 0$ for $k \neq y_i$.
  2. From proof of Lemma C.1, for any $\boldsymbol{W}^\perp \in \mathcal{S}_{\text{fin}}^\perp$ and $t \in \mathcal{R}_i$, $\boldsymbol{x}_{it}^\top (\boldsymbol{W}^\perp + \boldsymbol{W}^\parallel) \bar{\boldsymbol{x}}_i = \boldsymbol{x}_{it}^\top \boldsymbol{W}^\parallel \bar{\boldsymbol{x}}_i + \bar{a}_i$, where $\bar{a}_i$ is some constant associated with $\boldsymbol{W}^\perp$ and remains the same value within the same SCC. Let $\boldsymbol{b}_i = \boldsymbol{X}_i \boldsymbol{W}^\parallel \bar{\boldsymbol{x}}_i$. Then the probabilities for $t \in \mathcal{R}_i$ (if denoted by s_{it}) obey

$$\frac{s_{it}}{\sum_{t' \in \mathcal{R}_i} s_{it'}} = \frac{e^{b_{it} + \bar{a}_i}}{\sum_{t' \in \mathcal{R}_i} e^{b_{it'} + \bar{a}_i}} = \frac{e^{b_{it}}}{\sum_{t' \in \mathcal{R}_i} e^{b_{it'}}}, \tag{C.38}$$

     which means that the probability distribution over set $\mathcal{R}_i$ remains the same with varying $\boldsymbol{W}^\perp$.

  Combining both, we define the maximal score as follows:

$$\gamma_i^{\max} := |\mathcal{O}_i| \cdot \bar{s}_i \quad \text{where} \quad \bar{s}_i = \frac{e^{\boldsymbol{e}_{y_i}^\top \boldsymbol{W}^\parallel \bar{\boldsymbol{x}}_i}}{\sum_{t' \in \mathcal{R}_i} e^{b_{it'}}}.$$

  Note that if consider the cyclic subdataset $\overline{\texttt{DSET}}$ as in (C.3). Let $(\bar{\boldsymbol{X}}_i, y_i) \in \overline{\texttt{DSET}}$ where $\bar{\boldsymbol{X}}_i$ is the corresponding sequence by removing the tokens in $\bar{\mathcal{R}}_i$. Then we have $\gamma_i^{\max} = \boldsymbol{c}_{y_i}^\top \bar{\boldsymbol{X}}_i^\top \mathbb{S}(\bar{\boldsymbol{X}}_i \boldsymbol{W}^\parallel \bar{\boldsymbol{x}}_i)$.

Hence, given $\boldsymbol{b}_i = \boldsymbol{X}_i \boldsymbol{W}^\parallel \bar{\boldsymbol{x}}_i$, we obtain

$$\gamma_i^{\max} = \begin{cases} 1, & i \in \bar{\mathcal{I}} \\ \frac{|\mathcal{O}_i| e^{\boldsymbol{e}_{y_i}^\top \boldsymbol{W}^\parallel \bar{\boldsymbol{x}}_i}}{\sum_{t' \in \mathcal{R}_i} e^{b_{it'}}}, & i \in \mathcal{I}. \end{cases}$$

Then we define the optimal risk of (C.37) and its corresponding softmax probabilities $s_{\max}(\rangle i$, $i \in [n]$ as follows:

$$\mathcal{L}_\star^{\boldsymbol{W}^\parallel} := \frac{1}{n} \sum_{i=1}^n \ell(\gamma_i^{\max}), \quad \text{and} \quad s_{it}^{\max} = \begin{cases} 0, & t \in \bar{\mathcal{R}}_i \\ \frac{e^{b_{it}}}{\sum_{t' \in \mathcal{R}_i} e^{b_{it'}}}, & t \in \mathcal{R}_i \end{cases} \quad \text{for all } i \in [n].$$

Note that we also have

$$\gamma_i^{\max} = \boldsymbol{c}_{y_i}^\top \boldsymbol{X}_i^\top s_{\max}()i = \sum_{t\in\mathcal{O}_i} s_{it}^{\max} = \frac{\sum_{t\in\mathcal{O}_i} e^{b_{it}}}{\sum_{t\in\mathcal{R}_i} e^{b_{it}}} \quad \text{and} \quad \mathcal{L}_\star^{\boldsymbol{W}^\parallel} = \frac{1}{n}\sum_{i\in\bar{\mathcal{I}}} \ell(1) + \bar{\mathcal{L}}(\boldsymbol{W}^\parallel) \quad \text{(C.39)}$$

where $\bar{\mathcal{L}}(\boldsymbol{W}^\parallel)$ is the empirical risk over cyclic subdataset defined in Definition C.2.

In the following, we will complete the proof in three steps.

Step 1: We first show that $\lim_{R\to\infty} \mathcal{L}(R\cdot \boldsymbol{W}^{\mathrm{mm}} + \boldsymbol{W}^\parallel) = \mathcal{L}_\star^{\boldsymbol{W}^\parallel}$. It can be easily proven using Lemma C.4 and (C.39) by showing that for any $\boldsymbol{W}^\parallel$ with $\|\boldsymbol{W}^\parallel\|_F < \infty$,

$$\lim_{R\to\infty} \mathcal{L}(R\cdot \boldsymbol{W}^{\mathrm{mm}} + \boldsymbol{W}^\parallel) = \min_{\boldsymbol{W}\in\mathcal{S}_{\mathrm{fin}}^\perp} \mathcal{L}(\boldsymbol{W} + \boldsymbol{W}^\parallel) = \frac{|\bar{\mathcal{I}}|}{n}\ell(1) + \bar{\mathcal{L}}(\boldsymbol{W}^\parallel) = \mathcal{L}_\star^{\boldsymbol{W}^\parallel}.$$

Step 2: Next, we will prove that for any $\boldsymbol{W}^\parallel \in \mathcal{S}_{\mathrm{fin}}$ with $\|\boldsymbol{W}^\parallel\|_F < \infty$, $\bar{\boldsymbol{W}}_R^\perp$ achieves the optimal risk as $R\to\infty$ – rather than problem having finite optima. It is to show that there is no finite R can achieve optimal risk. Consider any $\boldsymbol{W} \in \mathcal{S}_{\mathrm{fin}}^\perp$ with $\|\boldsymbol{W}\|_F < \infty$. Let $\boldsymbol{s}_i := \mathbb{S}(\boldsymbol{X}_i(\boldsymbol{W}+\boldsymbol{W}^\parallel)\bar{\boldsymbol{x}}_i)$ and $\gamma_i^{\boldsymbol{W}} := \boldsymbol{c}_{y_i}^\top \boldsymbol{X}_i^\top \boldsymbol{s}_i$. Then we have that

$$\gamma_i^{\boldsymbol{W}} = \sum_{t\in\mathcal{O}_i} s_{it} = \sum_{t\in\mathcal{O}_i} \frac{\sum_{t'\in\mathcal{R}_i} s_{it'}}{\sum_{t'\in[T_i]} s_{it'}} s_{it}^{\max} \le \sum_{t\in\mathcal{O}_i} s_{it}^{\max} = \gamma_i^{\max}$$

where the equality holds when $\mathcal{R}_i = [T_i]$. Since $\boldsymbol{W}^{\mathrm{mm}} \neq 0$, then there exists some $i\in[n]$ such that $\gamma_i^{\boldsymbol{W}} < \gamma_i^{\max}$. Therefore, for any finite $\boldsymbol{W}\in\mathbb{R}^{d\times d}$ and $\boldsymbol{W}\in\mathcal{S}_{\mathrm{fin}}^\perp$, since loss function is strictly decreasing

$$\mathcal{L}(\boldsymbol{W}+\boldsymbol{W}^\parallel) = \frac{1}{n}\sum_{i=1}^n \ell(\gamma_i^{\boldsymbol{W}}) > \frac{1}{n}\sum_{i=1}^n \ell(\gamma^{\max}) = \mathcal{L}_\star^{\boldsymbol{W}^\parallel}.$$

Step 3: Now, it remains to show that $\bar{\boldsymbol{W}}_R^\perp$ converges in direction to $\boldsymbol{W}^{\mathrm{mm}}$. Suppose convergence fails. We will obtain a contradiction by showing that $R\cdot\boldsymbol{W}^{\mathrm{mm}}/\|\boldsymbol{W}^{\mathrm{mm}}\|_F$ achieves a strictly superior loss compared to $\bar{\boldsymbol{W}}_R^\perp$ given sufficiently large R. Since $\bar{\boldsymbol{W}}_R^\perp$ fails to converge to $\boldsymbol{W}^{\mathrm{mm}}$, for some $\delta>0$, there exists arbitrarily large $R>0$ such that

$$\|\bar{\boldsymbol{W}}_R^\perp \cdot \|\boldsymbol{W}^{\mathrm{mm}}\|_F/R - \boldsymbol{W}^{\mathrm{mm}}\|_F \ge \delta.$$

Let $\boldsymbol{W}' = \bar{\boldsymbol{W}}_R^\perp \cdot \|\boldsymbol{W}^{\mathrm{mm}}\|_F/R$ where we have $\|\boldsymbol{W}'\|_F \le \|\boldsymbol{W}^{\mathrm{mm}}\|_F$ and $\boldsymbol{W}' \neq \boldsymbol{W}^{\mathrm{mm}}$. Following

Definition 4.1, we obtain

$$(\boldsymbol{e}_i - \boldsymbol{e}_j)^\top \boldsymbol{W}' \boldsymbol{e}_k = 0 \quad \text{where} \quad (i \asymp j) \in \mathcal{G}^{(k)}.$$

Then for some $\epsilon := \epsilon(\delta)$, there exists i, j, k such that

$$(\boldsymbol{e}_i - \boldsymbol{e}_j)^\top \boldsymbol{W}' \boldsymbol{e}_k \le 1 - \epsilon \quad \text{where} \quad (i \Rightarrow j) \in \mathcal{G}^{(k)}.$$

Now, we will argue that this leads to a contradiction by proving $\mathcal{L}(R \cdot \boldsymbol{W}^{\mathrm{mm}}/\|\boldsymbol{W}^{\mathrm{mm}}\|_F + \boldsymbol{W}^{\|}) < \mathcal{L}(\bar{\boldsymbol{W}}_R + \boldsymbol{W}^{\|})$ for sufficiently large R. Let $\Theta = 1/\|\boldsymbol{W}^{\mathrm{mm}}\|_F$ and we will show that $\mathcal{L}(R\Theta \cdot \boldsymbol{W}^{\mathrm{mm}} + \boldsymbol{W}^{\|}) < \mathcal{L}(R\Theta \cdot \boldsymbol{W}' + \boldsymbol{W}^{\|})$ for sufficiently large R.

To obtain the result, we establish a refined softmax probability control by studying the distance to $\mathcal{L}_\star^{\boldsymbol{W}^{\|}}$. Recap the definitions of $\gamma_i^{\max}$ and $s_{\max(}i$, and let $\boldsymbol{b}_i = \boldsymbol{X}_i \boldsymbol{W}^{\|} \bar{\boldsymbol{x}}_i$, $\boldsymbol{a}_i^\star = \boldsymbol{X}_i(R\Theta \cdot \boldsymbol{W}^{\mathrm{mm}})\bar{\boldsymbol{x}}_i$, $\boldsymbol{s}_i^\star = \mathbb{S}(\boldsymbol{X}_i(R\Theta \cdot \boldsymbol{W}^{\mathrm{mm}} + \boldsymbol{W}^{\|})\bar{\boldsymbol{x}}_i) = \mathbb{S}(\boldsymbol{a}_i^\star + \boldsymbol{b}_i)$, and $\gamma_i^\star := \boldsymbol{c}_{y_i}^\top \boldsymbol{X}_i^\top \boldsymbol{s}_i^\star$. Additionally, let $\boldsymbol{a}_i^R = \boldsymbol{X}_i(R\Theta \cdot \boldsymbol{W}')\bar{\boldsymbol{x}}_i$, $\boldsymbol{s}_i^R = \mathbb{S}(\boldsymbol{X}_i(R\Theta \cdot \boldsymbol{W}' + \boldsymbol{W}^{\|})\bar{\boldsymbol{x}}_i) = \mathbb{S}(\boldsymbol{a}_i^R + \boldsymbol{b}_i)$, and $\gamma_i^R := \boldsymbol{c}_{y_i}^\top \boldsymbol{X}_i^\top \boldsymbol{s}_i^R$.

Following Definition 4.1, we get for all $t, t' \in \mathcal{R}_i$

$$(\boldsymbol{x}_{it} - \boldsymbol{x}_{it'})^\top \boldsymbol{W} \bar{\boldsymbol{x}}_i = 0 \quad \text{for any } \boldsymbol{W} \perp \mathcal{S}_{\mathrm{fin}} \Longrightarrow a_{it}^\star - a_{it'}^\star = a_{it}^R - a_{it'}^R = 0.$$

Then

$$\sum_{t \in \mathcal{O}_i} s_{it}^R = \frac{\sum_{t \in \mathcal{O}_i} e^{a_{it}^R + b_{it}}}{\sum_{t \in [T_i]} e^{a_{it}^R + b_{it}}} \le \frac{\sum_{t \in \mathcal{O}_i} e^{b_{it}}}{\sum_{t \in \mathcal{R}_i} e^{b_{it}} + e^{-(1-\epsilon)R\Theta - \bar{b}}} \le \frac{c_i}{d_i + e^{-(1-\epsilon)R\Theta - \bar{b}}}, \quad \exists i \in [n]$$

$$\sum_{t \in \mathcal{O}_i} s_{it}^\star = \frac{\sum_{t \in \mathcal{O}_i} e^{a_{it}^\star + b_{it}}}{\sum_{t \in [T_i]} e^{a_{it}^\star + b_{it}}} \ge \frac{\sum_{t \in \mathcal{O}_i} e^{b_{it}}}{\sum_{t \in \mathcal{R}_i} e^{b_{it}} + \sum_{t \in \bar{\mathcal{R}}_i} e^{-R\Theta + b_{it}}} \ge \frac{c_i}{d_i + Te^{-R\Theta + \bar{b}}}, \quad \forall i \in [n],$$

where $c_i = \sum_{t \in \mathcal{O}_i} e^{b_{it}}$, $d_i = \sum_{t \in \mathcal{R}_i} e^{b_{it}}$, and $\bar{b} := \max_{t \in \mathcal{R}_i, i \in [n]} |b_{it}|$, and we have $\gamma_i^{\max} = c_i/d_i$.

Since ℓ is strictly decreasing and $|\ell'|$ is bounded, let $c_{\mathrm{dn}} \le -\ell' \le c_{\mathrm{up}}$ for some constants $c_{\mathrm{dn}}, c_{\mathrm{up}} > 0$. Note that $c_{\mathrm{dn}}, c_{\mathrm{up}}$ are data-dependent. Then we have

$$\begin{aligned}
\mathcal{L}(R\Theta \cdot \boldsymbol{W}' + \boldsymbol{W}^{\|}) - \mathcal{L}_\star^{\boldsymbol{W}^{\|}} &\ge \frac{1}{n}\left(\ell(\gamma_i^R) - \ell(\gamma_i^{\max})\right) \ge \frac{c_{\mathrm{dn}}}{n}\left(\gamma_i^{\max} - \gamma_i^R\right) \\
&= \frac{c_{\mathrm{dn}}}{n}\left(\gamma_i^{\max} - \sum_{t \in \mathcal{O}_i} s_{it}^R\right) \\
&\ge \frac{c_{\mathrm{dn}} \gamma_i^{\max}}{n}\left(1 - \frac{1}{1 + e^{-(1-\epsilon)R\Theta - \bar{b}}/d_i}\right)
\end{aligned}$$

and let $j := \max_{i\in[n]} (\ell(\gamma_i^\star) - \ell(\gamma_i^{\max}))$. We can upper-bound the loss difference for $R\Theta \cdot \boldsymbol{W}^{\mathrm{mm}} + \boldsymbol{W}^{\|}$ as follows:

$$\begin{aligned}
\mathcal{L}(R\Theta \cdot \boldsymbol{W}^{\mathrm{mm}} + \boldsymbol{W}^{\|}) - \mathcal{L}_\star^{\boldsymbol{W}^{\|}} &\le \max_{i\in[n]} (\ell(\gamma_i^\star) - \ell(\gamma_i^{\max})) \le c_{\mathrm{up}} \left(\gamma_j^{\max} - \gamma_j^\star\right) \\
&= c_{\mathrm{up}} \left(\gamma_j^{\max} - \sum_{t\in\mathcal{O}_j} s_{it}^\star\right) \\
&\le c_{\mathrm{up}} \gamma_j^{\max} \left(1 - \frac{1}{1 + Te^{-R\Theta+\bar{b}}/d_j}\right) \\
&\le \frac{c_{\mathrm{up}} \gamma_j^{\max} T}{d_j} e^{-R\Theta+\bar{b}}.
\end{aligned}$$

Combining them together results in that, $\mathcal{L}(R\Theta\cdot\boldsymbol{W}' + \boldsymbol{W}^{\|}) > \mathcal{L}(R\Theta\cdot\boldsymbol{W}^{\mathrm{mm}} + \boldsymbol{W}^{\|})$ whenever

$$\begin{aligned}
&\frac{c_{\mathrm{dn}}\gamma_i^{\max}}{n} \left(1 - \frac{1}{1 + e^{-(1-\epsilon)R\Theta-\bar{b}}/d_i}\right) > \frac{c_{\mathrm{up}}\gamma_j^{\max} T}{d_j} e^{-R\Theta+\bar{b}} \\
&\implies R > R_\epsilon := \frac{1}{\Theta \cdot \min(\epsilon, 1)} \log\left(\frac{2nTc_{\mathrm{up}}\gamma_j^{\max} \cdot \max(d_i, 1)}{c_{\mathrm{dn}}\gamma_i^{\max} d_j}\right) + \frac{2\bar{b}}{\Theta \cdot \min(\epsilon, 1)}. \qquad \text{(C.40)}
\end{aligned}$$

Note that since $\boldsymbol{W}^{\|}$ is finite, b_{it} for all $i \in [n], t \in [T_i]$ are bounded and fixed, and therefore, $0 < d_i < \infty$, for all $i \in [n]$ and $\bar{b} < \infty$. (C.40) completes the proof by contradiction. ∎

Now, gathering all the results we have obtained so far, we are ready to prove Theorem 4.3.

Proof of Theorem 4.3. Recap the dataset $\texttt{DSET} = (\boldsymbol{X}_i, y_i)_{i=1}^n$ and index sets $\mathcal{O}_i, \bar{\mathcal{O}}_i, \mathcal{R}_i, \bar{\mathcal{R}}_i$, $i \in [n]$ from (C.1). Let $\gamma_i = \boldsymbol{X}_i \boldsymbol{c}_{y_i}$ denote the score vector of i-th input. Since Assumption 4.1 holds, then

$$\gamma_{it} = \boldsymbol{x}_{it}^\top \boldsymbol{c}_{y_i} = \begin{cases} 1, & t \in \mathcal{O}_i \\ 0, & t \in \bar{\mathcal{O}}_i. \end{cases}$$

Let $\boldsymbol{s}_i^{\boldsymbol{W}} = \mathbb{S}(\boldsymbol{X}_i \boldsymbol{W} \bar{\boldsymbol{x}}_i)$. The regularization path solution of the ERM problem is defined as follows:

$$\bar{\boldsymbol{W}}_R = \arg\min_{\|\boldsymbol{W}\|_F \le R} \mathcal{L}(\boldsymbol{W}) \quad \text{where} \quad \mathcal{L}(\boldsymbol{W}) = \frac{1}{n}\sum_{i=1}^n \ell(\boldsymbol{c}_{y_i}^\top \boldsymbol{X}_i^\top \mathbb{S}(\boldsymbol{X}_i \boldsymbol{W} \bar{\boldsymbol{x}}_i)) = \frac{1}{n}\sum_{i=1}^n \ell\left(\sum_{t\in\mathcal{O}_i} s_{it}^{\boldsymbol{W}}\right).$$

Additionally, let $\boldsymbol{W}_R^\perp = \boldsymbol{\Pi}_{\mathcal{S}_{\mathrm{fin}}^\perp}(\bar{\boldsymbol{W}}_R)$ and $\boldsymbol{W}_R^{\|} = \boldsymbol{\Pi}_{\mathcal{S}_{\mathrm{fin}}}(\bar{\boldsymbol{W}}_R)$. Lemma C.13 has shown that for

any finite $\lim_{R\to\infty} \boldsymbol{W}_R^{\|}$,

$$\lim_{R\to\infty} \frac{\bar{\boldsymbol{W}}_R}{R} = \lim_{R\to\infty} \frac{\boldsymbol{W}_R^{\perp}}{\sqrt{R^2 - \|\boldsymbol{W}_R^{\|}\|_F^2}} = \lim_{R\to\infty} \frac{\boldsymbol{W}_R^{\perp}}{\|\boldsymbol{W}_R^{\perp}\|_F} = \frac{\boldsymbol{W}^{\mathrm{mm}}}{\|\boldsymbol{W}^{\mathrm{mm}}\|_F}.$$

Therefore it remains to prove that $\lim_{R\to\infty} \boldsymbol{W}_R^{\|} \in \mathcal{W}^{\mathrm{fin}}$.

Suppose $\lim_{R\to\infty} \boldsymbol{W}_R^{\|} := \boldsymbol{W}' \notin \mathcal{W}^{\mathrm{fin}}$. Then for any $\boldsymbol{W}^{\|} \in \mathcal{W}^{\mathrm{fin}}$, applying Lemma C.4, we obtain

$$\min_{\boldsymbol{W}^{\perp}\in\mathcal{S}_{\mathrm{fin}}^{\perp}} \mathcal{L}(\boldsymbol{W}^{\perp} + \boldsymbol{W}') > \min_{\boldsymbol{W}^{\perp}\in\mathcal{S}_{\mathrm{fin}}^{\perp}} \mathcal{L}(\boldsymbol{W}^{\perp} + \boldsymbol{W}^{\|}) = \lim_{R\to\infty} \mathcal{L}(R\cdot\boldsymbol{W}^{\mathrm{mm}} + \boldsymbol{W}^{\|}).$$

Therefore, $\boldsymbol{W}'$ does not achieve the minimal loss as $R \to \infty$. ∎

C.3.2 Proof of Lemma 4.3

Proof. Let $\gamma_i = \boldsymbol{X}_i \boldsymbol{c}_{y_i}$ denote the score vector of i-th input and $\gamma_{it} = \boldsymbol{x}_{it}^{\top} \boldsymbol{c}_{y_i}$. Let $\gamma_i^{\max} = \boldsymbol{e}_{y_i}^{\top} \boldsymbol{c}_{y_i} = \max_{t\in[T_i]} \gamma_{it}$ following Assumptions 4.2 and 4.3. What's more, since loss ℓ is strictly decreasing, we define the optimal loss as follows:

$$\mathcal{L}_\star := \frac{1}{n}\sum_{i=1}^{n} \ell(\gamma_i^{\max}).$$

For any $\boldsymbol{W} \in \mathbb{R}^{d\times d}$, let $\boldsymbol{s}_i = \boldsymbol{X}_i \boldsymbol{W} \bar{\boldsymbol{x}}_i$, $i \in [n]$. If $\|\boldsymbol{W}\|_F < \infty$, then $\min_{t\in[T_i], i\in[n]} s_{it} > 0$ and for any $i \in [n]$

$$\boldsymbol{s}_i^{\top}\gamma_i = \sum_{t=1}^{T_i} s_{it}\gamma_{it} < \gamma_i^{\max}.$$

Since loss function ℓ is strictly decreasing, we get

$$\mathcal{L}(\boldsymbol{W}) = \frac{1}{n}\sum_{i=1}^{n} \ell(\boldsymbol{s}_i^{\top}\gamma_i) > \frac{1}{n}\sum_{i=1}^{n} \ell(\gamma_i^{\max}) = \mathcal{L}_\star.$$

Let $\boldsymbol{W}$ be any attention weight satisfying all the "≥ 1" constraints in (Acyc-SVM). We next prove that $\lim_{R\to\infty} \mathcal{L}(R\cdot\boldsymbol{W}) = \mathcal{L}_\star$. Recap $\mathcal{O}_i$ and $\bar{\mathcal{O}}_i$ from (C.1). Since token $\boldsymbol{e}_{y_i}$ is always contained in $\boldsymbol{X}_i$ following Assumption 4.2, we have $|\mathcal{O}_i| \geq 1, i \in [n]$, and $\boldsymbol{X}_i$ contains $|\mathcal{O}_i|$ optimal tokens $\boldsymbol{e}_{y_i}$. Note that under acyclic data setting, $\boldsymbol{W}$ separates tokens $\boldsymbol{e}_{y_i}$ from the rest of the tokens within $\boldsymbol{X}_i$. Then $\lim_{R\to\infty} \mathbb{S}(\boldsymbol{X}_i(R\cdot\boldsymbol{W})\bar{\boldsymbol{x}}_i)$ will output $1/|\mathcal{O}_i|$ for $t \in \mathcal{O}_i$ and zero for the left. Specifically, let $\boldsymbol{s}_i^R := \mathbb{S}(\boldsymbol{X}_i(R\cdot\boldsymbol{W})\bar{\boldsymbol{x}}_i)$, and following the SVM objective

(Acyc-SVM) for any $i \in [n]$, we get

$$s_{it}^R = \frac{e^{\boldsymbol{x}_{it}^\top (R\cdot \boldsymbol{W})\bar{\boldsymbol{x}}_i}}{\sum_{t\in[T_i]} e^{\boldsymbol{x}_{it}^\top (R\cdot \boldsymbol{W})\bar{\boldsymbol{x}}_i}} = \frac{1}{|\mathcal{O}_i| + \sum_{t\in\bar{\mathcal{O}}_i} e^{(\boldsymbol{x}_{it}-\boldsymbol{e}_{y_i})^\top (R\cdot \boldsymbol{W})\bar{\boldsymbol{x}}_i}} \geq \frac{1}{|\mathcal{O}_i| + e^{-R}} \quad \text{for all } t \in \mathcal{O}_i$$

$$\text{and then,} \quad \sum_{t\in\mathcal{O}_i} s_{it}^R = \frac{|\mathcal{O}_i|}{|\mathcal{O}_i| + \sum_{t\in\bar{\mathcal{O}}_i} e^{(\boldsymbol{x}_{it}-\boldsymbol{e}_{y_i})^\top (R\cdot \boldsymbol{W})\bar{\boldsymbol{x}}_i}} \geq \frac{1}{1+e^{-R}}.$$

Then $\lim_{R\to\infty} \sum_{t\in\mathcal{O}_i} s_{it}^R = 1$ and therefore,

$$\lim_{R\to\infty} s_{it}^R = 1/|\mathcal{O}_i| \quad \text{for } t \in \mathcal{O}_i, \qquad \text{and} \qquad \lim_{R\to\infty} s_{it}^R = 0 \quad \text{for } t \in \bar{\mathcal{O}}_i.$$

Hence we have

$$\lim_{R\to\infty} \boldsymbol{X}_i^\top \boldsymbol{s}_i^R = \sum_{t\in\mathcal{O}_i} \frac{1}{|\mathcal{O}_i|} \boldsymbol{e}_{y_i} = \boldsymbol{e}_{y_i}.$$

Since $|\ell'|$ is bounded, then $\lim_{R\to\infty} \mathcal{L}(R\cdot \boldsymbol{W}) = \frac{1}{n}\sum_{i=1}^n \ell(\boldsymbol{c}_{y_i}^\top \boldsymbol{e}_{y_i}) = \frac{1}{n}\sum_{i=1}^n \ell(\gamma_i^{\max}) = \mathcal{L}_\star$. ■

C.3.3 Proof of Theorem 4.4

Proof. Recap the dataset $\texttt{DSET} = (\boldsymbol{X}_i, y_i)_{i=1}^n$. The regularization path solution of the ERM problem (per Algo-RP and (ERM)) is defined as follows:

$$\bar{\boldsymbol{W}}_R = \arg\min_{\|\boldsymbol{W}\|_F \leq R} \mathcal{L}(\boldsymbol{W}) \quad \text{where} \quad \mathcal{L}(\boldsymbol{W}) = \frac{1}{n}\sum_{i=1}^n \ell(\boldsymbol{c}_{y_i}^\top \boldsymbol{X}_i^\top \mathbb{S}(\boldsymbol{X}_i \boldsymbol{W} \bar{\boldsymbol{x}}_i)).$$

The proof is similar to the proof of Theorem 2 in [185] by choosing $\text{opt}_i = y_i$. However in our work, we allow each sequence contains more than one optimal tokens, while [185] forces that the optimal token is unique.

Following the proof in Lemma 4.3, let $\gamma_i = \boldsymbol{X}_i \boldsymbol{c}_{y_i}$, $\gamma_i^{\max} = \boldsymbol{e}_{y_i}^\top \boldsymbol{c}_{y_i} = \max_{t\in[T_i]} \gamma_{it}$, and the optimal training risk

$$\mathcal{L}_\star := \frac{1}{n}\sum_{i=1}^n \ell(\gamma_i^{\max}).$$

From Lemma 4.3, we have that for any finite $\boldsymbol{W}$, $\mathcal{L}(\boldsymbol{W}) < \lim_{R\to\infty} \mathcal{L}(R\cdot \boldsymbol{W}^{\text{mm}}) = \mathcal{L}^\star$. Then the optimal risk $\mathcal{L}_\star$ is achievable and to achieve the limit, R has to be infinite. Then it remains to prove that $\bar{\boldsymbol{W}}_R$ converges in direction to $\boldsymbol{W}^{\text{mm}}$.

Suppose convergence fails. We will obtain a contradiction by showing that $R \cdot \boldsymbol{W}^{\text{mm}}/\|\boldsymbol{W}^{\text{mm}}\|_F$ achieves a strictly superior loss compared to $\bar{\boldsymbol{W}}_R$. Suppose $\bar{\boldsymbol{W}}_R$ fails to directionally converge towards $\boldsymbol{W}^{\text{mm}}$. For some $\delta > 0$, there exists arbitrarily large $R > 0$

such that

$$\|\bar{\boldsymbol{W}}_R \cdot \|\boldsymbol{W}^{\mathrm{mm}}\|_F/R - \boldsymbol{W}^{\mathrm{mm}}\|_F \geq \delta.$$

Let $\boldsymbol{W}' = \bar{\boldsymbol{W}}_R \cdot \|\boldsymbol{W}^{\mathrm{mm}}\|_F/R$ where we have $\|\boldsymbol{W}'\|_F \leq \|\boldsymbol{W}^{\mathrm{mm}}\|_F$ and $\boldsymbol{W}' \neq \boldsymbol{W}^{\mathrm{mm}}$. Since $\boldsymbol{W}^{\mathrm{mm}}$ is the min-norm solution of (Acyc-SVM), then for some $\epsilon := \epsilon(\delta)$, there exists i, j, k such that

$$(\boldsymbol{e}_i - \boldsymbol{e}_j)^\top \boldsymbol{W}' \boldsymbol{e}_k \leq 1 - \epsilon \quad \text{where} \quad (i \Rightarrow j) \in \mathcal{G}^{(k)}.$$

Now, we will argue that this leads to a contradiction by proving $\mathcal{L}(R \cdot \boldsymbol{W}^{\mathrm{mm}}/\|\boldsymbol{W}^{\mathrm{mm}}\|_F) < \mathcal{L}(\bar{\boldsymbol{W}}_R)$ for sufficiently large R. Let $\Theta = 1/\|\boldsymbol{W}^{\mathrm{mm}}\|_F$ and we will show that $\mathcal{L}(R\Theta \cdot \boldsymbol{W}^{\mathrm{mm}}) < \mathcal{L}(R\Theta \cdot \boldsymbol{W}')$ for sufficiently large R.

To obtain the result, we establish a refined softmax probability control as in the proof of Theorem 4.3 by studying the distance to $\mathcal{L}_\star$. Let $\boldsymbol{a}_i^\star = \boldsymbol{X}_i(R\Theta \cdot \boldsymbol{W}^{\mathrm{mm}})\bar{\boldsymbol{x}}_i$, $\boldsymbol{a}_i^R = \boldsymbol{X}_i(R\Theta \cdot \boldsymbol{W}')\bar{\boldsymbol{x}}_i$, $\boldsymbol{s}_i^\star := \mathbb{S}(\boldsymbol{a}_i^\star)$, $\boldsymbol{s}_i^R := \mathbb{S}(\boldsymbol{a}_i^R)$, $\gamma_i^\star := \boldsymbol{c}_{y_i}^\top \boldsymbol{X}_i^\top \boldsymbol{s}_i^\star$, and $\gamma_i^R := \boldsymbol{c}_{y_i}^\top \boldsymbol{X}_i^\top \boldsymbol{s}_i^R$. Recap that $\gamma_i^{\max} = \boldsymbol{e}_{y_i}^\top \boldsymbol{c}_{y_i}$. Then

$$\sum_{t\in\mathcal{O}_i} s_{it}^R = \frac{\sum_{t\in\mathcal{O}_i} e^{a_{it}^R}}{\sum_{t\in[T_i]} e^{a_{it}^R}} \leq \frac{|\mathcal{O}_i|}{|\mathcal{O}_i| + e^{-(1-\epsilon)R\Theta}} \leq \frac{1}{1 + e^{-(1-\epsilon)R\Theta}/T}, \quad \exists i \in [n] \tag{C.41}$$

$$\sum_{t\in\mathcal{O}_i} s_{it}^\star = \frac{\sum_{t\in\mathcal{O}_i} e^{a_{it}^\star}}{\sum_{t\in[T_i]} e^{a_{it}^\star}} \geq \frac{|\mathcal{O}_i|}{|\mathcal{O}_i| + (T - |\mathcal{O}_i|)e^{-R\Theta}} \geq \frac{1}{1 + Te^{-R\Theta}}, \quad \forall i \in [n]. \tag{C.42}$$

Since ℓ is strictly decreasing and $|\ell'|$ is bounded, let $c_{\mathrm{dn}} \leq -\ell' \leq c_{\mathrm{up}}$ for some constants $c_{\mathrm{dn}}, c_{\mathrm{up}} > 0$. Note that $c_{\mathrm{dn}}, c_{\mathrm{up}}$ are data-dependent. Additionally, define the minimal/maximal score gaps as

$$c_{\min} = \min_{y,k\in[K],y\neq k} (\boldsymbol{e}_y - \boldsymbol{e}_k)^\top \boldsymbol{c}_y, \quad c_{\max} = \max_{y,k\in[K],y\neq k} (\boldsymbol{e}_y - \boldsymbol{e}_k)^\top \boldsymbol{c}_y$$

where $c_{\max} \geq c_{\min} > 0$. Then we have that there exists $i \in [n]$,

$$\begin{aligned}
\mathcal{L}(R\Theta \cdot \boldsymbol{W}') - \mathcal{L}_\star &\geq \frac{1}{n}\left(\ell(\gamma_i^R) - \ell(\gamma_i^{\max})\right) \geq \frac{c_{\mathrm{dn}}}{n}\left(\gamma_i^{\max} - \gamma_i^R\right) \\
&\geq \frac{c_{\mathrm{dn}}}{n} c_{\min}\left(1 - \sum_{t\in\mathcal{O}_i} s_{it}^R\right) \geq \frac{c_{\mathrm{dn}} c_{\min}}{n} \frac{1}{1 + Te^{(1-\epsilon)R\Theta}}
\end{aligned} \tag{C.43}$$

and letting $j := \arg\max_{i\in[n]} (\ell(\gamma_i^\star) - \ell(\gamma_i^{\max}))$, we can upper-bound the loss difference for

$R\Theta \cdot \boldsymbol{W}^{\mathrm{mm}}$ as follows:

$$\begin{aligned}\mathcal{L}(R\Theta \cdot \boldsymbol{W}^{\mathrm{mm}}) - \mathcal{L}_\star &\leq \max_{i\in[n]} \left(\ell(\gamma_i^\star) - \ell(\gamma_i^{\max})\right) \leq c_{\mathrm{up}} \left(\gamma_j^{\max} - \gamma_j^\star\right) \\ &\leq c_{\mathrm{up}} c_{\max} \left(1 - \sum_{t\in\mathcal{O}_i} s_{it}^\star\right) \leq c_{\mathrm{up}} c_{\max} \frac{1}{1 + e^{R\Theta}/T} \leq c_{\mathrm{up}} c_{\max} T e^{-R\Theta}.\end{aligned} \tag{C.44}$$

Combining them together results in that, $\mathcal{L}(R\Theta \cdot \boldsymbol{W}') > \mathcal{L}(R\Theta \cdot \boldsymbol{W}^{\mathrm{mm}})$ whenever

$$\frac{c_{\mathrm{dn}} c_{\min}}{n} \frac{1}{1 + Te^{(1-\epsilon)R\Theta}} > c_{\mathrm{up}} c_{\max} T e^{-R\Theta} \Longrightarrow R > \frac{1}{\Theta \cdot \min(\epsilon, 1)} \log\left(\frac{2nT^2 c_{\mathrm{up}} c_{\max}}{c_{\mathrm{dn}} c_{\min}}\right).$$

This completes the proof by contradiction. ∎

C.4 Implementation Details and Additional Experiments

C.4.1 Implementation Details

In all the experiments, we train single-layer self-attention layer models using PyTorch and SGD optimizer. We conduct normalized gradient descent method to enhance the increasing of the norm of attention weight, so that softmax can easily saturate. Specifically, at each iteration τ, we update attention weight $\boldsymbol{W}$ via

$$\boldsymbol{W}(\tau + 1) = \boldsymbol{W}(\tau) - \eta \frac{\nabla \mathcal{L}(\boldsymbol{W}(\tau))}{\|\nabla \mathcal{L}(\boldsymbol{W}(\tau))\|_F}.$$

All the results are averaged over 100 random trails and in each trail, we create the dataset and its corresponding TPGs, SCCs as follows:

1. Given dimension d and vocabulary size K, generate random embedding table $\boldsymbol{E} = [\boldsymbol{e}_1 \ \cdots \ \boldsymbol{e}_K]^\top \in \mathbb{R}^{K\times d}$ such that each $\boldsymbol{e} \in \boldsymbol{E}$ is randomly sampled from unit sphere.

2. Given sample size n and sequence length T, create dataset $\texttt{DSET} = (\boldsymbol{X}_i, y_i)_{i=1}^n$ and $\boldsymbol{X}_i = [\boldsymbol{x}_{it} \ \cdots \ \boldsymbol{x}_{iT}]^\top \in \mathbb{R}^{T\times d}$ where $\boldsymbol{x}_{it}$ are randomly sampled from $\boldsymbol{E}$. For acyclic setting, label $\boldsymbol{e}_{y_i}$ is determined by the token in the $\boldsymbol{X}_i$ that has the highest priority order; while for general cyclic setting, $\boldsymbol{e}_{y_i}$ are also randomly sampled from $\boldsymbol{X}_i$.

3. Construct TPGs and apply Tarjan's algorithm [184] to find SCCs of each TPG. For global convergence experiments (Section 4.3), TPGs are created based on the token relations between $\boldsymbol{x}_{it}$'s and $\boldsymbol{e}_{y_i}$'s in the dataset DSET; while for local convergence analysis

(Section 4.5), we instead establish the token relations between $\boldsymbol{x}_{it}$'s and $\hat{\boldsymbol{e}}_{y_i}$'s following the instruction in Section 4.5, where $\hat{\boldsymbol{e}}_{y_i}$ is determined by the GD solution.

4. $\overline{\texttt{DSET}}$ is created following Definition C.1 based on the SCCs of the corresponding TPGs.

Here, we set the sequence length to be the same for all the samples in DSET, and we emphasize that though DSET contains inputs with same number of tokens, the randomness in sampling $\boldsymbol{x}_{it}$ and $\boldsymbol{e}_{y_i}$ will still result in a variety of TPGs and SCCs, and $\overline{\texttt{DSET}}$ may contain inputs with varying sequence lengths (see Figure 4.3).

• **Generating $\boldsymbol{W}^{\text{fin}}$ and $\widetilde{\boldsymbol{W}}^{\text{fin}}$.** Inspired by the convexity and finiteness of $\bar{\mathcal{L}}(\boldsymbol{W})$ per Definition C.2 under the setting of Theorem 4.2, we can derive $\boldsymbol{W}^{\text{fin}}$ via gradient descent. Hence, to obtain $\boldsymbol{W}^{\text{fin}}$, we train separate models but with the same architecture from zero initialization on the sub-dataset $\overline{\texttt{DSET}}$. As for the experiments shown in Section 4.5, we follow the same method as generating $\boldsymbol{W}^{\text{fin}}$. However, we emphasize that under the local convergence setting, there is no guarantee that gradient descent will converge to the $\widetilde{\boldsymbol{W}}^{\text{fin}}$ solution as problem is more general, i.e., with nonconvex head, and dataset $\overline{\texttt{DSET}}$ might not be enough to capture the performance of tokens within the same SCCs. Though, our results in Figures 4.6 and 4.7 indicate that $\widetilde{\boldsymbol{W}}^{\text{fin}}$ can predict the GD convergence performance better than $\boldsymbol{W}^{\text{fin}}$ which is drawn from the dataset-based TPGs. We defer a rigorous definition of local $\widetilde{\boldsymbol{W}}^{\text{fin}}$ and guarantees related to gradient descent for future exploration.

• **Local convergence experiments (Figures 4.6 and 4.7).** To evaluate our local convergence conjecture, we conduct random experiments with more general head (satisfying Assumption 4.3) and, and consider squared loss $\ell(u) = (1-u)^2$ in Figure 4.6 and cross-entropy loss in Figure 4.7. In both experiment, we create embedding tabels with $K = 8, d = 8$ and datasets with $n = 4, T = 6$. We choose step size $\eta = 0.1$ and also conduct normalized gradient descent. Correlations are reported in Figs. 4.6a and 4.7a and the distance of $\|\boldsymbol{\Pi}_{\widetilde{\mathcal{S}}_{\text{fin}}}(\boldsymbol{W}(\tau)) - \widetilde{\boldsymbol{W}}^{\text{fin}}\|_F$ are presented in the orange curves in Figs. 4.6b and 4.7b. In both experiments, correlations between $\frac{\boldsymbol{W}(\tau)}{\|\boldsymbol{W}(\tau)\|_F}$ and $\frac{\widetilde{\boldsymbol{W}}^{\text{svm}}}{\|\widetilde{\boldsymbol{W}}^{\text{svm}}\|_F}$ end with > 0.99 values. Fig. 4.6b achieves 0 distance error since employing squared loss, attention is inclined to select tokens that appear mostly frequently in the labels of the dataset, resulting in $\mathcal{R}_i = \mathcal{O}_i$ for $i \in [n]$ and $\overline{\texttt{DSET}} = \emptyset$. While in Fig. 4.7b, the global and local norm of difference is around 9.59 and 0.09 respectively, where $\overline{\texttt{DSET}} \neq \emptyset$. This implies that the distance of $\boldsymbol{\Pi}_{\widetilde{\mathcal{S}}_{\text{fin}}}(\boldsymbol{W}(\tau))$ is much closer to $\widetilde{\boldsymbol{W}}^{\text{fin}}$ compared to the distance between $\boldsymbol{\Pi}_{\mathcal{S}_{\text{fin}}}(\boldsymbol{W}(\tau))$ and $\boldsymbol{W}^{\text{fin}}$.

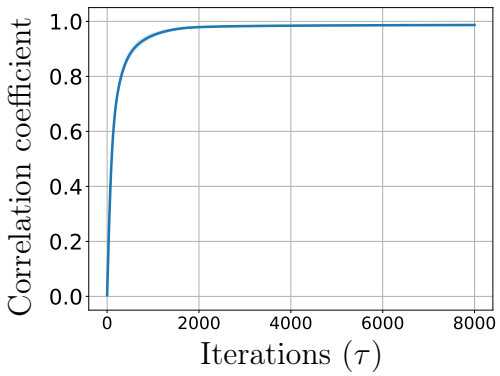


Figure C.2: $\frac{\boldsymbol{W}(\tau)}{\|\boldsymbol{W}(\tau)\|_F} \to \frac{\boldsymbol{W}^{\mathrm{mm}}}{\|\boldsymbol{W}^{\mathrm{mm}}\|_F}$

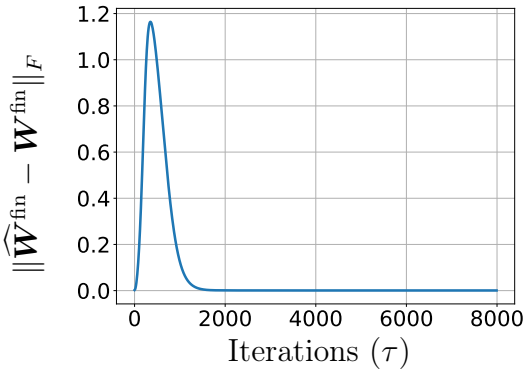


Figure C.3: $\boldsymbol{\Pi}_{\mathcal{S}_{\mathrm{fin}}}(\boldsymbol{W}(\tau)) \to \boldsymbol{W}^{\mathrm{fin}}$

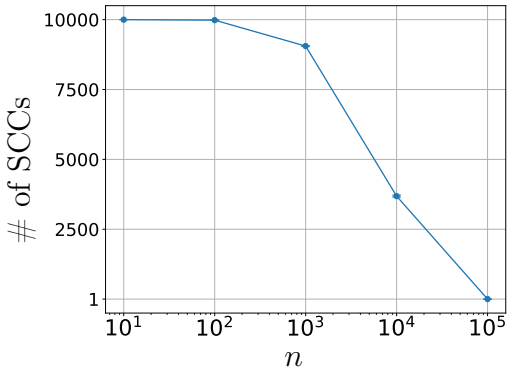


Figure C.4: Number of of SCCs vs n

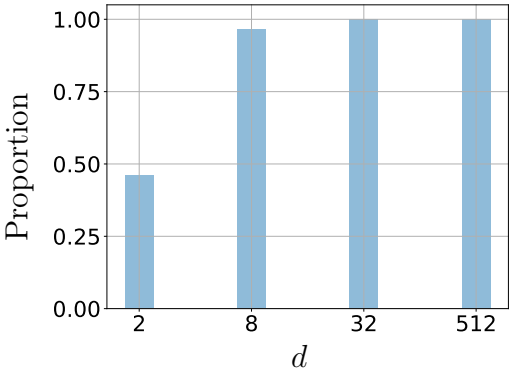


Figure C.5: Feasibility of $\boldsymbol{W}^{\mathrm{mm}}$

C.4.2 Additional Experiments

Global convergence experiments on large K (Figures C.2 and C.3). Assumption 4.1 in our work requires $K \leq d$, which helps make the optimization landscape more benign such that global convergence of GD is guaranteed. Unlike previous work [186, 185] that relies on strong equal score conditions to induce global convergence, our assumption is much less strict. Empirically, we argue that this constraint is not necessary as we can apply a mask $\boldsymbol{M} \in \mathbb{R}^{n \times K \times T}$ to directly collect the attention probability for each distinct token from the attention map without explicitly calculating the linear head. Therefore, we can still impose Assumption 4.1 when $K > d$, which aligns more closely with the real-world setting. In Figs. C.2 and C.3, we repeat the global convergence experiments by setting $n = 16, T = 64, d = 128$ and $K = 10000$. Results are averaged over 100 random instances. The averaged correlation is ≈ 0.987 and the soft component error reaches 0.025. The results again validate Theorem 4.2.

SCC Structure on large n (Figure C.4). When we increase the sample size n and fix others, more edges and cycles will be added to the graphs. Different SCCs can then be merged into one. As illustrated in Fig. C.4, the graph eventually collapses to a single SCC.

Feasible condition of (Graph-SVM) **(Figure C.5).** To verify that (Graph-SVM) is feasible when $d \geq K$ (Lemma 4.1), in Fig. C.5, we run experiments with fixed $n = 16, T = 128, K = 512$ and varying d from 2 to 512. Define $\mathcal{C}_y$ as the SCC that the label token belongs to. We calculate the proportion of selected tokens that are in $\mathcal{C}_y$ to the size of $\mathcal{C}_y$, and (Graph-SVM) is feasible when the value reaches 1. The interpretation is that: When d is small, the problem focuses on separating an optimally feasible subset of training data from the others and the empirical SVM bias is captured by a relaxed Graph-SVM solution with constraints based on the subset. As d grows, the exact Graph-SVM becomes feasible. This is similar to the findings in [90].

APPENDIX D

Appendix for Chapter 5

D.1 Equivalence among Gradient Descent, Attention, and State-Space Models

In this section, we present the proofs related to Section 5.2. Recap that given data

$$
\begin{aligned}
&\boldsymbol{X} = [\boldsymbol{x}_1 \ \cdots \ \boldsymbol{x}_n]^\top \in \mathbb{R}^{n \times d}, \\
&\boldsymbol{\xi} = [\xi_1 \ \cdots \ \xi_n]^\top \in \mathbb{R}^n, \\
&\boldsymbol{y} = [y_1 \ \cdots \ y_n]^\top = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\xi} \in \mathbb{R}^n, \\
&\boldsymbol{Z}_0 = [\boldsymbol{z}_1 \ \ldots \ \boldsymbol{z}_n \ \boldsymbol{0}_{d+1}]^\top = \begin{bmatrix} \boldsymbol{x}_1 & \ldots & \boldsymbol{x}_n & \boldsymbol{0}_d \\ y_1 & \ldots & y_n & 0 \end{bmatrix}^\top \in \mathbb{R}^{(n+1)\times(d+1)},
\end{aligned}
$$

and corresponding prediction functions

$$
\begin{aligned}
g_{\text{pgd}}(\boldsymbol{Z}) &= \boldsymbol{x}^\top \boldsymbol{W} \boldsymbol{X}^\top \boldsymbol{y}, && \text{(D.1a)} \\
g_{\text{wpgd}}(\boldsymbol{Z}) &= \boldsymbol{x}^\top \boldsymbol{W} \boldsymbol{X}^\top (\boldsymbol{\omega} \odot \boldsymbol{y}), && \text{(D.1b)} \\
g_{\text{att}}(\boldsymbol{Z}) &= (\boldsymbol{z}^\top \boldsymbol{W}_q \boldsymbol{W}_k^\top \boldsymbol{Z}_0^\top) \boldsymbol{Z}_0 \boldsymbol{W}_v \boldsymbol{v}, && \text{(D.1c)} \\
g_{\text{ssm}}(\boldsymbol{Z}) &= \left((\boldsymbol{z}^\top \boldsymbol{W}_q)^\top \odot ((\boldsymbol{Z}_0 \boldsymbol{W}_k \odot \boldsymbol{Z}_0 \boldsymbol{W}_v) * \boldsymbol{f})_{n+1} \right) \boldsymbol{v}, && \text{(D.1d)}
\end{aligned}
$$

we have objectives

$$
\begin{aligned}
\min_{\boldsymbol{W}} \mathcal{L}_{\text{pgd}}(\mathcal{W}) \quad \text{where} \quad & \mathcal{L}_{\text{pgd}}(\mathcal{W}) = \mathbb{E}\left[(y - g_{\text{pgd}}(\boldsymbol{Z}))^2 \right], && \text{(D.2a)} \\
\min_{\boldsymbol{W}, \boldsymbol{\omega}} \mathcal{L}_{\text{wpgd}}(\mathcal{W}) \quad \text{where} \quad & \mathcal{L}_{\text{wpgd}}(\mathcal{W}) = \mathbb{E}\left[(y - g_{\text{wpgd}}(\boldsymbol{Z}))^2 \right], && \text{(D.2b)} \\
\min_{\boldsymbol{W}_k, \boldsymbol{W}_q, \boldsymbol{W}_v, \boldsymbol{v}} \mathcal{L}_{\text{att}}(\mathcal{W}) \quad \text{where} \quad & \mathcal{L}_{\text{att}}(\mathcal{W}) = \mathbb{E}\left[(y - g_{\text{att}}(\boldsymbol{Z}))^2 \right], && \text{(D.2c)} \\
\min_{\boldsymbol{W}_k, \boldsymbol{W}_q, \boldsymbol{W}_v, \boldsymbol{v}, \boldsymbol{f}} \mathcal{L}_{\text{ssm}}(\mathcal{W}) \quad \text{where} \quad & \mathcal{L}_{\text{ssm}}(\mathcal{W}) = \mathbb{E}\left[(y - g_{\text{ssm}}(\boldsymbol{Z}))^2 \right]. && \text{(D.2d)}
\end{aligned}
$$

Here, the expectation is over the randomness in $(\boldsymbol{x}_i, \xi_i)_{i=1}^n$ and $\boldsymbol{\beta}$, and the search space for $\boldsymbol{W}$ is $\mathbb{R}^{d\times d}$, for $\boldsymbol{\omega}$ is $\mathbb{R}^n$, for $\boldsymbol{W}_k, \boldsymbol{W}_q, \boldsymbol{W}_v$ is $\mathbb{R}^{(d+1)\times(d+1)}$, for $\boldsymbol{v}$ is $\mathbb{R}^{d+1}$, and for $\boldsymbol{f}$ is $\mathbb{R}^{n+1}$.

D.1.1 Proof of Proposition 5.1

Consider the problem setting as discussed in Section 5.2, Proposition 5.1 can be proven by the following two lemmas.

Lemma D.1 *Suppose Assumptions 5.1 and 5.2 hold. Then, given the objectives* (D.2a) *and* (D.2c)*, we have*

$$\min_{\boldsymbol{W}_q, \boldsymbol{W}_k, \boldsymbol{W}_v, \boldsymbol{v}} \mathcal{L}_{att}(\mathcal{W}) = \min_{\boldsymbol{W}} \mathcal{L}_{pgd}(\mathcal{W}).$$

Proof. Recap the linear attention estimator from (D.1c) and denote

$$\boldsymbol{W}_q \boldsymbol{W}_k^\top = \begin{bmatrix} \bar{\boldsymbol{W}} & \boldsymbol{w}_1 \\ \boldsymbol{w}_2^\top & w \end{bmatrix} \qquad \text{and} \qquad \boldsymbol{W}_v \boldsymbol{v} = \begin{bmatrix} \boldsymbol{v}_1 \\ v \end{bmatrix},$$

where $\bar{\boldsymbol{W}} \in \mathbb{R}^{d\times d}$, $\boldsymbol{w}_1, \boldsymbol{w}_2, \boldsymbol{v}_1 \in \mathbb{R}^d$, and $w, v \in \mathbb{R}$. Then we have

$$\begin{aligned}
g_{\text{att}}(\boldsymbol{Z}) &= (\boldsymbol{z}^\top \boldsymbol{W}_q \boldsymbol{W}_k^\top \boldsymbol{Z}_0^\top) \boldsymbol{Z}_0 \boldsymbol{W}_v \boldsymbol{v} \\
&= [\boldsymbol{x}^\top \ 0] \begin{bmatrix} \bar{\boldsymbol{W}} & \boldsymbol{w}_1 \\ \boldsymbol{w}_2^\top & w \end{bmatrix} \begin{bmatrix} \boldsymbol{X}^\top & \boldsymbol{0}_d \\ \boldsymbol{y}^\top & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{X} & \boldsymbol{y} \\ \boldsymbol{0}_d^\top & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{v}_1 \\ v \end{bmatrix} \\
&= (\boldsymbol{x}^\top \bar{\boldsymbol{W}} \boldsymbol{X}^\top + \boldsymbol{x}^\top \boldsymbol{w}_1 \boldsymbol{y}^\top)(\boldsymbol{X}\boldsymbol{v}_1 + \boldsymbol{y} v) \\
&= \boldsymbol{x}^\top (v\bar{\boldsymbol{W}}) \boldsymbol{X}^\top \boldsymbol{y} + \boldsymbol{x}^\top \boldsymbol{w}_1 \boldsymbol{y}^\top \boldsymbol{X} \boldsymbol{v}_1 + \boldsymbol{x}^\top \left(\bar{\boldsymbol{W}} \boldsymbol{X}^\top \boldsymbol{X} \boldsymbol{v}_1 + v \|\boldsymbol{y}\|^2 \boldsymbol{w}_1 \right) \\
&= \boldsymbol{x}^\top (v\bar{\boldsymbol{W}} + \boldsymbol{w}_1 \boldsymbol{v}_1^\top) \boldsymbol{X}^\top \boldsymbol{y} + \boldsymbol{x}^\top \left(\bar{\boldsymbol{W}} \boldsymbol{X}^\top \boldsymbol{X} \boldsymbol{v}_1 + v \|\boldsymbol{y}\|^2 \boldsymbol{w}_1 \right) \\
&= \underbrace{\boldsymbol{x}^\top \tilde{\boldsymbol{W}} \boldsymbol{X}^\top \boldsymbol{y}}_{\tilde{g}_{\text{att}}(\boldsymbol{Z})} + \underbrace{\boldsymbol{x}^\top \left(\bar{\boldsymbol{W}} \boldsymbol{X}^\top \boldsymbol{X} \boldsymbol{v}_1 + v \|\boldsymbol{y}\|^2 \boldsymbol{w}_1 \right)}_{\epsilon},
\end{aligned} \tag{D.3}$$

where $\tilde{\boldsymbol{W}} := v\bar{\boldsymbol{W}} + \boldsymbol{w}_1 \boldsymbol{v}_1^\top$.

We first show that for any given parameters $\boldsymbol{W}_k, \boldsymbol{W}_q, \boldsymbol{W}_v, \boldsymbol{v}$,

$$\mathbb{E}\left[(g_{\text{att}}(\boldsymbol{Z}) - y)^2 \right] \geq \mathbb{E}\left[(\tilde{g}_{\text{att}}(\boldsymbol{Z}) - y)^2 \right]. \tag{D.4}$$

To this goal, we have

$$\begin{aligned}
\mathbb{E}\left[(g_{\text{att}}(\boldsymbol{Z}) - y)^2 \right] - \mathbb{E}\left[(\tilde{g}_{\text{att}}(\boldsymbol{Z}) - y)^2 \right] &= \mathbb{E}\left[(\tilde{g}_{\text{att}}(\boldsymbol{Z}) + \epsilon - y)^2 \right] - \mathbb{E}\left[(\tilde{g}_{\text{att}}(\boldsymbol{Z}) - y)^2 \right] \\
&= \mathbb{E}[\epsilon^2] + 2\,\mathbb{E}[(\tilde{g}_{\text{att}}(\boldsymbol{Z}) - y)\epsilon]
\end{aligned} \tag{D.5}$$

where we have decomposition

$$
\begin{aligned}
(\tilde{g}_{\text{att}}(\boldsymbol{Z}) - y)\epsilon &= (\boldsymbol{x}^\top \tilde{\boldsymbol{W}} \boldsymbol{X}^\top \boldsymbol{y} - y)\boldsymbol{x}^\top \left(\bar{\boldsymbol{W}} \boldsymbol{X}^\top \boldsymbol{X} \boldsymbol{v}_1 + v\|\boldsymbol{y}\|^2 \boldsymbol{w}_1\right) \\
&= \boldsymbol{y}^\top \boldsymbol{X} \tilde{\boldsymbol{W}}^\top \boldsymbol{x}\boldsymbol{x}^\top \left(\bar{\boldsymbol{W}} \boldsymbol{X}^\top \boldsymbol{X} \boldsymbol{v}_1 + v\|\boldsymbol{y}\|^2 \boldsymbol{w}_1\right) - y\boldsymbol{x}^\top \left(\bar{\boldsymbol{W}} \boldsymbol{X}^\top \boldsymbol{X} \boldsymbol{v}_1 + v\|\boldsymbol{y}\|^2 \boldsymbol{w}_1\right) \\
&= \underbrace{\boldsymbol{y}^\top \boldsymbol{X} \tilde{\boldsymbol{W}}^\top \boldsymbol{x}\boldsymbol{x}^\top \bar{\boldsymbol{W}} \boldsymbol{X}^\top \boldsymbol{X} \boldsymbol{v}_1}_{(a)} + \underbrace{v\|\boldsymbol{y}\|^2 \boldsymbol{y}^\top \boldsymbol{X} \tilde{\boldsymbol{W}}^\top \boldsymbol{x}\boldsymbol{x}^\top \boldsymbol{w}_1}_{(b)} \\
&\quad - \underbrace{y\boldsymbol{x}^\top \bar{\boldsymbol{W}} \boldsymbol{X}^\top \boldsymbol{X} \boldsymbol{v}_1}_{(c)} - \underbrace{vy\|\boldsymbol{y}\|^2 \boldsymbol{x}^\top \boldsymbol{w}_1}_{(d)}.
\end{aligned}
$$

In the following, we consider the expectations of $(a), (b), (c), (d)$ sequentially, which return zeros under Assumptions 5.1 and 5.2. Note that since Assumption 5.1 holds, expectation of any odd *order* of monomial of the entries of $\boldsymbol{X}, \boldsymbol{x}, \boldsymbol{\beta}$ returns zero, i.e., order of $\boldsymbol{x}^\top \boldsymbol{\beta} \boldsymbol{x}$ is 3 and therefore $\mathbb{E}[\boldsymbol{x}^\top \boldsymbol{\beta} \boldsymbol{x}] = \boldsymbol{0}_d$.

$$
\begin{aligned}
(a): \quad & \mathbb{E}\left[\boldsymbol{y}^\top \boldsymbol{X} \tilde{\boldsymbol{W}}^\top \boldsymbol{x}\boldsymbol{x}^\top \bar{\boldsymbol{W}} \boldsymbol{X}^\top \boldsymbol{X} \boldsymbol{v}_1\right] \\
&= \mathbb{E}\left[(\boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\xi})^\top \boldsymbol{X} \tilde{\boldsymbol{W}}^\top \boldsymbol{x}\boldsymbol{x}^\top \bar{\boldsymbol{W}} \boldsymbol{X}^\top \boldsymbol{X} \boldsymbol{v}_1\right] \\
&= \mathbb{E}\left[\boldsymbol{\beta}^\top \boldsymbol{X}^\top \boldsymbol{X} \tilde{\boldsymbol{W}}^\top \boldsymbol{x}\boldsymbol{x}^\top \bar{\boldsymbol{W}} \boldsymbol{X}^\top \boldsymbol{X} \boldsymbol{v}_1\right] + \mathbb{E}\left[\boldsymbol{\xi}^\top \boldsymbol{X} \tilde{\boldsymbol{W}}^\top \boldsymbol{x}\boldsymbol{x}^\top \bar{\boldsymbol{W}} \boldsymbol{X}^\top \boldsymbol{X} \boldsymbol{v}_1\right] \\
&= 0.
\end{aligned}
$$

$$
\begin{aligned}
(b): \quad & \mathbb{E}\left[v\|\boldsymbol{y}\|^2 \boldsymbol{y}^\top \boldsymbol{X} \tilde{\boldsymbol{W}}^\top \boldsymbol{x}\boldsymbol{x}^\top \boldsymbol{w}_1\right] \\
&= \mathbb{E}\left[v(\boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\xi})^\top (\boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\xi})(\boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\xi})^\top \boldsymbol{X} \tilde{\boldsymbol{W}}^\top \boldsymbol{x}\boldsymbol{x}^\top \boldsymbol{w}_1\right] \\
&= \mathbb{E}\left[v\|\boldsymbol{\xi}\|^2 \boldsymbol{\xi}^\top \boldsymbol{X} \tilde{\boldsymbol{W}}^\top \boldsymbol{x}\boldsymbol{x}^\top \boldsymbol{w}_1\right] \\
&= 0.
\end{aligned}
$$

$$
\begin{aligned}
(c): \quad & \mathbb{E}\left[y\boldsymbol{x}^\top \bar{\boldsymbol{W}} \boldsymbol{X}^\top \boldsymbol{X} \boldsymbol{v}_1\right] \\
&= \mathbb{E}\left[(\boldsymbol{x}^\top \boldsymbol{\beta} + \xi)\boldsymbol{x}^\top \bar{\boldsymbol{W}} \boldsymbol{X}^\top \boldsymbol{X} \boldsymbol{v}_1\right] \\
&= \mathbb{E}\left[\boldsymbol{\beta}^\top \boldsymbol{x}\boldsymbol{x}^\top \bar{\boldsymbol{W}} \boldsymbol{X}^\top \boldsymbol{X} \boldsymbol{v}_1\right] + \mathbb{E}\left[\xi \boldsymbol{x}^\top \bar{\boldsymbol{W}} \boldsymbol{X}^\top \boldsymbol{X} \boldsymbol{v}_1\right] \\
&= 0.
\end{aligned}
$$

$$
\begin{aligned}
(d): \quad & \mathbb{E}\left[vy\|\boldsymbol{y}\|^2\boldsymbol{x}^\top\boldsymbol{w}_1\right] \\
& = v\,\mathbb{E}\left[(\boldsymbol{\beta}^\top\boldsymbol{x}+\xi)(\boldsymbol{X}\boldsymbol{\beta}+\boldsymbol{\xi})^\top(\boldsymbol{X}\boldsymbol{\beta}+\boldsymbol{\xi})\boldsymbol{x}^\top\boldsymbol{w}_1\right] \\
& = v\,\mathbb{E}\left[\xi\|\boldsymbol{\xi}\|^2\boldsymbol{x}^\top\boldsymbol{w}_1\right] \\
& = 0.
\end{aligned}
$$

Combining the results with (D.5) returns that

$$
\mathbb{E}\left[(g_{\text{att}}(\boldsymbol{Z})-y)^2\right] - \mathbb{E}\left[(\tilde{g}_{\text{att}}(\boldsymbol{Z})-y)^2\right] = \mathbb{E}[\epsilon^2] \geq 0 \tag{D.6}
$$

which completes the proof of (D.4). Therefore, we obtain

$$
\min_{\boldsymbol{W}_q,\boldsymbol{W}_k,\boldsymbol{W}_v,\boldsymbol{v}} \mathbb{E}\left[(g_{\text{att}}(\boldsymbol{Z})-y)^2\right] \geq \min_{\tilde{\boldsymbol{W}}} \mathbb{E}\left[(\tilde{g}_{\text{att}}(\boldsymbol{Z})-y)^2\right] = \min_{\boldsymbol{W}} \mathbb{E}\left[(g_{\text{pgd}}(\boldsymbol{Z})-y)^2\right].
$$

We conclude the proof of this lemma by showing that for any $\boldsymbol{W} \in \mathbb{R}^{d\times d}$ in g_{pgd}, there exist $\boldsymbol{W}_k, \boldsymbol{W}_q, \boldsymbol{W}_v, \boldsymbol{v}$ such that $g_{\text{att}}(\boldsymbol{Z}) = g_{\text{pgd}}(\boldsymbol{Z})$. Let

$$
\boldsymbol{W}_k = \boldsymbol{W}_v = \boldsymbol{I}_{d+1}, \qquad \boldsymbol{W}_q = \begin{bmatrix} \boldsymbol{W} & \boldsymbol{0}_d \\ \boldsymbol{0}_d^\top & 0 \end{bmatrix}, \quad \text{and} \quad \boldsymbol{v} = \begin{bmatrix} \boldsymbol{0}_d \\ 1 \end{bmatrix}.
$$

Then we obtain

$$
g_{\text{att}}(\boldsymbol{Z}) = \boldsymbol{x}^\top\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{y} = g_{\text{pgd}}(\boldsymbol{Z}), \tag{D.7}
$$

which completes the proof. ■

Lemma D.2 *Suppose Assumptions 5.1 and 5.2 hold. Then, given the objectives in* (D.2)*, we have*

$$
\min_{\boldsymbol{W}_q,\boldsymbol{W}_k,\boldsymbol{W}_v,\boldsymbol{v},\boldsymbol{f}} \mathcal{L}_{ssm}(\mathcal{W}) = \min_{\boldsymbol{W},\boldsymbol{\omega}} \mathcal{L}_{wpgd}(\mathcal{W}). \tag{D.8}
$$

Additionally, if the examples $(\boldsymbol{x}_i, y_i)_{i=1}^n$ *follow the same distribution and are conditionally independent given* $\boldsymbol{x}$ *and* $\boldsymbol{\beta}$*, then SSM/H3 can achieve the optimal loss using the all-ones filter and*

$$
\min_{\boldsymbol{W},\boldsymbol{\omega}} \mathcal{L}_{wpgd}(\mathcal{W}) = \min_{\boldsymbol{W}} \mathcal{L}_{pgd}(\mathcal{W}). \tag{D.9}
$$

Proof. Recap the SSM estimator from (D.1d) and let

$$\begin{aligned}\boldsymbol{W}_q &= \begin{bmatrix}\boldsymbol{w}_{q1} & \boldsymbol{w}_{q2} & \cdots & \boldsymbol{w}_{q,d+1}\end{bmatrix},\\ \boldsymbol{W}_k &= \begin{bmatrix}\boldsymbol{w}_{k1} & \boldsymbol{w}_{k2} & \cdots & \boldsymbol{w}_{k,d+1}\end{bmatrix},\\ \boldsymbol{W}_v &= \begin{bmatrix}\boldsymbol{w}_{v1} & \boldsymbol{w}_{v2} & \cdots & \boldsymbol{w}_{v,d+1}\end{bmatrix},\end{aligned}$$

where $\boldsymbol{w}_{qj}, \boldsymbol{w}_{kj}, \boldsymbol{w}_{vj} \in \mathbb{R}^{d+1}$ for $j \leq d+1$, and let

$$\boldsymbol{v} = \begin{bmatrix} v_1 \\ v_2 \\ \cdots \\ v_{d+1}\end{bmatrix}, \quad \text{and} \quad \boldsymbol{f} = \begin{bmatrix} f_0 \\ f_1 \\ \cdots \\ f_n\end{bmatrix}.$$

Then we have

$$\begin{aligned} g_{\mathrm{ssm}}(\boldsymbol{Z}) &= \left((\boldsymbol{z}^\top \boldsymbol{W}_q)^\top \odot ((\boldsymbol{Z}_0 \boldsymbol{W}_k \odot \boldsymbol{Z}_0 \boldsymbol{W}_v) * \boldsymbol{f})_{n+1}\right) \boldsymbol{v} \\ &= \sum_{i=1}^{n} f_{n+1-i} \cdot \boldsymbol{v}^\top \left(\begin{bmatrix} \boldsymbol{w}_{q1}^\top \boldsymbol{z} \\ \cdots \\ \boldsymbol{w}_{q,d+1}^\top \boldsymbol{z}\end{bmatrix} \odot \begin{bmatrix} \boldsymbol{w}_{k1}^\top \boldsymbol{z}_i \boldsymbol{w}_{v1}^\top \boldsymbol{z}_i \\ \cdots \\ \boldsymbol{w}_{k,d+1}^\top \boldsymbol{z}_i \boldsymbol{w}_{v,d+1}^\top \boldsymbol{z}_i\end{bmatrix} \right) \\ &= \sum_{i=1}^{n} f_{n+1-i} \cdot \boldsymbol{v}^\top \begin{bmatrix} \boldsymbol{w}_{q1}^\top \boldsymbol{z} \boldsymbol{w}_{k1}^\top \boldsymbol{z}_i \boldsymbol{w}_{v1}^\top \boldsymbol{z}_i \\ \cdots \\ \boldsymbol{w}_{q,d+1}^\top \boldsymbol{z} \boldsymbol{w}_{k,d+1}^\top \boldsymbol{z}_i \boldsymbol{w}_{v,d+1}^\top \boldsymbol{z}_i\end{bmatrix}. \end{aligned}$$

Next for all $j \leq d+1$, let

$$\boldsymbol{w}_{qj} = \begin{bmatrix} \bar{\boldsymbol{w}}_{qj} \\ w_{qj}\end{bmatrix}, \quad \boldsymbol{w}_{kj} = \begin{bmatrix} \bar{\boldsymbol{w}}_{kj} \\ w_{kj}\end{bmatrix}, \quad \boldsymbol{w}_{vj} = \begin{bmatrix} \bar{\boldsymbol{w}}_{vj} \\ w_{vj}\end{bmatrix}$$

where $\bar{\boldsymbol{w}}_{qj}, \bar{\boldsymbol{w}}_{kj}, \bar{\boldsymbol{w}}_{vj} \in \mathbb{R}^d$ and $w_{qj}, w_{kj}, w_{vj} \in \mathbb{R}$. Then we have

$$\begin{aligned} \boldsymbol{w}_{qj}^\top \boldsymbol{z} \boldsymbol{w}_{kj}^\top \boldsymbol{z}_i \boldsymbol{w}_{vj}^\top \boldsymbol{z}_i &= \left(\bar{\boldsymbol{w}}_{qj}^\top \boldsymbol{x}\right) \left(\bar{\boldsymbol{w}}_{kj}^\top \boldsymbol{x}_i + w_{kj} y_i\right) \left(\bar{\boldsymbol{w}}_{vj}^\top \boldsymbol{x}_i + w_{vj} y_i\right) \\ &= \boldsymbol{x}^\top \bar{\boldsymbol{w}}_{qj} \left(w_{vj} \bar{\boldsymbol{w}}_{kj}^\top + w_{kj} \bar{\boldsymbol{w}}_{vj}^\top\right) \boldsymbol{x}_i y_i + \left(\bar{\boldsymbol{w}}_{qj}^\top \boldsymbol{x}\right) \left(\bar{\boldsymbol{w}}_{kj}^\top \boldsymbol{x}_i\right) \left(\bar{\boldsymbol{w}}_{vj}^\top \boldsymbol{x}_i\right) + \left(w_{kj} w_{vj} \bar{\boldsymbol{w}}_{qj}^\top \boldsymbol{x} y_i^2\right) \\ &= \boldsymbol{x}^\top \boldsymbol{W}_j' \boldsymbol{x}_i y_i + \delta_j(\boldsymbol{x}, \boldsymbol{x}_i, \boldsymbol{x}_i) + \boldsymbol{w}_j'^\top \boldsymbol{x} y_i^2 \end{aligned}$$

where

$$\begin{aligned}
&\boldsymbol{W}_j' := \bar{\boldsymbol{w}}_{qj}\left(w_{vj}\bar{\boldsymbol{w}}_{kj}^\top + w_{kj}\bar{\boldsymbol{w}}_{vj}^\top\right) \in \mathbb{R}^{d\times d},\\
&\boldsymbol{w}_j' := w_{kj}w_{vj}\bar{\boldsymbol{w}}_{qj} \in \mathbb{R}^d,\\
&\delta_j(\boldsymbol{x},\boldsymbol{x}_i,\boldsymbol{x}_i) := \left(\bar{\boldsymbol{w}}_{qj}^\top\boldsymbol{x}\right)\left(\bar{\boldsymbol{w}}_{kj}^\top\boldsymbol{x}_i\right)\left(\bar{\boldsymbol{w}}_{vj}^\top\boldsymbol{x}_i\right) \in \mathbb{R}.
\end{aligned}$$

Then

$$\begin{aligned}
g_{\text{ssm}}(\boldsymbol{Z}) &= \sum_{i=1}^{n} f_{n+1-i}\cdot\sum_{j=1}^{d+1} v_j\left(\boldsymbol{x}^\top\boldsymbol{W}_j'\boldsymbol{x}_i y_i + \delta_j(\boldsymbol{x},\boldsymbol{x}_i,\boldsymbol{x}_i) + \boldsymbol{w}_j'^\top\boldsymbol{x}y_i^2\right)\\
&= \boldsymbol{x}^\top\left(\sum_{j=1}^{d+1} v_j\boldsymbol{W}_j'\right)\boldsymbol{X}(\boldsymbol{y}\odot\tilde{\boldsymbol{f}}) + \sum_{i=1}^{n} f_{n+1-i}\cdot\sum_{j=1}^{d+1} v_j\cdot\delta_j(\boldsymbol{x},\boldsymbol{x}_i,\boldsymbol{x}_i)\\
&\quad + \left(\sum_{j=1}^{d+1} v_j\boldsymbol{w}_j'^\top\right)\boldsymbol{x}\boldsymbol{y}^\top(\boldsymbol{y}\odot\tilde{\boldsymbol{f}})\\
&= \underbrace{\boldsymbol{x}^\top\tilde{\boldsymbol{W}}\boldsymbol{X}\tilde{\boldsymbol{y}}}_{\tilde{g}_{\text{ssm}}(\boldsymbol{Z})} + \underbrace{\tilde{\delta}(\boldsymbol{x},\boldsymbol{X},\boldsymbol{X})}_{\epsilon_1} + \underbrace{\tilde{\boldsymbol{w}}^\top\boldsymbol{x}\boldsymbol{y}^\top\tilde{\boldsymbol{y}}}_{\epsilon_2}.
\end{aligned}$$

where

$$\begin{aligned}
&\tilde{\boldsymbol{f}} := [f_n\ \cdots\ f_1]^\top \in \mathbb{R}^n,\\
&\tilde{\boldsymbol{y}} := \boldsymbol{y}\odot\tilde{\boldsymbol{f}} \in \mathbb{R}^n,\\
&\tilde{\boldsymbol{W}} := \sum_{j=1}^{d+1} v_j\boldsymbol{W}_j' \in \mathbb{R}^{d\times d},\\
&\tilde{\boldsymbol{w}} := \sum_{j=1}^{d+1} v_j\boldsymbol{w}_j' \in \mathbb{R}^d,\\
&\tilde{\delta}(\boldsymbol{x},\boldsymbol{X},\boldsymbol{X}) := \sum_{i=1}^{n} f_{n+1-i}\cdot\sum_{j=1}^{d+1} v_j\cdot\delta_j(\boldsymbol{x},\boldsymbol{x}_i,\boldsymbol{x}_i) \in \mathbb{R}.
\end{aligned}$$

Next we will show that for any $\boldsymbol{W}_k, \boldsymbol{W}_q, \boldsymbol{W}_v, \boldsymbol{v}$,

$$\mathbb{E}\left[(g_{\text{ssm}}(\boldsymbol{Z}) - y)^2\right] \geq \mathbb{E}\left[(\tilde{g}_{\text{ssm}}(\boldsymbol{Z}) - y)^2\right].$$

To start with, we obtain

$$\begin{aligned}\mathbb{E}\left[(g_{\text{ssm}}(\boldsymbol{Z})-y)^2\right] &= \mathbb{E}\left[(\tilde{g}_{\text{ssm}}(\boldsymbol{Z})+\epsilon_1+\epsilon_2-y)^2\right] \\ &= \mathbb{E}\left[(\tilde{g}_{\text{ssm}}(\boldsymbol{Z})-y)^2\right]+\mathbb{E}\left[(\epsilon_1+\epsilon_2)^2\right]+2\,\mathbb{E}\left[(\tilde{g}_{\text{ssm}}(\boldsymbol{Z})-y)(\epsilon_1+\epsilon_2)\right]\end{aligned} \tag{D.10}$$

where there is decomposition

$$\begin{aligned}&(\tilde{g}_{\text{ssm}}(\boldsymbol{Z})-y)(\epsilon_1+\epsilon_2) \\ &= \underbrace{\tilde{\delta}(\boldsymbol{x},\boldsymbol{X},\boldsymbol{X})\cdot\boldsymbol{x}^\top\tilde{\boldsymbol{W}}\boldsymbol{X}\tilde{\boldsymbol{y}}}_{(a)}-\underbrace{\tilde{\delta}(\boldsymbol{x},\boldsymbol{X},\boldsymbol{X})y}_{(b)}+\underbrace{\tilde{\boldsymbol{w}}^\top\boldsymbol{x}\boldsymbol{y}^\top\tilde{\boldsymbol{y}}\cdot\boldsymbol{x}^\top\tilde{\boldsymbol{W}}\boldsymbol{X}\tilde{\boldsymbol{y}}}_{(c)}-\underbrace{y\cdot\tilde{\boldsymbol{w}}^\top\boldsymbol{x}\boldsymbol{y}^\top\tilde{\boldsymbol{y}}}_{(d)}.\end{aligned}$$

In the following, similar to the proof of Lemma D.1, we consider the expectations of $(a),(b),(c),(d)$ sequentially, which return zeros under Assumptions 5.1 and 5.2. Note that $\delta_j(\boldsymbol{x},\boldsymbol{x}_i,\boldsymbol{x}_i)$'s and $\tilde{\delta}(\boldsymbol{x},\boldsymbol{X},\boldsymbol{X})$ are summation of monomials of entries of $(\boldsymbol{x},\boldsymbol{X},\boldsymbol{\beta})$ with order 3, and entries of $\boldsymbol{y}$ and y are summation of monomials of entries of $(\boldsymbol{x},\boldsymbol{X},\boldsymbol{\beta})$ with even orders: e.g., $y=\boldsymbol{x}^\top\boldsymbol{\beta}+\xi$ where ξ is of oder 0 and $\boldsymbol{x}^\top\boldsymbol{\beta}$ is of order 2.

$$\begin{aligned}(a):\quad &\mathbb{E}\left[\tilde{\delta}(\boldsymbol{x},\boldsymbol{X},\boldsymbol{X})\cdot\boldsymbol{x}^\top\tilde{\boldsymbol{W}}\boldsymbol{X}\tilde{\boldsymbol{y}}\right] \\ &= \mathbb{E}\left[\tilde{\delta}(\boldsymbol{x},\boldsymbol{X},\boldsymbol{X})\cdot\boldsymbol{x}^\top\tilde{\boldsymbol{W}}\boldsymbol{X}(\boldsymbol{X}\boldsymbol{\beta}\odot\tilde{\boldsymbol{f}})\right]+\mathbb{E}\left[\tilde{\delta}(\boldsymbol{x},\boldsymbol{X},\boldsymbol{X})\cdot\boldsymbol{x}^\top\tilde{\boldsymbol{W}}\boldsymbol{X}(\boldsymbol{\xi}\odot\tilde{\boldsymbol{f}})\right] \\ &= \mathbb{E}\left[\tilde{\delta}(\boldsymbol{x},\boldsymbol{X},\boldsymbol{X})\cdot\boldsymbol{x}^\top\tilde{\boldsymbol{W}}\boldsymbol{X}\right]\mathbb{E}\left[\boldsymbol{\xi}\odot\tilde{\boldsymbol{f}}\right] \\ &= 0.\end{aligned}$$

$$\begin{aligned}(b):\quad &\mathbb{E}\left[\tilde{\delta}(\boldsymbol{x},\boldsymbol{X},\boldsymbol{X})y\right] \\ &= \mathbb{E}\left[\tilde{\delta}(\boldsymbol{x},\boldsymbol{X},\boldsymbol{X})(\boldsymbol{x}^\top\boldsymbol{\beta}+\xi)\right] \\ &= \mathbb{E}\left[\tilde{\delta}(\boldsymbol{x},\boldsymbol{X},\boldsymbol{X})\boldsymbol{x}^\top\boldsymbol{\beta}\right]+\mathbb{E}\left[\tilde{\delta}(\boldsymbol{x},\boldsymbol{X},\boldsymbol{X})\xi\right] \\ &= 0.\end{aligned}$$

$$\begin{aligned}(c):\quad &\mathbb{E}\left[\tilde{\boldsymbol{w}}^\top\boldsymbol{x}\boldsymbol{y}^\top\tilde{\boldsymbol{y}}\cdot\boldsymbol{x}^\top\tilde{\boldsymbol{W}}\boldsymbol{X}\tilde{\boldsymbol{y}}\right] \\ &= \mathbb{E}\left[\tilde{\boldsymbol{w}}^\top\boldsymbol{x}(\boldsymbol{X}\boldsymbol{\beta}+\boldsymbol{\xi})^\top(\boldsymbol{X}\boldsymbol{\beta}\odot\tilde{\boldsymbol{f}}+\boldsymbol{\xi}\odot\tilde{\boldsymbol{f}})\cdot\boldsymbol{x}^\top\tilde{\boldsymbol{W}}\boldsymbol{X}(\boldsymbol{X}\boldsymbol{\beta}\odot\tilde{\boldsymbol{f}}+\boldsymbol{\xi}\odot\tilde{\boldsymbol{f}})\right] \\ &= 0.\end{aligned}$$

$$
\begin{aligned}
(d): \quad & \mathbb{E}\left[y \cdot \tilde{\boldsymbol{w}}^\top \boldsymbol{x}\boldsymbol{y}^\top \tilde{\boldsymbol{y}}\right] \\
& = \mathbb{E}\left[(\boldsymbol{x}^\top \boldsymbol{\beta} + \xi) \cdot \tilde{\boldsymbol{w}}^\top \boldsymbol{x}(\boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\xi})^\top (\boldsymbol{X}\boldsymbol{\beta} \odot \tilde{\boldsymbol{f}} + \boldsymbol{\xi} \odot \tilde{\boldsymbol{f}})\right] \\
& = 0.
\end{aligned}
$$

Combining the results with (D.10) results that

$$
\mathbb{E}\left[(g_{\mathrm{ssm}}(\boldsymbol{Z}) - y)^2\right] - \mathbb{E}\left[(\tilde{g}_{\mathrm{ssm}}(\boldsymbol{Z}) - y)^2\right] = \mathbb{E}\left[(\epsilon_1 + \epsilon_2)^2\right] \geq 0.
$$

Therefore we obtain,

$$
\min_{\boldsymbol{W}_q, \boldsymbol{W}_k, \boldsymbol{W}_v, \boldsymbol{v}, \boldsymbol{f}} \mathbb{E}\left[(g_{\mathrm{ssm}}(\boldsymbol{Z}) - y)^2\right] \geq \min_{\tilde{\boldsymbol{W}}, \tilde{\boldsymbol{f}}} \mathbb{E}\left[(\tilde{g}_{\mathrm{ssm}}(\boldsymbol{Z}) - y)^2\right] = \min_{\boldsymbol{W}, \boldsymbol{\omega}} \mathbb{E}\left[(g_{\mathrm{wpgd}}(\boldsymbol{Z}) - y)^2\right].
$$

Next we show that for any choices of $\boldsymbol{W}$ and $\boldsymbol{\omega}$ in g_{wpgd}, there are $\boldsymbol{W}_{q,k,v}, \boldsymbol{v}, \boldsymbol{f}$ such that $g_{\mathrm{ssm}} \equiv g_{\mathrm{wpgd}}$. To this end, given $\boldsymbol{\omega} = [\omega_1 \ \ldots \ \omega_n]^\top$, let

$$
\boldsymbol{W}_q = \boldsymbol{I}_{d+1}, \quad \boldsymbol{W}_k = \begin{bmatrix} \boldsymbol{W}^\top & \boldsymbol{0}_d \\ \boldsymbol{0}_d^\top & 0 \end{bmatrix}, \quad \boldsymbol{W}_v = \begin{bmatrix} \boldsymbol{0}_{d\times d} & \boldsymbol{0}_d \\ \boldsymbol{1}_d^\top & 0 \end{bmatrix}, \quad \boldsymbol{v} = \begin{bmatrix} \boldsymbol{1}_d \\ 0 \end{bmatrix} \quad \text{and} \quad \boldsymbol{f} = \begin{bmatrix} 0 \\ \omega_n \\ \cdots \\ \omega_1 \end{bmatrix}.
$$

Then we get

$$
\begin{aligned}
((\boldsymbol{Z}_0\boldsymbol{W}_k \odot \boldsymbol{Z}_0\boldsymbol{W}_v) * \boldsymbol{f})_{n+1} &= \left(\left(\begin{bmatrix} \boldsymbol{X}\boldsymbol{W}^\top & \boldsymbol{0}_n \\ \boldsymbol{0}_d & 0 \end{bmatrix} \odot \begin{bmatrix} \boldsymbol{y}\boldsymbol{1}_d^\top & \boldsymbol{0}_n \\ \boldsymbol{0}_d & 0 \end{bmatrix}\right) * \boldsymbol{f}\right)_{n+1} \\
&= \begin{bmatrix} \sum_{i=1}^n \omega_i \cdot y_i \boldsymbol{W}\boldsymbol{x}_i \\ 0 \end{bmatrix} \\
&= \begin{bmatrix} \boldsymbol{W}\boldsymbol{X}^\top(\boldsymbol{y} \odot \boldsymbol{\omega}) \\ 0 \end{bmatrix},
\end{aligned}
$$

and therefore

$$
g_{\mathrm{ssm}}(\boldsymbol{Z}) = \boldsymbol{x}^\top \boldsymbol{W}\boldsymbol{X}^\top(\boldsymbol{y} \odot \boldsymbol{\omega}) = g_{\mathrm{wpgd}}(\boldsymbol{Z}),
$$

which completes the proof of (D.8).

Next, to show (D.9), for any $\boldsymbol{W} \in \mathbb{R}^{d\times d}$, let $\mathcal{L}(\boldsymbol{\omega}) = \mathbb{E}\left[\left(\boldsymbol{x}^\top \boldsymbol{W}\boldsymbol{X}^\top(\boldsymbol{y} \odot \boldsymbol{\omega}) - y\right)^2\right]$. Then

we have

$$\frac{\partial \mathcal{L}(\boldsymbol{\omega})}{\partial \omega_i} = \mathbb{E}\left[2\left(\boldsymbol{x}^\top \boldsymbol{W}\sum_{j=1}^n \omega_j y_j \boldsymbol{x}_j - y\right)\left(\boldsymbol{x}^\top \boldsymbol{W} y_i \boldsymbol{x}_i\right)\right]$$
$$= 2\sum_{j=1}^n \omega_j \,\mathbb{E}\left[(\boldsymbol{x}^\top \boldsymbol{W} y_j \boldsymbol{x}_j)(\boldsymbol{x}^\top \boldsymbol{W} y_i \boldsymbol{x}_i)\right] - 2\,\mathbb{E}\left[y\boldsymbol{x}^\top \boldsymbol{W} y_i \boldsymbol{x}_i\right].$$

Here since $(\boldsymbol{x}_i, y_i)_{i=1}^n$ follow the same distribution and are conditionally independent given $\boldsymbol{x}$ and $\boldsymbol{\beta}$, for any $i \neq j \neq j'$, $\mathbb{E}\left[(\boldsymbol{x}^\top \boldsymbol{W} y_i \boldsymbol{x}_i)^2\right] = \mathbb{E}\left[(\boldsymbol{x}^\top \boldsymbol{W} y_j \boldsymbol{x}_j)^2\right]$ and $\mathbb{E}\left[(\boldsymbol{x}^\top \boldsymbol{W} y_j \boldsymbol{x}_j)(\boldsymbol{x}^\top \boldsymbol{W} y_i \boldsymbol{x}_i)\right] = \mathbb{E}\left[(\boldsymbol{x}^\top \boldsymbol{W} y_{j'} \boldsymbol{x}_{j'})(\boldsymbol{x}^\top \boldsymbol{W} y_i \boldsymbol{x}_i)\right]$. Then let

$$\mathbb{E}\left[(\boldsymbol{x}^\top \boldsymbol{W} y_j \boldsymbol{x}_j)(\boldsymbol{x}^\top \boldsymbol{W} y_i \boldsymbol{x}_i)\right] = \begin{cases} c_1, & i \neq j \\ c_2, & i = j \end{cases} \quad \text{and} \quad \mathbb{E}\left[y\boldsymbol{x}^\top \boldsymbol{W} y_i \boldsymbol{x}_i\right] = c_3,$$

where $(c_1, c_2, c_3) := (c_1(\boldsymbol{W}), c_2(\boldsymbol{W}), c_3(\boldsymbol{W}))$. We get

$$\frac{\partial \mathcal{L}(\boldsymbol{\omega})}{\partial \omega_i} = 2c_1 \boldsymbol{\omega}^\top \mathbf{1}_n + 2(c_2 - c_1)\omega_i - 2c_3.$$

If $c_2 - c_1 = 0$, then $\frac{\partial \mathcal{L}(\boldsymbol{\omega})}{\partial \omega_i} \equiv 2c_1 \boldsymbol{\omega}^\top \mathbf{1}_n - 2c_3$ for all $i \leq n$ and any $\boldsymbol{\omega} \in \mathbb{R}^n$ achieves the same performance.

If $c_2 - c_1 \neq 0$, setting $\frac{\partial \mathcal{L}(\boldsymbol{\omega})}{\partial \omega_i} = 0$ returns

$$\omega_i = \frac{c_3 - c_1 \sum_{j=1}^n \omega_j}{c_2 - c_1} := C \quad \text{for all } i \leq n.$$

Therefore the optimal loss is achieved via setting $\boldsymbol{\omega} = C\mathbf{1}_n$. Without loss of generality, we can update $\boldsymbol{W} \to C\boldsymbol{W}$. Then $\boldsymbol{\omega} = \mathbf{1}_n$, and we obtain

$$\min_{\boldsymbol{W},\boldsymbol{\omega}} \mathbb{E}\left[\left(\boldsymbol{x}^\top \boldsymbol{W}\boldsymbol{X}^\top(\boldsymbol{y} \odot \boldsymbol{\omega}) - y\right)^2\right] = \min_{\boldsymbol{W}} \mathbb{E}\left[(\boldsymbol{x}^\top \boldsymbol{W}\boldsymbol{X}^\top \boldsymbol{y} - y)^2\right]$$

which completes the proof of (D.9). ■

D.1.2 Proof of Lemma 5.1

Proof. Recap the loss $\mathcal{L}_{\text{pgd}}(\mathcal{W})$ in (D.2a) and prediction $g_{\text{pgd}}(\boldsymbol{Z})$ in (D.1a), we have

$$
\begin{aligned}
&\mathcal{L}_{\text{pgd}}(\mathcal{W}) && \text{(D.11)}\\
&= \mathbb{E}[(y - g_{\text{pgd}}(\boldsymbol{Z}))^2]\\
&= \mathbb{E}\left[\left(\boldsymbol{x}^\top\boldsymbol{\beta} + \xi - \boldsymbol{x}^\top\boldsymbol{W}\boldsymbol{X}^\top(\boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\xi})\right)^2\right]\\
&= \mathbb{E}\left[(\boldsymbol{x}^\top\boldsymbol{\beta} - \boldsymbol{x}^\top\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{X}\boldsymbol{\beta})^2 + 2(\boldsymbol{x}^\top\boldsymbol{\beta} - \boldsymbol{x}^\top\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{X}\boldsymbol{\beta})(\xi - \boldsymbol{x}^\top\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{\xi}) + (\xi - \boldsymbol{x}^\top\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{\xi})^2\right]\\
&= \mathbb{E}\left[(\boldsymbol{x}^\top\boldsymbol{\beta} - \boldsymbol{x}^\top\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{X}\boldsymbol{\beta})^2 + (\xi - \boldsymbol{x}^\top\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{\xi})^2\right] + 2\,\mathbb{E}[(\boldsymbol{x}^\top\boldsymbol{\beta} - \boldsymbol{x}^\top\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{X}\boldsymbol{\beta})(\xi - \boldsymbol{x}^\top\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{\xi})]\\
&= \mathbb{E}\left[(\boldsymbol{x}^\top\boldsymbol{\beta} - \boldsymbol{x}^\top\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{X}\boldsymbol{\beta})^2 + (\xi - \boldsymbol{x}^\top\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{\xi})^2\right] && \text{(D.12)}\\
&= \underbrace{\mathbb{E}\left[(\boldsymbol{x}^\top\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{X}\boldsymbol{\beta})^2 + (\boldsymbol{x}^\top\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{\xi})^2\right]}_{f_1(\boldsymbol{W})} \underbrace{-2\,\mathbb{E}[\boldsymbol{\beta}^\top\boldsymbol{x}\boldsymbol{x}^\top\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{X}\boldsymbol{\beta} + \xi\boldsymbol{x}^\top\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{\xi}]}_{f_2(\boldsymbol{W})} + \underbrace{\mathbb{E}[(\boldsymbol{x}^\top\boldsymbol{\beta})^2 + \xi^2]}_{\text{constant}}
\end{aligned}
$$

where (D.12) follows Assumption 5.2. Since $f_2(\boldsymbol{W})$ is convex, $\mathcal{L}_{\text{pgd}}(\mathcal{W})$ is strongly-convex if and only if $f_1(\boldsymbol{W})$ is strongly-convex, which completes the proof of strong convexity.

Next, (D.6) and (D.7) in the proof of Lemma D.1 demonstrate that the optimal loss is achievable and is achieved at $\epsilon = 0$. Subsequently, (D.3) indicates that $g^\star_{\text{att}}$ has the same form as $g^\star_{\text{pgd}}$. Under the strong convexity assumption, $g^\star_{\text{pgd}}$ is unique, which leads to the conclusion that $g^\star_{\text{pgd}} = g^\star_{\text{att}}$. ∎

D.1.3 Proof of Lemma 5.2

Proof. According to Lemma 5.1, $\mathcal{L}_{\text{pgd}}(\mathcal{W})$ is strongly-convex as long as either $\mathbb{E}[(\boldsymbol{x}^\top\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{X}\boldsymbol{\beta})^2]$ or $\mathbb{E}[(\boldsymbol{x}^\top\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{\xi})^2]$ is strongly-convex. Therefore, in this lemma, the two claims correspond to the strong convexity of $\mathbb{E}[(\boldsymbol{x}^\top\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{\xi})^2]$ and $\mathbb{E}[(\boldsymbol{x}^\top\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{X}\boldsymbol{\beta})^2]$ terms, respectively.

Suppose the decomposition claim holds. Without losing generality, we may assume $(\boldsymbol{x}_1, \boldsymbol{\beta}_1, \boldsymbol{X}_1)$ are zero-mean because we can allocate the mean component to $(\boldsymbol{x}_2, \boldsymbol{\beta}_2, \boldsymbol{X}_2)$ without changing the covariance.

• **Claim 1:** Let $\bar{\boldsymbol{\Sigma}}_{\boldsymbol{x}} = \mathbb{E}[\boldsymbol{x}_1\boldsymbol{x}_1^\top]$, $\bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} = \mathbb{E}[\boldsymbol{\beta}_1\boldsymbol{\beta}_1^\top]$, and $\bar{\boldsymbol{\Sigma}}_{\boldsymbol{X}} = \mathbb{E}[\boldsymbol{X}_1^\top\boldsymbol{X}_1]$. If the first claim holds, using independence, observe that we can write

$$
\mathbb{E}[(\boldsymbol{x}^\top\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{\xi})^2] = \mathbb{E}[(\boldsymbol{x}_1^\top\boldsymbol{W}\boldsymbol{X}_1^\top\boldsymbol{\xi})^2] + \mathbb{E}[(\boldsymbol{x}_1^\top\boldsymbol{W}\boldsymbol{X}_2^\top\boldsymbol{\xi})^2] + \mathbb{E}[(\boldsymbol{x}_2^\top\boldsymbol{W}\boldsymbol{X}_1^\top\boldsymbol{\xi})^2] + \mathbb{E}[(\boldsymbol{x}_2^\top\boldsymbol{W}\boldsymbol{X}_2^\top\boldsymbol{\xi})^2],
$$

where the last three terms of the right hand side are convex and the first term obeys

$$\begin{aligned}\mathbb{E}[(\boldsymbol{x}_1^\top \boldsymbol{W}\boldsymbol{X}_1^\top \boldsymbol{\xi})^2] &= \sigma^2\,\mathbb{E}[\boldsymbol{x}_1^\top \boldsymbol{W}\boldsymbol{X}_1^\top \boldsymbol{X}_1\boldsymbol{W}^\top \boldsymbol{x}_1] \\ &= \sigma^2 \mathtt{tr}\left(\mathbb{E}[\boldsymbol{x}_1\boldsymbol{x}_1^\top \boldsymbol{W}\boldsymbol{X}_1^\top \boldsymbol{X}_1\boldsymbol{W}^\top]\right) \\ &= \sigma^2 \mathtt{tr}\left(\bar{\boldsymbol{\Sigma}}_{\boldsymbol{x}}\boldsymbol{W}\bar{\boldsymbol{\Sigma}}_{\boldsymbol{X}}\boldsymbol{W}^\top\right) \\ &= \sigma^2 \|\sqrt{\bar{\boldsymbol{\Sigma}}_{\boldsymbol{x}}}\boldsymbol{W}\sqrt{\bar{\boldsymbol{\Sigma}}_{\boldsymbol{X}}}\|_F^2.\end{aligned}$$

Since noise level $\sigma > 0$, using the full-rankness of covariance matrices $\bar{\boldsymbol{\Sigma}}_{\boldsymbol{x}}$ and $\bar{\boldsymbol{\Sigma}}_{\boldsymbol{X}}$, we conclude with strong convexity of $\mathbb{E}[(\boldsymbol{x}^\top \boldsymbol{W}\boldsymbol{X}^\top \boldsymbol{\xi})^2]$.

• **Claim 2:** Now recall that $\bar{\boldsymbol{\Sigma}}_{\boldsymbol{X}} = \mathbb{E}[\boldsymbol{X}_1^\top \boldsymbol{X}_1]$ and set $\boldsymbol{A} = \boldsymbol{X}_1^\top \boldsymbol{X}_1 - \bar{\boldsymbol{\Sigma}}_{\boldsymbol{X}}$ and $\boldsymbol{B} = \boldsymbol{X}_2^\top \boldsymbol{X}_2 + \bar{\boldsymbol{\Sigma}}_{\boldsymbol{X}}$. Observe that $\mathbb{E}[\boldsymbol{A}] = 0$. If the second claim holds, $\mathbb{E}[\boldsymbol{X}^\top \boldsymbol{X}] = \mathbb{E}[\boldsymbol{A} + \boldsymbol{B}]$. Note that $(\boldsymbol{A}, \boldsymbol{\beta}_1, \boldsymbol{x}_1)$ are independent of each other and $(\boldsymbol{B}, \boldsymbol{\beta}_2, \boldsymbol{x}_2)$. Using independence and $\mathbb{E}[\boldsymbol{A}] = 0$, similarly write

$$\mathbb{E}[(\boldsymbol{x}^\top \boldsymbol{W}\boldsymbol{X}^\top \boldsymbol{X}\boldsymbol{\beta})^2] = \mathbb{E}[(\boldsymbol{x}^\top \boldsymbol{W}\boldsymbol{A}\boldsymbol{\beta})^2] + \mathbb{E}[(\boldsymbol{x}^\top \boldsymbol{W}\boldsymbol{B}\boldsymbol{\beta})^2].$$

Now using $\mathbb{E}[\boldsymbol{\beta}_1] = \mathbb{E}[\boldsymbol{x}_1] = 0$ and their independence from rest, these terms obeys

$$\begin{aligned}\mathbb{E}[(\boldsymbol{x}^\top \boldsymbol{W}\boldsymbol{A}\boldsymbol{\beta})^2] &= \mathbb{E}[(\boldsymbol{x}_1^\top \boldsymbol{W}\boldsymbol{A}\boldsymbol{\beta}_1)^2] + \mathbb{E}[(\boldsymbol{x}_1^\top \boldsymbol{W}\boldsymbol{A}\boldsymbol{\beta}_2)^2] + \mathbb{E}[(\boldsymbol{x}_2^\top \boldsymbol{W}\boldsymbol{A}\boldsymbol{\beta}_1)^2] + \mathbb{E}[(\boldsymbol{x}_2^\top \boldsymbol{W}\boldsymbol{A}\boldsymbol{\beta}_2)^2] \\ \mathbb{E}[(\boldsymbol{x}^\top \boldsymbol{W}\boldsymbol{B}\boldsymbol{\beta})^2] &= \mathbb{E}[(\boldsymbol{x}_1^\top \boldsymbol{W}\boldsymbol{B}\boldsymbol{\beta}_1)^2] + \mathbb{E}[(\boldsymbol{x}_1^\top \boldsymbol{W}\boldsymbol{B}\boldsymbol{\beta}_2)^2] + \mathbb{E}[(\boldsymbol{x}_2^\top \boldsymbol{W}\boldsymbol{B}\boldsymbol{\beta}_1)^2] + \mathbb{E}[(\boldsymbol{x}_2^\top \boldsymbol{W}\boldsymbol{B}\boldsymbol{\beta}_2)^2].\end{aligned}$$

In both equations, the last three terms of the right hand side are convex. To proceed, we focus on the first terms. Using independence and setting $\boldsymbol{\Sigma}_{\boldsymbol{X}} = \mathbb{E}[\boldsymbol{X}^\top \boldsymbol{X}] \succeq \bar{\boldsymbol{\Sigma}}_{\boldsymbol{X}} \succ 0$, we note that

$$\mathbb{E}[(\boldsymbol{x}_1^\top \boldsymbol{W}\boldsymbol{A}\boldsymbol{\beta}_1)^2] + \mathbb{E}[(\boldsymbol{x}_1^\top \boldsymbol{W}\boldsymbol{B}\boldsymbol{\beta}_1)^2] = \mathbb{E}[(\boldsymbol{x}_1^\top \boldsymbol{W}\boldsymbol{X}^\top \boldsymbol{X}\boldsymbol{\beta}_1)^2]$$

where $\boldsymbol{x}_1, \boldsymbol{\beta}_1, \boldsymbol{X}$ are independent and full-rank covariance. To proceed, note that

$$\mathbb{E}[(\boldsymbol{x}_1^\top \boldsymbol{W}\boldsymbol{X}^\top \boldsymbol{X}\boldsymbol{\beta}_1)^2] = \mathbb{E}[(\boldsymbol{x}_1^\top \boldsymbol{W}\boldsymbol{\Sigma}_{\boldsymbol{X}}\boldsymbol{\beta}_1)^2] + \mathbb{E}[(\boldsymbol{x}_1^\top \boldsymbol{W}(\boldsymbol{X}^\top \boldsymbol{X} - \boldsymbol{\Sigma}_{\boldsymbol{X}})\boldsymbol{\beta}_1)^2].$$

Observing the convexity of the right hand side and focusing on the first term, we get

$$\mathbb{E}[(\boldsymbol{x}_1^\top \boldsymbol{W}\boldsymbol{\Sigma}_{\boldsymbol{X}}\boldsymbol{\beta}_1)^2] = \mathtt{tr}\left(\bar{\boldsymbol{\Sigma}}_{\boldsymbol{x}}\boldsymbol{W}\boldsymbol{\Sigma}_{\boldsymbol{X}}\bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}\boldsymbol{\Sigma}_{\boldsymbol{X}}\boldsymbol{W}^\top\right) = \|\sqrt{\bar{\boldsymbol{\Sigma}}_x}\boldsymbol{W}\boldsymbol{\Sigma}_{\boldsymbol{X}}\sqrt{\bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}}\|_F^2.$$

Using the fact that covariance matrices, $\bar{\boldsymbol{\Sigma}}_{\boldsymbol{x}}, \boldsymbol{\Sigma}_{\boldsymbol{X}}, \bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}$, are full rank concludes the strong convexity proof of $\mathbb{E}[(\boldsymbol{x}^\top \boldsymbol{W}\boldsymbol{X}^\top \boldsymbol{X}\boldsymbol{\beta})^2]$. ∎

D.2 Analysis of General Data Distribution

In this section, we provide the proofs in Section 5.3, which focuses on solving Objective (5.5a). For the sake of clean notation, let $\mathcal{L}(\boldsymbol{W}) := \mathcal{L}_{\text{pgd}}(\mathcal{W})$ and $g := g_{\text{pgd}}$ in this section.

D.2.1 Supporting Results

We begin by deriving the even moments of random variables.

• $2n$**'th moment of a normally distributed variable:** Let $u \sim \mathcal{N}(0, \sigma^2)$. Then we have

$$\mathbb{E}[u^{2n}] = \sigma^{2n}(2n-1)!!. \tag{D.13}$$

• 4**'th moment:** Let $\boldsymbol{u} \sim \mathcal{N}(0, \boldsymbol{I}_d)$. Then for any $\boldsymbol{W}, \boldsymbol{W}' \in \mathbb{R}^{d\times d}$, we have

$$\begin{aligned}
&\mathbb{E}\left[(\boldsymbol{u}^\top \boldsymbol{W}\boldsymbol{u})(\boldsymbol{u}^\top \boldsymbol{W}'\boldsymbol{u})\right] \\
&= \mathbb{E}\left[\left(\sum_{i,j=1}^d W_{ij}u_iu_j\right)\left(\sum_{i,j=1}^d W'_{ij}u_iu_j\right)\right] \\
&= \mathbb{E}\left[\left(\sum_{i=1}^d W_{ii}u_i^2\right)\left(\sum_{i=1}^d W'_{ii}u_i^2\right)\right] + \mathbb{E}\left[\left(\sum_{i\neq j} W_{ij}u_iu_j\right)\left(\sum_{i\neq j} W'_{ij}u_iu_j\right)\right] \\
&= \sum_{i=1}^d W_{ii}W'_{ii}\,\mathbb{E}\left[u_i^4\right] + \sum_{i\neq j} W_{ii}W'_{jj}\,\mathbb{E}[u_i^2]\,\mathbb{E}[u_j^2] + \sum_{i\neq j} W_{ij}W'_{ij}\,\mathbb{E}[u_i^2]\,\mathbb{E}[u_j^2] + \sum_{i\neq j} W_{ij}W'_{ji}\,\mathbb{E}[u_i^2]\,\mathbb{E}[u_j^2] \\
&= 3\sum_{i=1}^d W_{ii}W'_{ii} + \sum_{i\neq j} W_{ii}W'_{jj} + \sum_{i\neq j} W_{ij}W'_{ij} + \sum_{i\neq j} W_{ij}W'_{ji} \\
&= \sum_{i,j=1}^d W_{ii}W'_{jj} + \sum_{i,j=1}^d W_{ij}W'_{ij} + \sum_{i,j=1}^d W_{ij}W'_{ji} \\
&= \mathtt{tr}\left(\boldsymbol{W}\right)\mathtt{tr}\left(\boldsymbol{W}'\right) + \mathtt{tr}\left(\boldsymbol{W}'\boldsymbol{W}^\top\right) + \mathtt{tr}\left(\boldsymbol{W}\boldsymbol{W}'\right).
\end{aligned} \tag{D.14}$$

• 4**'th cross-moment:** Let $\boldsymbol{u}, \boldsymbol{v} \sim \mathcal{N}(0, \boldsymbol{I}_d)$ and for any $\boldsymbol{W} \in \mathbb{R}^{d\times d}$, let $\boldsymbol{\Lambda}_{\boldsymbol{W}} = \boldsymbol{W} \odot \boldsymbol{I}_d$.

Then we have

$$
\begin{aligned}
&\mathbb{E}\left[(\boldsymbol{u}^\top \boldsymbol{W}\boldsymbol{v}\boldsymbol{v}^\top \boldsymbol{u})^2\right] \\
&= \mathbb{E}\left[\left(\sum_{i,j=1}^d W_{ij}u_iv_j\right)^2\left(\sum_{i=1}^d u_iv_i\right)^2\right] \\
&= \mathbb{E}\left[\left(\sum_{i,j=1}^d W_{ij}^2u_i^2v_j^2\right)\left(\sum_{i=1}^d u_i^2v_i^2\right) + \left(\sum_{i\neq j} W_{ij}W_{ji}u_i^2u_j^2v_i^2v_j^2\right)\right] \\
&= \mathbb{E}\left[\left(\sum_{i=1}^d W_{ii}^2u_i^2v_i^2 + \sum_{i\neq j} W_{ij}^2u_i^2v_j^2\right)\left(\sum_{i=1}^d u_i^2v_i^2\right)\right] + \sum_{i\neq j} W_{ij}W_{ji} \\
&= 9\sum_{i=1}^d W_{ii}^2 + (d-1)\sum_{i=1}^d W_{ii}^2 + 6\sum_{i\neq j} W_{ij}^2 + (d-2)\sum_{i\neq j} W_{ij}^2 + \sum_{i\neq j} W_{ij}W_{ji} \\
&= 3\sum_{i=1}^d W_{ii}^2 + (d+4)\sum_{i,j=1}^d W_{ij}^2 + \sum_{i,j=1}^d W_{ij}W_{ji} \\
&= 3\mathtt{tr}\left(\boldsymbol{\Lambda}_{\boldsymbol{W}}^2\right) + (d+4)\mathtt{tr}\left(\boldsymbol{W}\boldsymbol{W}^\top\right) + \mathtt{tr}\left(\boldsymbol{W}^2\right).
\end{aligned} \tag{D.15}
$$

• **6'th moment:** Let $\boldsymbol{u} \sim \mathcal{N}(0, \boldsymbol{I}_d)$. Then for any $\boldsymbol{W}, \boldsymbol{W}' \in \mathbb{R}^{d\times d}$, we have

$$
\begin{aligned}
&\mathbb{E}\left[(\boldsymbol{u}^\top \boldsymbol{W}\boldsymbol{u})(\boldsymbol{u}^\top \boldsymbol{W}'\boldsymbol{u})\|\boldsymbol{u}\|^2\right] \\
&= \mathbb{E}\left[\left(\sum_{i,j=1}^d W_{ij}u_iu_j\right)\left(\sum_{i,j=1}^d W'_{ij}u_iu_j\right)\left(\sum_{i=1}^d u_i^2\right)\right] \\
&= \mathbb{E}\left[\left(\sum_{i=1}^d W_{ii}u_i^2\right)\left(\sum_{i=1}^d W'_{ii}u_i^2\right)\left(\sum_{i=1}^d u_i^2\right)\right] + \mathbb{E}\left[\left(\sum_{i\neq j} W_{ij}u_iu_j\right)\left(\sum_{i\neq j} W'_{ij}u_iu_j\right)\left(\sum_{i=1}^d u_i^2\right)\right] \\
&= \sum_{i=1}^d W_{ii}W'_{ii}\,\mathbb{E}\left[u_i^4\left(\sum_{i'=1}^d u_{i'}^2\right)\right] + \sum_{i\neq j} W_{ii}W'_{jj}\,\mathbb{E}\left[u_i^2u_j^2\left(\sum_{i'=1}^d u_{i'}^2\right)\right] \\
&\quad + \sum_{i\neq j} W_{ij}W'_{ij}\,\mathbb{E}\left[u_i^2u_j^2\left(\sum_{i'=1}^d u_{i'}^2\right)\right] + \sum_{i\neq j} W_{ij}W'_{ji}\,\mathbb{E}\left[u_i^2u_j^2\left(\sum_{i'=1}^d u_{i'}^2\right)\right] \\
&= (d+4)\left(3\sum_{i=1}^d W_{ii}W'_{ii} + \sum_{i\neq j} W_{ii}W'_{jj} + \sum_{i\neq j} W_{ij}W'_{ij} + \sum_{i\neq j} W_{ij}W'_{ji}\right) \\
&= (d+4)\left(\sum_{i,j=1}^d W_{ii}W'_{jj} + \sum_{i,j=1}^d W_{ij}W'_{ij} + \sum_{i,j=1}^d W_{ij}W'_{ji}\right) \\
&= (d+4)\left(\mathtt{tr}\left(\boldsymbol{W}\right)\mathtt{tr}\left(\boldsymbol{W}'\right) + \mathtt{tr}\left(\boldsymbol{W}'\boldsymbol{W}^\top\right) + \mathtt{tr}\left(\boldsymbol{W}\boldsymbol{W}'\right)\right),
\end{aligned} \tag{D.16}
\tag{D.17}
$$

where (D.16) is obtained by following

$$\mathbb{E}\left[u_i^4\left(\sum_{i'=1}^d u_{i'}^2\right)\right] = \mathbb{E}[u^6] + (d-1)\,\mathbb{E}[u^4]\,\mathbb{E}[u^2] = 3(d+4),$$
$$\mathbb{E}\left[u_i^2 u_j^2\left(\sum_{i'=1}^d u_{i'}^2\right)\right] = 2\,\mathbb{E}[u^4]\,\mathbb{E}[u^2] + (d-2)\,\mathbb{E}[u^2]\,\mathbb{E}[u^2]\,\mathbb{E}[u^2] = d+4.$$

- 8**'th moment:** Let $\boldsymbol{u} \sim \mathcal{N}(0, \boldsymbol{I}_d)$. Then for any $\boldsymbol{W}, \boldsymbol{W}' \in \mathbb{R}^{d\times d}$, we have

$$\begin{aligned}
&\mathbb{E}\left[(\boldsymbol{u}^\top \boldsymbol{W}\boldsymbol{u})(\boldsymbol{u}^\top \boldsymbol{W}'\boldsymbol{u})\|\boldsymbol{u}\|^4\right] \\
&= \mathbb{E}\left[\left(\sum_{i,j=1}^d W_{ij}u_iu_j\right)\left(\sum_{i,j=1}^d W'_{ij}u_iu_j\right)\left(\sum_{i,j=1}^d u_i^2u_j^2\right)\right] \\
&= \sum_{i=1}^d W_{ii}W'_{ii}\,\mathbb{E}\left[u_i^4\left(\sum_{i'=1}^d u_{i'}^4 + \sum_{i'\neq j'} u_{i'}^2u_{j'}^2\right)\right] + \sum_{i\neq j} W_{ii}W'_{jj}\,\mathbb{E}\left[u_i^2u_j^2\left(\sum_{i'=1}^d u_{i'}^4 + \sum_{i'\neq j'} u_{i'}^2u_{j'}^2\right)\right] \\
&\quad + \sum_{i\neq j} W_{ij}W'_{ij}\,\mathbb{E}\left[u_i^2u_j^2\left(\sum_{i'=1}^d u_{i'}^4 + \sum_{i'\neq j'} u_{i'}^2u_{j'}^2\right)\right] + \sum_{i\neq j} W_{ij}W'_{ji}\,\mathbb{E}\left[u_i^2u_j^2\left(\sum_{i'=1}^d u_{i'}^4 + \sum_{i'\neq j'} u_{i'}^2u_{j'}^2\right)\right] \\
&= (d+4)(d+6)\left(3\sum_{i=1}^d W_{ii}W'_{ii} + \sum_{i\neq j} W_{ii}W'_{jj} + \sum_{i\neq j} W_{ij}W'_{ij} + \sum_{i\neq j} W_{ij}W'_{ji}\right) \qquad \text{(D.18)} \\
&= (d+4)(d+6)\left(\sum_{i,j=1}^d W_{ii}W'_{jj} + \sum_{i,j=1}^d W_{ij}W'_{ij} + \sum_{i,j=1}^d W_{ij}W'_{ji}\right) \\
&= (d+4)(d+6)\left(\mathtt{tr}\left(\boldsymbol{W}\right)\mathtt{tr}\left(\boldsymbol{W}'\right) + \mathtt{tr}\left(\boldsymbol{W}'\boldsymbol{W}^\top\right) + \mathtt{tr}\left(\boldsymbol{W}\boldsymbol{W}'\right)\right). \qquad \text{(D.19)}
\end{aligned}$$

where (D.18) is obtained by following

$$
\begin{aligned}
&\mathbb{E}\left[u_i^4\left(\sum_{i'=1}^{d} u_{i'}^4 + \sum_{i'\neq j'} u_{i'}^2 u_{j'}^2\right)\right] \\
&= \mathbb{E}[u^8] + (d-1)\,\mathbb{E}[u^4]\,\mathbb{E}[u^4] + 2(d-1)\,\mathbb{E}[u^6]\,\mathbb{E}[u^2] + (d-1)(d-2)\,\mathbb{E}[u^4]\,\mathbb{E}[u^2]\,\mathbb{E}[u^2] \\
&= 105 + 9(d-1) + 30(d-1) + 3(d-1)(d-2) \\
&= 3(d+4)(d+6), \\
&\mathbb{E}\left[u_i^2 u_j^2\left(\sum_{i'=1}^{d} u_{i'}^4 + \sum_{i'\neq j'} u_{i'}^2 u_{j'}^2\right)\right] \\
&= 2\,\mathbb{E}[u^6]\,\mathbb{E}[u^2] + (d-2)\,\mathbb{E}[u^4](\mathbb{E}[u^2])^2 + 2\,\mathbb{E}[u^4]\,\mathbb{E}[u^4] \\
&\qquad + 4(d-2)\,\mathbb{E}[u^4](\mathbb{E}[u^2])^2 + (d-2)(d-3)(\mathbb{E}[u^2])^4 \\
&= 30 + 3(d-2) + 18 + 12(d-2) + (d-2)(d-3) \\
&= (d+4)(d+6).
\end{aligned}
$$

D.2.2 Independent Data with General Covariance

Proof of Theorem 5.1. Consider a general independent linear model as defined in (5.7) where $\boldsymbol{\Sigma}_{\boldsymbol{x}}$ and $\boldsymbol{\Sigma}_{\boldsymbol{\beta}}$ are full-rank feature and task convariance matrices and

$$
\boldsymbol{x} \sim \mathcal{N}(0, \boldsymbol{\Sigma}_{\boldsymbol{x}}), \quad \boldsymbol{\beta} \sim \mathcal{N}(0, \boldsymbol{\Sigma}_{\boldsymbol{\beta}}), \quad \xi \sim \mathcal{N}(0, \sigma^2), \quad \text{and} \quad y = \boldsymbol{x}^\top \boldsymbol{\beta} + \xi.
$$

Let

$$
\boldsymbol{X} = [\boldsymbol{x}_1 \ \cdots \ \boldsymbol{x}_n]^\top, \quad \boldsymbol{\xi} = [\xi_1 \ \cdots \ \xi_n]^\top, \quad \text{and} \quad \boldsymbol{y} = [y_1 \ \cdots \ y_n]^\top = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\xi}.
$$

To simplify and without loss of generality, let $\bar{\boldsymbol{x}} = \boldsymbol{\Sigma}_{\boldsymbol{x}}^{-1/2}\boldsymbol{x}$, $\bar{\boldsymbol{X}} = \boldsymbol{X}\boldsymbol{\Sigma}_{\boldsymbol{x}}^{-1/2}$, $\bar{\boldsymbol{\theta}} = \boldsymbol{\Sigma}_{\boldsymbol{x}}^{1/2}\boldsymbol{\beta}$ where we have

$$
\bar{\boldsymbol{x}} \sim \mathcal{N}(0, \boldsymbol{I}), \qquad \bar{\boldsymbol{\theta}} \sim \mathcal{N}(0, \boldsymbol{\Sigma}_{\boldsymbol{x}}^{1/2}\boldsymbol{\Sigma}_{\boldsymbol{\beta}}\boldsymbol{\Sigma}_{\boldsymbol{x}}^{1/2})
$$

and

$$
y = \bar{\boldsymbol{x}}^\top \bar{\boldsymbol{\theta}} + \xi, \qquad \boldsymbol{y} = \bar{\boldsymbol{X}}\bar{\boldsymbol{\theta}} + \boldsymbol{\xi}.
$$

Then recap the loss from (5.5a), and we obtain

$$
\begin{aligned}
\mathcal{L}(\boldsymbol{W}) &= \mathbb{E}\left[(y - g(\boldsymbol{Z}))^2\right] \\
&= \mathbb{E}\left[\left(\boldsymbol{x}^\top\boldsymbol{\beta} + \xi - \boldsymbol{x}^\top\boldsymbol{W}\boldsymbol{X}^\top(\boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\xi})\right)^2\right] \\
&= \mathbb{E}\left[(\boldsymbol{x}^\top\boldsymbol{\beta} - \boldsymbol{x}^\top\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{X}\boldsymbol{\beta})^2\right] + \mathbb{E}\left[(\boldsymbol{x}^\top\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{\xi})^2\right] + \sigma^2, \qquad \text{(D.20)}
\end{aligned}
$$

where the last equality comes from the independence of label noise $\xi, \boldsymbol{\xi}$.

We first consider the following term

$$\mathbb{E}\left[(\boldsymbol{x}^\top \boldsymbol{W} \boldsymbol{X}^\top \boldsymbol{\xi})^2\right] = \mathbb{E}\left[(\bar{\boldsymbol{x}}^\top (\boldsymbol{\Sigma}_{\boldsymbol{x}}^{1/2} \boldsymbol{W} \boldsymbol{\Sigma}_{\boldsymbol{x}}^{1/2}) \bar{\boldsymbol{X}}^\top \boldsymbol{\xi})^2\right] = n\sigma^2 \cdot \mathtt{tr}\left(\bar{\boldsymbol{W}} \bar{\boldsymbol{W}}^\top\right)$$

where we define $\bar{\boldsymbol{W}} = \boldsymbol{\Sigma}_{\boldsymbol{x}}^{1/2} \boldsymbol{W} \boldsymbol{\Sigma}_{\boldsymbol{x}}^{1/2}$. Next, focus on the following

$$\begin{aligned}
&\mathbb{E}\left[(\boldsymbol{x}^\top \boldsymbol{\beta} - \boldsymbol{x}^\top \boldsymbol{W} \boldsymbol{X}^\top \boldsymbol{X} \boldsymbol{\beta})^2\right] \\
&= \mathbb{E}\left[(\bar{\boldsymbol{x}}^\top \bar{\boldsymbol{\theta}} - \bar{\boldsymbol{x}}^\top \bar{\boldsymbol{W}} \bar{\boldsymbol{X}}^\top \bar{\boldsymbol{X}} \bar{\boldsymbol{\theta}})^2\right] \\
&= \mathbb{E}\left[\left(\bar{\boldsymbol{x}}^\top \left(\boldsymbol{I} - \bar{\boldsymbol{W}} \bar{\boldsymbol{X}}^\top \bar{\boldsymbol{X}}\right) \bar{\boldsymbol{\theta}}\right)^2\right] \\
&= \mathtt{tr}\left(\mathbb{E}\left[\left(\boldsymbol{I} - \bar{\boldsymbol{W}} \bar{\boldsymbol{X}}^\top \bar{\boldsymbol{X}}\right) \boldsymbol{\Sigma} \left(\boldsymbol{I} - \bar{\boldsymbol{W}} \bar{\boldsymbol{X}}^\top \bar{\boldsymbol{X}}\right)^\top\right]\right) \\
&= \mathtt{tr}\left(\boldsymbol{\Sigma}\right) - \mathtt{tr}\left(\boldsymbol{\Sigma}(\bar{\boldsymbol{W}} + \bar{\boldsymbol{W}}^\top)\, \mathbb{E}[\bar{\boldsymbol{X}}^\top \bar{\boldsymbol{X}}]\right) + \mathtt{tr}\left(\bar{\boldsymbol{W}}^\top \bar{\boldsymbol{W}}\, \mathbb{E}[\bar{\boldsymbol{X}}^\top \bar{\boldsymbol{X}} \boldsymbol{\Sigma} \bar{\boldsymbol{X}}^\top \bar{\boldsymbol{X}}]\right) \\
&= \mathtt{tr}\left(\boldsymbol{\Sigma}\right) - 2n \cdot \mathtt{tr}\left(\boldsymbol{\Sigma} \bar{\boldsymbol{W}}\right) + \mathtt{tr}\left(\bar{\boldsymbol{W}}^\top \bar{\boldsymbol{W}}\, \mathbb{E}[\bar{\boldsymbol{X}}^\top \bar{\boldsymbol{X}} \boldsymbol{\Sigma} \bar{\boldsymbol{X}}^\top \bar{\boldsymbol{X}}]\right),
\end{aligned}$$

where $\boldsymbol{\Sigma} := \boldsymbol{\Sigma}_{\boldsymbol{x}}^{1/2} \boldsymbol{\Sigma}_{\boldsymbol{\beta}} \boldsymbol{\Sigma}_{\boldsymbol{x}}^{1/2}$.

Let $\bar{\boldsymbol{x}}_i \in \mathbb{R}^n$ be the i'th column of $\bar{\boldsymbol{X}}$ and $\boldsymbol{\Sigma}_{ij}$ be the (i,j)'th entry of $\boldsymbol{\Sigma}$. Then the (i,j) entry of matrix $\bar{\boldsymbol{X}}^\top \bar{\boldsymbol{X}} \boldsymbol{\Sigma} \bar{\boldsymbol{X}}^\top \bar{\boldsymbol{X}}$ is

$$(\bar{\boldsymbol{X}}^\top \bar{\boldsymbol{X}} \boldsymbol{\Sigma} \bar{\boldsymbol{X}}^\top \bar{\boldsymbol{X}})_{ij} = \sum_{k=1}^{d} \sum_{p=1}^{d} \boldsymbol{\Sigma}_{kp} \bar{\boldsymbol{x}}_i^\top \bar{\boldsymbol{x}}_k \bar{\boldsymbol{x}}_p^\top \bar{\boldsymbol{x}}_j.$$

Then we get

$$\begin{aligned}
i \neq j: \quad \mathbb{E}\left[\left(\bar{\boldsymbol{X}}^\top \bar{\boldsymbol{X}} \boldsymbol{\Sigma} \bar{\boldsymbol{X}}^\top \bar{\boldsymbol{X}}\right)_{ij}\right] &= \boldsymbol{\Sigma}_{ij}\, \mathbb{E}[\bar{\boldsymbol{x}}_i^\top \bar{\boldsymbol{x}}_i \bar{\boldsymbol{x}}_j^\top \bar{\boldsymbol{x}}_j] + \boldsymbol{\Sigma}_{ji}\, \mathbb{E}[\bar{\boldsymbol{x}}_i^\top \bar{\boldsymbol{x}}_j \bar{\boldsymbol{x}}_i^\top \bar{\boldsymbol{x}}_j] = n^2 \boldsymbol{\Sigma}_{ij} + n \boldsymbol{\Sigma}_{ji} \\
i = j: \quad \mathbb{E}\left[\left(\bar{\boldsymbol{X}}^\top \bar{\boldsymbol{X}} \boldsymbol{\Sigma} \bar{\boldsymbol{X}}^\top \bar{\boldsymbol{X}}\right)_{ii}\right] &= \boldsymbol{\Sigma}_{ii}\, \mathbb{E}\left[\bar{\boldsymbol{x}}_i^\top \bar{\boldsymbol{x}}_i \bar{\boldsymbol{x}}_i^\top \bar{\boldsymbol{x}}_i\right] + \sum_{j \neq i} \boldsymbol{\Sigma}_{jj}\, \mathbb{E}\left[\bar{\boldsymbol{x}}_i^\top \bar{\boldsymbol{x}}_j \bar{\boldsymbol{x}}_j^\top \bar{\boldsymbol{x}}_i\right] \\
&= \boldsymbol{\Sigma}_{ii}\, \mathbb{E}\left[(x_{i1}^2 + \cdots + x_{in}^2)^2\right] + n \sum_{j \neq i} \boldsymbol{\Sigma}_{jj} \\
&= \boldsymbol{\Sigma}_{ii}(3n + n(n-1)) + n \sum_{j \neq i} \boldsymbol{\Sigma}_{jj} \\
&= n\left(\boldsymbol{\Sigma}_{ii}(n+1) + \sum_{j=1}^{d} \boldsymbol{\Sigma}_{jj}\right) \\
&= n\left(\boldsymbol{\Sigma}_{ii}(n+1) + \mathtt{tr}\left(\boldsymbol{\Sigma}\right)\right).
\end{aligned}$$

Therefore

$$\mathbb{E}[\bar{\boldsymbol{X}}^\top \bar{\boldsymbol{X}} \boldsymbol{\Sigma} \bar{\boldsymbol{X}}^\top \bar{\boldsymbol{X}}] = n(n+1)\boldsymbol{\Sigma} + n \cdot \mathtt{tr}\left(\boldsymbol{\Sigma}\right) \boldsymbol{I}.$$

Combining all together results in

$$\begin{aligned}\mathcal{L}(\boldsymbol{W}) &= \mathtt{tr}\left(\boldsymbol{\Sigma}\right) - 2n\mathtt{tr}\left(\boldsymbol{\Sigma}\bar{\boldsymbol{W}}\right) + n(n+1)\mathtt{tr}\left(\boldsymbol{\Sigma}\bar{\boldsymbol{W}}^\top\bar{\boldsymbol{W}}\right) + n(\mathtt{tr}\left(\boldsymbol{\Sigma}\right)+\sigma^2)\mathtt{tr}\left(\bar{\boldsymbol{W}}\bar{\boldsymbol{W}}^\top\right) + \sigma^2, \\ &= M - 2n\mathtt{tr}\left(\boldsymbol{\Sigma}\bar{\boldsymbol{W}}\right) + n(n+1)\mathtt{tr}\left(\boldsymbol{\Sigma}\bar{\boldsymbol{W}}^\top\bar{\boldsymbol{W}}\right) + nM\mathtt{tr}\left(\bar{\boldsymbol{W}}\bar{\boldsymbol{W}}^\top\right), \end{aligned} \tag{D.21}$$

where $M := \mathtt{tr}\left(\boldsymbol{\Sigma}\right) + \sigma^2$. Setting $\nabla_{\bar{\boldsymbol{W}}}\mathcal{L}(\boldsymbol{W}) = 0$ returns

$$-2n\cdot\boldsymbol{\Sigma} + 2n(n+1)\cdot\boldsymbol{\Sigma}\bar{\boldsymbol{W}} + 2nM\bar{\boldsymbol{W}} = 0 \Longrightarrow \bar{\boldsymbol{W}}_\star = \left((n+1)\boldsymbol{I} + M\boldsymbol{\Sigma}^{-1}\right)^{-1}.$$

Then we have

$$\boldsymbol{W}_\star = \boldsymbol{\Sigma}_{\boldsymbol{x}}^{-1/2}\left((n+1)\boldsymbol{I} + M\boldsymbol{\Sigma}^{-1}\right)^{-1}\boldsymbol{\Sigma}_{\boldsymbol{x}}^{-1/2}$$

and

$$\mathcal{L}_\star = \mathcal{L}(\boldsymbol{W}_\star) = M - n\mathtt{tr}\left(((n+1)\boldsymbol{\Sigma}^{-1} + M\boldsymbol{\Sigma}^{-2})^{-1}\right).$$

■

D.2.3 Retrieval Augmented Generation with α Correlation

In this section, we consider the retrieval augmented generation (RAG) linear model similar to (5.9), where we first draw the query vector $\boldsymbol{x}$ and task vector $\boldsymbol{\beta}$ via

$$\boldsymbol{x} \sim \mathcal{N}(0, \boldsymbol{I}) \quad \text{and} \quad \boldsymbol{\beta} \sim \mathcal{N}(0, \boldsymbol{I}).$$

We then draw data $(\boldsymbol{x}_i)_{i=1}^n$ to be used in-context according to the rule $\text{corr\_coef}(\boldsymbol{x}, \boldsymbol{x}_i) \geq \alpha \geq 0$. Hence, for $i \leq n$ we sample

$$\boldsymbol{x}_i \;\big|\; \boldsymbol{x} \sim \mathcal{N}(\alpha\boldsymbol{x}, \gamma^2\boldsymbol{I}), \quad \xi_i \sim \mathcal{N}(0, \sigma^2) \quad \text{and} \quad y_i = \boldsymbol{x}_i^\top\boldsymbol{\beta} + \xi_i, \tag{D.22}$$

which results in (5.9) by setting $\gamma^2 = 1 - \alpha^2$.

Theorem D.1 (Extended version of Theorem 5.2) *Consider linear model as defined in* (D.22). *Recap the objective from* (5.5a) *and let* $\boldsymbol{W}_\star := \arg\min_{\boldsymbol{W}} \mathcal{L}_{pgd}(\boldsymbol{W})$, *and* $\mathcal{L}_\star = \mathcal{L}_{pgd}(\boldsymbol{W}_\star)$. *Then* $\boldsymbol{W}^{mm}$ *and* $\mathcal{L}_\star$ *satisfy*

$$\boldsymbol{W}_\star = c\boldsymbol{I} \qquad \textit{and} \qquad \mathcal{L}_\star = d + \sigma^2 - cnd(\alpha^2(d+2) + \gamma^2) \tag{D.23}$$

where

$$c = \frac{\alpha^2(d+2)+\gamma^2}{\alpha^4 n(d+2)(d+4)+\alpha^2\gamma^2(d+2)(d+2n+3)+\gamma^4(d+n+1)+\sigma^2(\alpha^2(d+2)+\gamma^2)}.$$

Suppose $\alpha = \mathcal{O}(1/\sqrt{d})$, $d/n = \mathcal{O}(1)$ *and* d *is sufficiently large. Let* $\kappa = \alpha^2 d + 1$ *and* $\gamma^2 = 1-\alpha^2$. *Then* $\boldsymbol{W}^{mm}$ *and* $\mathcal{L}_\star$ *have approximate forms*

$$\boldsymbol{W}_\star \approx \frac{1}{\kappa n + d + \sigma^2}\boldsymbol{I} \qquad \textit{and} \qquad \mathcal{L}_\star \approx d + \sigma^2 - \frac{\kappa n d}{\kappa n + d + \sigma^2}. \tag{D.24}$$

Proof. Here, for clean notation and without loss of generality, we define and rewrite (D.22) via

$$\boldsymbol{g}_i \sim \mathcal{N}(0, \boldsymbol{I}), \quad \xi_i \sim \mathcal{N}(0,\sigma^2) \quad \text{and} \quad \boldsymbol{x}_i = \alpha\boldsymbol{x} + \gamma\boldsymbol{g}_i, \quad y_i = (\alpha\boldsymbol{x}+\gamma\boldsymbol{g}_i)^\top\boldsymbol{\beta} + \xi_i.$$

Then we obtain

$$\begin{aligned}
\mathcal{L}(\boldsymbol{W}) &= \mathbb{E}\left[(y - g(\boldsymbol{Z}))^2\right] \\
&= \mathbb{E}\left[\left(\boldsymbol{x}^\top\boldsymbol{\beta} + \xi - \boldsymbol{x}^\top\boldsymbol{W}\boldsymbol{X}^\top(\boldsymbol{X}\boldsymbol{\beta}+\boldsymbol{\xi})\right)^2\right] \\
&= \mathbb{E}\left[(\boldsymbol{x}^\top\boldsymbol{\beta} - \boldsymbol{x}^\top\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{X}\boldsymbol{\beta})^2\right] + \mathbb{E}\left[(\boldsymbol{x}^\top\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{\xi})^2\right] + \sigma^2.
\end{aligned} \tag{D.25}$$

To begin with, let

$$N_1 = \mathtt{tr}\left(\boldsymbol{W}\right)^2 + \mathtt{tr}\left(\boldsymbol{W}\boldsymbol{W}^\top\right) + \mathtt{tr}\left(\boldsymbol{W}^2\right), \quad N_2 = \mathtt{tr}\left(\boldsymbol{W}\boldsymbol{W}^\top\right), \quad \text{and} \quad N_3 = \mathtt{tr}\left(\boldsymbol{W}\right).$$

We first focus on the second term in (D.25)

$$\begin{aligned}
\mathbb{E}\left[(\boldsymbol{x}^\top\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{\xi})^2\right] &= \mathbb{E}\left[\left(\sum_{i=1}^n \xi_i \boldsymbol{x}^\top\boldsymbol{W}(\alpha\boldsymbol{x}+\gamma\boldsymbol{g}_i)\right)^2\right] \\
&= n\sigma^2\,\mathbb{E}\left[\boldsymbol{x}^\top\boldsymbol{W}(\alpha\boldsymbol{x}+\gamma\boldsymbol{g})(\alpha\boldsymbol{x}+\gamma\boldsymbol{g})^\top\boldsymbol{W}^\top\boldsymbol{x}\right] \\
&= n\sigma^2\left(\alpha^2\,\mathbb{E}[\boldsymbol{x}^\top\boldsymbol{W}\boldsymbol{x}\boldsymbol{x}^\top\boldsymbol{W}^\top\boldsymbol{x}] + \gamma^2\,\mathbb{E}[\boldsymbol{x}^\top\boldsymbol{W}\boldsymbol{g}\boldsymbol{g}^\top\boldsymbol{W}^\top\boldsymbol{x}]\right) \\
&= n\sigma^2\left(\alpha^2 N_1 + \gamma^2 N_2\right). \quad \text{(It follows (D.14) and independence of } \boldsymbol{x}, \boldsymbol{g}.\text{)}
\end{aligned}$$

Next, the first term in (D.25) can be decomposed into

$$\mathbb{E}\left[(\boldsymbol{x}^\top\boldsymbol{\beta} - \boldsymbol{x}^\top\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{X}\boldsymbol{\beta})^2\right] = \underbrace{\mathbb{E}\left[(\boldsymbol{x}^\top\boldsymbol{\beta})^2\right]}_{(a)} + \underbrace{\mathbb{E}\left[(\boldsymbol{x}^\top\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{X}\boldsymbol{\beta})^2\right]}_{(b)} - 2\underbrace{\mathbb{E}\left[\boldsymbol{x}^\top\boldsymbol{\beta}\boldsymbol{x}^\top\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{X}\boldsymbol{\beta}\right]}_{(c)}.$$

In the following, we consider solving (a)-(c) sequentially.

$$(a): \quad \mathbb{E}\left[(\boldsymbol{x}^\top\boldsymbol{\beta})^2\right] = d.$$

$$\begin{aligned}
(b): \quad & \mathbb{E}\left[(\boldsymbol{x}^\top\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{X}\boldsymbol{\beta})^2\right] \\
&= \mathbb{E}\left[\left(\boldsymbol{x}^\top\boldsymbol{W}\sum_{i=1}^n(\alpha\boldsymbol{x}+\gamma\boldsymbol{g}_i)(\alpha\boldsymbol{x}+\gamma\boldsymbol{g}_i)^\top\boldsymbol{\beta}\right)^2\right] \\
&= \mathbb{E}\left[\left(\sum_{i=1}^n\boldsymbol{x}^\top\boldsymbol{W}(\alpha^2\boldsymbol{x}\boldsymbol{x}^\top+\gamma^2\boldsymbol{g}_i\boldsymbol{g}_i^\top+\alpha\gamma\boldsymbol{x}\boldsymbol{g}_i^\top+\alpha\gamma\boldsymbol{g}_i\boldsymbol{x}^\top)\boldsymbol{\beta}\right)^2\right] \\
&= \left(\alpha^4n^2(d+4)+\alpha^2\gamma^2n(2n+d+2)\right)N_1+\left(\alpha^2\gamma^2n(d+2)+\gamma^4n(d+n+1)\right)N_2 \\
&= A_1N_1+A_2N_2.
\end{aligned}$$

$$\begin{aligned}
(c): \quad & \mathbb{E}\left[\boldsymbol{x}^\top\boldsymbol{\beta}\boldsymbol{x}^\top\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{X}\boldsymbol{\beta}\right] = \mathbb{E}\left[\sum_{i=1}^n\boldsymbol{x}^\top\boldsymbol{\beta}\boldsymbol{x}^\top\boldsymbol{W}(\alpha\boldsymbol{x}+\gamma\boldsymbol{g}_i)(\alpha\boldsymbol{x}+\gamma\boldsymbol{g}_i)^\top\boldsymbol{\beta}\right] \\
&= \mathbb{E}\left[\sum_{i=1}^n\boldsymbol{x}^\top\boldsymbol{\beta}\boldsymbol{x}^\top\boldsymbol{W}(\alpha^2\boldsymbol{x}\boldsymbol{x}^\top+\gamma^2\boldsymbol{g}_i\boldsymbol{g}_i^\top+\alpha\gamma\boldsymbol{x}\boldsymbol{g}_i^\top+\alpha\gamma\boldsymbol{g}_i\boldsymbol{x}^\top)\boldsymbol{\beta}\right] \\
&= \alpha^2n\,\mathbb{E}\left[\boldsymbol{x}^\top\boldsymbol{\beta}\boldsymbol{x}^\top\boldsymbol{W}\boldsymbol{x}\boldsymbol{x}^\top\boldsymbol{\beta}\right]+\gamma^2n\,\mathbb{E}\left[\boldsymbol{x}^\top\boldsymbol{\beta}\boldsymbol{x}^\top\boldsymbol{W}\boldsymbol{g}\boldsymbol{g}^\top\boldsymbol{\beta}\right] \\
&= \alpha^2n(d+2)\mathtt{tr}\left(\boldsymbol{W}\right)+\gamma^2n\mathtt{tr}\left(\boldsymbol{W}\right) \\
&= \left(\alpha^2n(d+2)+\gamma^2n\right)N_3 \\
&= A_3N_3.
\end{aligned}$$

Here, (b) utilizes the 4'th and 6'th moment results (D.14) and (D.17) and we define

$$\begin{aligned}
A_1 &= \alpha^4n^2(d+4)+\alpha^2\gamma^2n(2n+d+2) \\
A_2 &= \alpha^2\gamma^2n(d+2)+\gamma^4n(d+n+1) \\
A_3 &= \alpha^2n(d+2)+\gamma^2n.
\end{aligned}$$

Then combining all together results in

$$\mathcal{L}(\boldsymbol{W}) = A_1N_1+A_2N_2-2A_3N_3+n\sigma^2(\alpha^2N_1+\gamma^2N_2)+d+\sigma^2.$$

To find the optimal solution, set $\nabla\mathcal{L}(\boldsymbol{W}) = 0$ and we obtain

$$A_1\nabla N_1 + A_2\nabla N_2 - 2A_3\nabla N_3 + n\sigma^2(\alpha^2\nabla N_1 + \gamma^2\nabla N_2) = 0. \tag{D.26}$$

Note that we have

$$\begin{aligned}
\nabla N_1 &= \nabla\left(\mathtt{tr}\left(\boldsymbol{W}\right)^2 + \mathtt{tr}\left(\boldsymbol{W}\boldsymbol{W}^\top\right) + \mathtt{tr}\left(\boldsymbol{W}^2\right)\right) = 2\mathtt{tr}\left(\boldsymbol{W}\right)\boldsymbol{I} + 2\boldsymbol{W} + 2\boldsymbol{W}^\top \\
\nabla N_2 &= \nabla\mathtt{tr}\left(\boldsymbol{W}\boldsymbol{W}^\top\right) = 2\boldsymbol{W} \\
\nabla N_3 &= \nabla\mathtt{tr}\left(\boldsymbol{W}\right) = \boldsymbol{I}.
\end{aligned}$$

Therefore, (D.26) returns

$$2A_1\left(\mathtt{tr}\left(\boldsymbol{W}\right)\boldsymbol{I} + \boldsymbol{W} + \boldsymbol{W}^\top\right) + 2A_2\boldsymbol{W} - 2A_3 + 2n\sigma^2(\alpha^2(\mathtt{tr}\left(\boldsymbol{W}\right)\boldsymbol{I} + \boldsymbol{W} + \boldsymbol{W}^\top) + \gamma^2\boldsymbol{W})\boldsymbol{I} = 0, \tag{D.27}$$

which implies that the optimal solution $\boldsymbol{W}_\star$ has the form of $c\boldsymbol{I}$ for some constant c. Then suppose $\boldsymbol{W}_\star = c\boldsymbol{I}$, we have $\mathtt{tr}\left(\boldsymbol{W}\right) = cd$ and (D.27) returns

$$2A_1(d+2)c\boldsymbol{I} + 2A_2c\boldsymbol{I} - 2A_3\boldsymbol{I} + 2n\sigma^2(\alpha^2(d+2)c\boldsymbol{I} + \gamma^2c\boldsymbol{I}) = 0$$

$$\begin{aligned}
\Longrightarrow c &= \frac{A_3}{A_1(d+2) + A_2 + n\sigma^2(\alpha^2(d+2) + \gamma^2)} \\
&= \frac{\alpha^2(d+2) + \gamma^2}{\alpha^4n(d+2)(d+4) + \alpha^2\gamma^2(d+2)(d+2n+3) + \gamma^4(d+n+1) + \sigma^2(\alpha^2(d+2) + \gamma^2)}.
\end{aligned}$$

Then the optimal loss is obtained by setting $\boldsymbol{W}_\star = c\boldsymbol{I}$ and

$$\begin{aligned}
\mathcal{L}_\star = \mathcal{L}(\boldsymbol{W}_\star) &= A_1c^2d(d+2) + A_2c^2d - 2A_3cd + n\sigma^2c^2d(\alpha^2(d+2) + \gamma^2) + d + \sigma^2 \\
&= c^2d\left(A_1(d+2) + A_2 + n\sigma^2(\alpha^2(d+2) + \gamma^2)\right) - 2A_3cd + d + \sigma^2 \\
&= d + \sigma^2 - A_3cd.
\end{aligned}$$

It completes the proof of (D.23). Now if assuming $\alpha = \mathcal{O}(1/\sqrt{d})$, $d/n = \mathcal{O}(1)$ and sufficiently large dimension d, we have the approximate

$$\begin{aligned} c &\approx \frac{\alpha^2 d + 1}{\alpha^4 d^2 n + \alpha^2 d(d+2n) + (d+n) + \sigma^2(\alpha^2 d + 1)} \\ &= \frac{\alpha^2 d + 1}{(\alpha^2 d + 1)^2 n + (\alpha^2 d + 1)d + \sigma^2(\alpha^2 d + 1)} \\ &= \frac{1}{(\alpha^2 d + 1)n + d + \sigma^2} \end{aligned}$$

and

$$\mathcal{L}_\star \approx d + \sigma^2 - \frac{(\alpha^2 d + 1)nd}{(\alpha^2 d + 1)n + d + \sigma^2}.$$

■

D.2.4 Task-feature Alignment with α Correlation

In this section, we consider the task-feature alignment data model similar to (5.11), where we first draw task vector $\boldsymbol{\beta}$ via

$$\boldsymbol{\beta} \sim \mathcal{N}(0, \boldsymbol{I}).$$

Then we generate examples $(\boldsymbol{x}_i, y_i)_{i=1}^{n+1}$ according to the rule corr_coef$(\boldsymbol{x}_i, \boldsymbol{\beta}) \geq \alpha \geq 0$ via

$$\boldsymbol{x}_i \;\big|\; \boldsymbol{\beta} \sim \mathcal{N}(\alpha\boldsymbol{\beta}, \boldsymbol{I}), \quad \xi_i \sim \mathcal{N}(0, \sigma^2) \quad \text{and} \quad y_i = \gamma \cdot \boldsymbol{x}_i^\top \boldsymbol{\beta} + \xi_i, \tag{D.28}$$

which results in (5.11) by setting $\gamma^2 = 1/(\alpha^2 d + 1)$.

Theorem D.2 (Extended version of Theorem 5.3) *Consider linear model as defined in* (D.28)*. Recap the objective from* (5.5a) *and let* $\boldsymbol{W}_\star := \arg\min_{\boldsymbol{W}} \mathcal{L}_{pgd}(\boldsymbol{W})$*, and* $\mathcal{L}_\star = \mathcal{L}_{pgd}(\boldsymbol{W}_\star)$*. Then* $\boldsymbol{W}_\star$ *and* $\mathcal{L}_\star$ *satisfy*

$$\boldsymbol{W}_\star = c\boldsymbol{I} \qquad \textit{and} \qquad \mathcal{L}_\star = d\gamma^2(\Delta_0\alpha^2 + 1) + \sigma^2 - cnd\gamma^2(\Delta_1\alpha^4 + 2\Delta_0\alpha^2 + 1) \tag{D.29}$$

where

$$c = \frac{\Delta_1\alpha^4 + 2\Delta_0\alpha^2 + 1}{\Delta_2\alpha^6 + \Delta_3\alpha^4 + \Delta_4\alpha^2 + (d+n+1) + \sigma^2(\Delta_0\alpha^4 + 2\alpha^2 + 1)/\gamma^2}$$

and

$$\begin{cases} \Delta_0 = d+2 \\ \Delta_1 = (d+2)(d+4) \\ \Delta_2 = (d+2)(d+4)(d+6)n \\ \Delta_3 = (d+2)(d+4)(3n+4) \\ \Delta_4 = (d+2)(3n+d+3) + (d+8). \end{cases}$$

Suppose $\alpha = \mathcal{O}(1/\sqrt{d})$, $d/n = \mathcal{O}(1)$ *and* d *is sufficiently large. Let* $\kappa = \alpha^2 d + 1$ *and* $\gamma^2 = 1/\kappa$. *Then* $\boldsymbol{W}^{mm}$ *and* $\mathcal{L}_\star$ *have approximate forms*

$$\boldsymbol{W}_\star \approx \frac{1}{\kappa n + (d+\sigma^2)/\kappa} \qquad \textit{and} \qquad \mathcal{L}_\star \approx d + \sigma^2 - \frac{\kappa n d}{\kappa n + (d+\sigma^2)/\kappa}. \tag{D.30}$$

Proof. Here, for clean notation and without loss of generality, we define and rewrite (D.28) via

$$\boldsymbol{g}_i \sim \mathcal{N}(0, \boldsymbol{I}), \quad \xi_i \sim \mathcal{N}(0, \sigma^2) \quad \text{and} \quad \boldsymbol{x}_i = \alpha\boldsymbol{\beta} + \boldsymbol{g}_i, \quad y_i = \gamma \boldsymbol{x}_i^\top \boldsymbol{\beta} + \xi_i = \gamma \cdot (\alpha\boldsymbol{\beta} + \boldsymbol{g}_i)^\top \boldsymbol{\beta} + \xi_i.$$

Recap the loss function from (5.5a), we obtain

$$\begin{aligned} \mathcal{L}(\boldsymbol{W}) &= \mathbb{E}\left[(y - g(\boldsymbol{Z}))^2\right] \\ &= \mathbb{E}\left[\left(\gamma \boldsymbol{x}^\top \boldsymbol{\beta} + \xi - \boldsymbol{x}^\top \boldsymbol{W} \boldsymbol{X}^\top (\gamma \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\xi})\right)^2\right] \\ &= \gamma^2 \mathbb{E}\left[(\boldsymbol{x}^\top \boldsymbol{\beta} - \boldsymbol{x}^\top \boldsymbol{W} \boldsymbol{X}^\top \boldsymbol{X}\boldsymbol{\beta})^2\right] + \mathbb{E}\left[(\boldsymbol{x}^\top \boldsymbol{W} \boldsymbol{X}^\top \boldsymbol{\xi})^2\right] + \sigma^2. \end{aligned} \tag{D.31}$$

Similar to Appendix D.2.3, to begin with, let

$$N_1 = \mathtt{tr}\left(\boldsymbol{W}\right)^2 + \mathtt{tr}\left(\boldsymbol{W}\boldsymbol{W}^\top\right) + \mathtt{tr}\left(\boldsymbol{W}^2\right), \quad N_2 = \mathtt{tr}\left(\boldsymbol{W}\boldsymbol{W}^\top\right), \quad \text{and} \quad N_3 = \mathtt{tr}\left(\boldsymbol{W}\right),$$

and additionally, given $\boldsymbol{\Lambda}_{\boldsymbol{W}} = \boldsymbol{W} \odot \boldsymbol{I}$, let

$$N_4 = 3\mathtt{tr}\left(\boldsymbol{\Lambda}_{\boldsymbol{W}}^2\right) + (d+4)\mathtt{tr}\left(\boldsymbol{W}\boldsymbol{W}^\top\right) + \mathtt{tr}\left(\boldsymbol{W}^2\right).$$

We first focus on the second term in (D.31)

$$\begin{aligned}
\mathbb{E}\left[(\boldsymbol{x}^\top \boldsymbol{W}\boldsymbol{X}^\top \boldsymbol{\xi})^2\right] &= \mathbb{E}\left[\left((\alpha\boldsymbol{\beta}+\boldsymbol{g})^\top \boldsymbol{W}\sum_{i=1}^n \xi_i(\alpha\boldsymbol{\beta}+\boldsymbol{g}_i)\right)^2\right] \\
&= n\sigma^2\,\mathbb{E}\left[\left((\alpha\boldsymbol{\beta}+\boldsymbol{g})^\top \boldsymbol{W}(\alpha\boldsymbol{\beta}+\boldsymbol{g}')\right)^2\right] \\
&= n\sigma^2\left(\alpha^4\,\mathbb{E}\left[(\boldsymbol{\beta}^\top\boldsymbol{W}\boldsymbol{\beta})^2\right] + 2\alpha^2\,\mathbb{E}\left[(\boldsymbol{\beta}^\top\boldsymbol{W}\boldsymbol{g}')^2\right] + \mathbb{E}\left[(\boldsymbol{g}^\top\boldsymbol{W}\boldsymbol{g}')^2\right]\right) \\
&= n\sigma^2\left(\alpha^4\left(\mathtt{tr}\left(\boldsymbol{W}\right)^2 + \mathtt{tr}\left(\boldsymbol{W}^2\right) + \mathtt{tr}\left(\boldsymbol{W}\boldsymbol{W}^\top\right)\right) + (2\alpha^2+1)\mathtt{tr}\left(\boldsymbol{W}\boldsymbol{W}^\top\right)\right) \\
&= n\sigma^2\left(\alpha^4 N_1 + (2\alpha^2+1)N_2\right).
\end{aligned}$$

(It follows (D.14) and independence of $\boldsymbol{\beta}, \boldsymbol{g}, \boldsymbol{g}'$.)

Next, the first term of (D.31) (omitting γ^2) returns the following decomposition:

$$\begin{aligned}
\mathbb{E}\left[(\boldsymbol{x}^\top\boldsymbol{\beta} - \boldsymbol{x}^\top\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{X}\boldsymbol{\beta})^2\right] &= \mathbb{E}\left[((\alpha\boldsymbol{\beta}+\boldsymbol{g})^\top(\boldsymbol{\beta} - \boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{X}\boldsymbol{\beta}))^2\right] \\
&= \mathbb{E}\left[\left(\alpha\boldsymbol{\beta}^\top\boldsymbol{\beta} - \alpha\boldsymbol{\beta}^\top\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{g}^\top\boldsymbol{\beta} - \boldsymbol{g}^\top\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{X}\boldsymbol{\beta}\right)^2\right] \\
&= \alpha^2 d(d+2) + \alpha^2\underbrace{\mathbb{E}[(\boldsymbol{\beta}^\top\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{X}\boldsymbol{\beta})^2]}_{(a)} + d + \underbrace{\mathbb{E}[(\boldsymbol{g}^\top\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{X}\boldsymbol{\beta})^2]}_{(b)} \\
&\quad - 2\alpha^2\underbrace{\mathbb{E}[\boldsymbol{\beta}^\top\boldsymbol{\beta}\boldsymbol{\beta}^\top\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{X}\boldsymbol{\beta}]}_{(c)} - 2\underbrace{\mathbb{E}[\boldsymbol{\beta}^\top\boldsymbol{g}\boldsymbol{g}^\top\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{X}\boldsymbol{\beta}]}_{(d)}.
\end{aligned}$$

Consider solving (a)-(d) sequentially as follows:

To begin with, we use the following decomposition for all (a)-(d):

$$\begin{aligned}
\boldsymbol{X}^\top\boldsymbol{X}\boldsymbol{\beta} &= \sum_{i=1}^n \boldsymbol{x}_i\boldsymbol{x}_i^\top\boldsymbol{\beta} \\
&= \sum_{i=1}^n (\alpha\boldsymbol{\beta}+\boldsymbol{g}_i)(\alpha\boldsymbol{\beta}+\boldsymbol{g}_i)^\top\boldsymbol{\beta} \\
&= \sum_{i=1}^n \alpha^2\boldsymbol{\beta}\boldsymbol{\beta}^\top\boldsymbol{\beta} + \alpha\boldsymbol{\beta}\boldsymbol{g}_i^\top\boldsymbol{\beta} + \alpha\boldsymbol{g}_i\boldsymbol{\beta}^\top\boldsymbol{\beta} + \boldsymbol{g}_i\boldsymbol{g}_i^\top\boldsymbol{\beta}.
\end{aligned}$$

Then, we have

$$
\begin{aligned}
(a): \quad & \mathbb{E}[(\boldsymbol{\beta}^\top \boldsymbol{W}\boldsymbol{X}^\top \boldsymbol{X}\boldsymbol{\beta})^2] \\
&= \mathbb{E}\left[\left(\sum_{i=1}^{n} \alpha^2 \boldsymbol{\beta}^\top \boldsymbol{W}\boldsymbol{\beta}\boldsymbol{\beta}^\top\boldsymbol{\beta} + \alpha\boldsymbol{\beta}^\top \boldsymbol{W}\boldsymbol{\beta}\boldsymbol{g}_i^\top\boldsymbol{\beta} + \alpha\boldsymbol{\beta}^\top \boldsymbol{W}\boldsymbol{g}_i\boldsymbol{\beta}^\top\boldsymbol{\beta} + \boldsymbol{\beta}^\top \boldsymbol{W}\boldsymbol{g}_i\boldsymbol{g}_i^\top\boldsymbol{\beta}\right)^2\right] \\
&= \alpha^4 n^2(d+4)(d+6)N_1 + \alpha^2 n(d+4)N_1 + \alpha^2 n(d+2)(d+4)N_2 && \text{(D.32)} \\
&\quad + n(n-1)N_1 + nN_4 && \text{(D.33)} \\
&\quad + 2\alpha^2 n^2(d+4)N_1 + 2\alpha^2 n(d+4)N_1 && \text{(D.34)} \\
&= \left(\alpha^2 n(d+4)(\alpha^2 n(d+6) + 2n + 3) + n(n-1)\right)N_1 + \alpha^2 n(d+2)(d+4)N_2 + nN_4 && \text{(D.35)} \\
&= B_1 N_1 + B_2 N_2 + nN_4,
\end{aligned}
$$

where (D.32) and (D.34) utilize (D.17) and (D.19), and (D.33) is obtained via

$$
\begin{aligned}
&\mathbb{E}\left[\left(\sum_{i=1}^{n} \boldsymbol{\beta}^\top \boldsymbol{W}\boldsymbol{g}_i\boldsymbol{g}_i^\top\boldsymbol{\beta}\right)^2\right] \\
&= n\,\mathbb{E}\left[\left(\boldsymbol{\beta}^\top \boldsymbol{W}\boldsymbol{g}'\boldsymbol{g}'^\top\boldsymbol{\beta}\right)^2\right] + n(n-1)\,\mathbb{E}\left[\boldsymbol{\beta}^\top \boldsymbol{W}\boldsymbol{g}'\boldsymbol{g}'^\top\boldsymbol{\beta}\boldsymbol{\beta}^\top \boldsymbol{W}\boldsymbol{g}''\boldsymbol{g}''^\top\boldsymbol{\beta}\right] \\
&= nN_4 + n(n-1)N_1,
\end{aligned}
$$

which follows (D.14) and (D.15).

$$
\begin{aligned}
(b): \quad & \mathbb{E}\left[(\boldsymbol{g}^\top \boldsymbol{W}\boldsymbol{X}^\top \boldsymbol{X}\boldsymbol{\beta})^2\right] \\
&= \mathbb{E}\left[\left(\sum_{i=1}^{n} \alpha^2 \boldsymbol{g}^\top \boldsymbol{W}\boldsymbol{\beta}\boldsymbol{\beta}^\top\boldsymbol{\beta} + \alpha\boldsymbol{g}^\top \boldsymbol{W}\boldsymbol{\beta}\boldsymbol{g}_i^\top\boldsymbol{\beta} + \alpha\boldsymbol{g}^\top \boldsymbol{W}\boldsymbol{g}_i\boldsymbol{\beta}^\top\boldsymbol{\beta} + \boldsymbol{g}^\top \boldsymbol{W}\boldsymbol{g}_i\boldsymbol{g}_i^\top\boldsymbol{\beta}\right)^2\right] \\
&= \alpha^4 n^2(d+2)(d+4)N_2 + \alpha^2 n(d+2)N_2 + \alpha^2 nd(d+2)N_2 + n(d+n+1)N_2 && \text{(D.36)} \\
&\quad + 2\alpha^2 n^2(d+2)N_2 + 2\alpha^2 n(d+2)N_2 && \text{(D.37)} \\
&= \left(\alpha^2 n(d+2)(\alpha^2 n(d+4) + 2n + d + 3) + n(d+n-1)\right)N_2 \\
&= B_3 N_2,
\end{aligned}
$$

where (D.36) and (D.37) are obtained using (D.14), (D.17) and

$$\begin{aligned}
&\mathbb{E}\left[\left(\sum_{i=1}^{n}\boldsymbol{g}^\top\boldsymbol{W}\boldsymbol{g}_i\boldsymbol{g}_i^\top\boldsymbol{\beta}\right)^2\right] \\
&= n\,\mathbb{E}\left[\left(\boldsymbol{g}^\top\boldsymbol{W}\boldsymbol{g}'\boldsymbol{g}'^\top\boldsymbol{\beta}\right)^2\right] + n(n-1)\,\mathbb{E}\left[\boldsymbol{g}^\top\boldsymbol{W}\boldsymbol{g}'\boldsymbol{g}'^\top\boldsymbol{\beta}\boldsymbol{g}^\top\boldsymbol{W}\boldsymbol{g}''\boldsymbol{g}''^\top\boldsymbol{\beta}\right] \\
&= n(d+2)N_2 + n(n-1)N_2 = n(n+d+1)N_2.
\end{aligned}$$

$$\begin{aligned}
(c): \quad &\mathbb{E}\left[\boldsymbol{\beta}^\top\boldsymbol{\beta}\boldsymbol{\beta}^\top\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{X}\boldsymbol{\beta}\right] \\
&= n\,\mathbb{E}\left[\boldsymbol{\beta}^\top\boldsymbol{\beta}\boldsymbol{\beta}^\top\boldsymbol{W}(\alpha\boldsymbol{\beta}+\boldsymbol{g}')(\alpha\boldsymbol{\beta}+\boldsymbol{g}')^\top\boldsymbol{\beta}\right] \\
&= \alpha^2 n\,\mathbb{E}\left[\boldsymbol{\beta}^\top\boldsymbol{\beta}\boldsymbol{\beta}^\top\boldsymbol{W}\boldsymbol{\beta}\boldsymbol{\beta}^\top\boldsymbol{\beta}\right] + n\,\mathbb{E}\left[\boldsymbol{\beta}^\top\boldsymbol{\beta}\boldsymbol{\beta}^\top\boldsymbol{W}\boldsymbol{g}'\boldsymbol{g}'^\top\boldsymbol{\beta}\right] \\
&= \alpha^2 n(d+2)(d+4)\mathtt{tr}\left(\boldsymbol{W}\right) + n(d+2)\mathtt{tr}\left(\boldsymbol{W}\right) \\
&= \left(\alpha^2 n(d+2)(d+4) + n(d+2)\right)N_3 \\
&= B_4 N_3.
\end{aligned}$$

$$\begin{aligned}
(d): \quad &\mathbb{E}\left[\boldsymbol{\beta}^\top\boldsymbol{g}\boldsymbol{g}^\top\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{X}\boldsymbol{\beta}\right] \\
&= n\,\mathbb{E}\left[\boldsymbol{\beta}^\top\boldsymbol{g}\boldsymbol{g}^\top\boldsymbol{W}(\alpha\boldsymbol{\beta}+\boldsymbol{g}')(\alpha\boldsymbol{\beta}+\boldsymbol{g}')^\top\boldsymbol{\beta}\right] \\
&= \alpha^2 n\,\mathbb{E}\left[\boldsymbol{\beta}^\top\boldsymbol{g}\boldsymbol{g}^\top\boldsymbol{W}\boldsymbol{\beta}\boldsymbol{\beta}^\top\boldsymbol{\beta}\right] + n\,\mathbb{E}\left[\boldsymbol{\beta}^\top\boldsymbol{g}\boldsymbol{g}^\top\boldsymbol{W}\boldsymbol{g}'\boldsymbol{g}'^\top\boldsymbol{\beta}\right] \\
&= \alpha^2 n(d+2)\mathtt{tr}\left(\boldsymbol{W}\right) + n\mathtt{tr}\left(\boldsymbol{W}\right) \\
&= \left(\alpha^2 n(d+2) + n\right)N_3 \\
&= B_5 N_3.
\end{aligned}$$

Here we define

$$\begin{aligned}
B_1 &= \alpha^2 n(d+4)(\alpha^2 n(d+6) + 2n + 3) + n(n-1) \\
B_2 &= \alpha^2 n(d+2)(d+4) \\
B_3 &= \alpha^2 n(d+2)(\alpha^2 n(d+4) + 2n + d + 3) + n(d+n-1) \\
B_4 &= \alpha^2 n(d+2)(d+4) + n(d+2) \\
B_5 &= \alpha^2 n(d+2) + n.
\end{aligned}$$

Then combining all together results in

$$
\begin{aligned}
&\mathcal{L}(\boldsymbol{W}) \\
&= \gamma^2\left(\alpha^2 d(d+2) + d + \alpha^2(B_1N_1 + B_2N_2 + nN_4) + B_3N_2 - 2\alpha^2 B_4N_3 - 2B_5N_3\right) + n\sigma^2(\alpha^4 N_1 + (2\alpha^2+1)N_2) + \sigma^2 \\
&= \gamma^2\left(\alpha^2 B_1N_1 + (\alpha^2B_2 + B_3)N_2 - 2(\alpha^2B_4 + B_5)N_3 + \alpha^2 nN_4\right) + n\sigma^2(\alpha^4 N_1 + (2\alpha^2+1)N_2) + \gamma^2 d\left(\alpha^2(d+2)+1\right) + \sigma^2
\end{aligned}
$$

and differentiating it results in

$$
\nabla\mathcal{L}(\boldsymbol{W}) = \gamma^2\left(\alpha^2 B_1\nabla N_1 + (\alpha^2B_2 + B_3)\nabla N_2 - 2(\alpha^2B_4 + B_5)\nabla N_3 + \alpha^2 n\nabla N_4\right) + n\sigma^2(\alpha^4\nabla N_1 + (2\alpha^2+1)\nabla N_2).
$$

Similar to the proof in Appendix D.2.3, $\boldsymbol{W}_\star$ has the form of $\boldsymbol{W}_\star = c\boldsymbol{I}$ and we have

$$
\begin{aligned}
\nabla N_1 &= \nabla\left(\mathtt{tr}\left(\boldsymbol{W}\right)^2 + \mathtt{tr}\left(\boldsymbol{W}\boldsymbol{W}^\top\right) + \mathtt{tr}\left(\boldsymbol{W}^2\right)\right) = 2\mathtt{tr}\left(\boldsymbol{W}\right)\boldsymbol{I} + 2\boldsymbol{W} + 2\boldsymbol{W}^\top = 2c(d+2)\boldsymbol{I} \\
\nabla N_2 &= \nabla\mathtt{tr}\left(\boldsymbol{W}\boldsymbol{W}^\top\right) = 2\boldsymbol{W} = 2c\boldsymbol{I} \\
\nabla N_3 &= \nabla\mathtt{tr}\left(\boldsymbol{W}\right) = \boldsymbol{I} \\
\nabla N_4 &= \nabla\left(3\mathtt{tr}\left(\boldsymbol{\Lambda}_{\boldsymbol{W}}^2\right) + (d+4)\mathtt{tr}\left(\boldsymbol{W}\boldsymbol{W}^\top\right) + \mathtt{tr}\left(\boldsymbol{W}^2\right)\right) \\
&= 6\cdot\mathrm{diag}(\boldsymbol{\Lambda}_{\boldsymbol{W}}) + 2(d+4)\boldsymbol{W} + 2\boldsymbol{W}^\top \\
&= 2c(d+8)\boldsymbol{I}.
\end{aligned}
$$

Therefore, setting $\nabla\mathcal{L}(\boldsymbol{W}) = 0$ returns

$$
\gamma^2\left(2c(d+2)\alpha^2B_1 + 2c(\alpha^2B_2 + B_3) - 2(\alpha^2B_4 + B_5) + 2c(d+8)\alpha^2 n\right) + 2cn\sigma^2(\alpha^4(d+2) + 2\alpha^2 + 1) = 0
$$

$$
\begin{aligned}
&\Longrightarrow c = \frac{\alpha^2B_4 + B_5}{(d+2)\alpha^2B_1 + (\alpha^2B_2 + B_3) + (d+8)\alpha^2 n + n\sigma^2(\alpha^4(d+2) + 2\alpha^2+1)/\gamma^2} \\
&= \frac{\alpha^4(d+2)(d+4) + 2\alpha^2(d+2) + 1}{\alpha^6 n(d+2)(d+4)(d+6) + \alpha^4(d+2)(d+4)(3n+4) + \alpha^2((d+2)(3n+d+3) + (d+8)) + (d+n+1) + \sigma^2(\alpha^4(d+2) + 2\alpha^2+1)/\gamma^2}.
\end{aligned}
$$

Then the optimal loss is obtained by setting $\boldsymbol{W}_\star = c\boldsymbol{I}$ and

$$
\mathcal{L}_\star = \mathcal{L}(\boldsymbol{W}_\star) = \gamma^2 d(\alpha^2(d+2)+1) + \sigma^2 - \gamma^2(\alpha^2B_4 + B_5)cd.
$$

It completes the proof of (D.29). Now if assuming $\alpha = \mathcal{O}(1/\sqrt{d}), d/n = \mathcal{O}(1)$, $\gamma^2 = 1/(\alpha^2 d +$

1) and sufficiently large dimension d, we have the approximate

$$\begin{aligned} c &\approx \frac{\alpha^4 d^2 + 2\alpha^2 d + 1}{n\alpha^6 d^3 + 3n\alpha^4 d^2 + (3n+d)\alpha^2 d + d + n + \sigma^2(\alpha^4 d + 2\alpha^2 + 1)/\gamma^2} \\ &\approx \frac{(\alpha^2 d + 1)^2}{n(\alpha^2 d + 1)^3 + d(\alpha^2 d + 1) + \sigma^2(\alpha^2 d + 1)} \\ &\approx \frac{1}{(\alpha^2 d + 1)n + (d + \sigma^2)/(\alpha^2 d + 1)} \end{aligned}$$

and

$$\begin{aligned} \mathcal{L}_\star &\approx \gamma^2 d(\alpha^2 d + 1) + \sigma^2 - \frac{\gamma^2(\alpha^2 d + 1)^2 nd}{(\alpha^2 d + 1)n + (d + \sigma^2)/(\alpha^2 d + 1)} \\ &= d + \sigma^2 - \frac{(\alpha^2 d + 1)nd}{(\alpha^2 d + 1)n + (d + \sigma^2)/(\alpha^2 d + 1)}. \end{aligned}$$

■

D.3 Analysis of Low-Rank Parameterization

D.3.1 Proof of Lemma 5.3

Proof. Recall the loss function from (D.21)

$$\mathcal{L}(\boldsymbol{W}) = M - 2n\mathtt{tr}\left(\boldsymbol{\Sigma}\bar{\boldsymbol{W}}\right) + n(n+1)\mathtt{tr}\left(\boldsymbol{\Sigma}\bar{\boldsymbol{W}}^\top\bar{\boldsymbol{W}}\right) + nM\mathtt{tr}\left(\bar{\boldsymbol{W}}\bar{\boldsymbol{W}}^\top\right)$$

where $\bar{\boldsymbol{W}} = \boldsymbol{\Sigma}_{\boldsymbol{x}}^{1/2}\boldsymbol{W}\boldsymbol{\Sigma}_{\boldsymbol{x}}^{1/2}$, $\boldsymbol{\Sigma} = \boldsymbol{\Sigma}_{\boldsymbol{x}}^{1/2}\boldsymbol{\Sigma}_{\boldsymbol{\beta}}\boldsymbol{\Sigma}_{\boldsymbol{x}}^{1/2}$ and $M = \mathtt{tr}\left(\boldsymbol{\Sigma}\right) + \sigma^2$. For any $\bar{\boldsymbol{W}}$, let us parameterize $\bar{\boldsymbol{W}} = \boldsymbol{U}\boldsymbol{E}\boldsymbol{U}^\top$ where $\boldsymbol{U} \in \mathbb{R}^{d\times r}$ denotes the eigenvectors of $\bar{\boldsymbol{W}}$ and $\boldsymbol{E} \in \mathbb{R}^{r\times r}$ is a symmetric square matrix. We will first treat $\boldsymbol{U}$ as fixed and optimize $\boldsymbol{E}$. We will then optimize $\boldsymbol{U}$. Fixing $\boldsymbol{U}$, setting $\bar{\boldsymbol{\Sigma}} = \boldsymbol{U}^\top\boldsymbol{\Sigma}\boldsymbol{U}$, we obtain

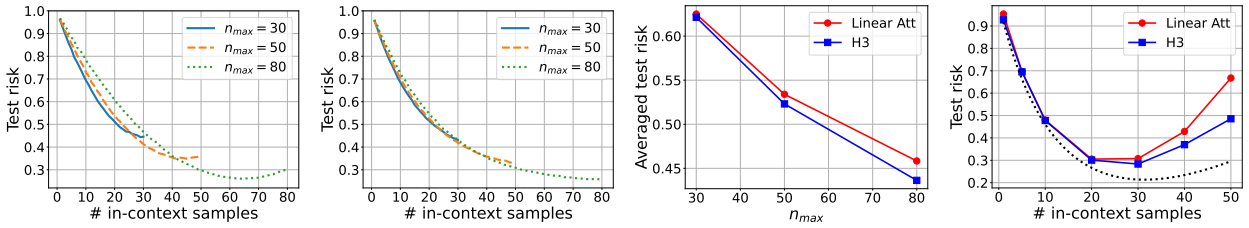
$$\mathcal{L}(\boldsymbol{E}) = M - 2n\mathtt{tr}\left(\bar{\boldsymbol{\Sigma}}\boldsymbol{E}\right) + n(n+1)\mathtt{tr}\left(\bar{\boldsymbol{\Sigma}}\boldsymbol{E}^2\right) + nM\mathtt{tr}\left(\boldsymbol{E}^2\right).$$

Differentiating, we obtain

$$0.5n^{-1}\nabla\mathcal{L}(\boldsymbol{E}) = -\bar{\boldsymbol{\Sigma}} + (n+1)\bar{\boldsymbol{\Sigma}}\boldsymbol{E} + M\boldsymbol{E}.$$

Setting $\nabla\mathcal{L}(\boldsymbol{E}) = 0$ returns

$$\boldsymbol{E}_\star = (M\boldsymbol{I} + (n+1)\bar{\boldsymbol{\Sigma}})^{-1}\bar{\boldsymbol{\Sigma}}. \tag{D.38}$$



(a) Linear attention (b) H3 (c) Averaged risk (d) Evolving $\boldsymbol{\beta}$

Figure D.1: Further comparison for linear attention and H3. In **(a)** and **(b)**, given maximum context lengths $n_{\max}$, we train linear attention and H3 models to minimize the average loss across all positions n from 1 to $n_{\max}$. Averaged test risks are presented in **(c)**. In **(d)**, the task vector $\boldsymbol{\beta}$ evolves gradually over the context positions $i \leq n$ via $\boldsymbol{\beta}_i = (i/n)\boldsymbol{\beta}_1 + (1-i/n)\boldsymbol{\beta}_2$. In both scenarios, H3 outperforms linear attention benefiting from its additional convolutional filter (c.f. $\boldsymbol{f}$ in (5.2b)). Implementation details are discussed in Section 5.4.

Let $\bar{\lambda}_i$ denote the i'th largest eigenvalue of $\bar{\boldsymbol{\Sigma}}$. Plugging in this value, we obtain the optimal risk as a function of $\boldsymbol{U}$ is given by

$$\mathcal{L}_\star(\boldsymbol{U}) = M - n \cdot \mathtt{tr}\left(\bar{\boldsymbol{\Sigma}}\boldsymbol{E}_\star\right) = M - n \cdot \mathtt{tr}\left((M\boldsymbol{I} + (n+1)\bar{\boldsymbol{\Sigma}})^{-1}\bar{\boldsymbol{\Sigma}}^2\right) \tag{D.39}$$

$$= M - n\sum_{i=1}^{r} \frac{\bar{\lambda}_i^2}{(n+1)\bar{\lambda}_i + M} = M - n\sum_{i=1}^{r} \frac{\bar{\lambda}_i}{n+1+M\bar{\lambda}_i^{-1}}. \tag{D.40}$$

Now observe that, the right hand side is strictly decreasing function of the eigenvalues $\bar{\lambda}_i$ of $\bar{\boldsymbol{\Sigma}} = \boldsymbol{U}^\top\boldsymbol{\Sigma}\boldsymbol{U}$. Thus, to minimize $\mathcal{L}_\star(\boldsymbol{U})$, we need to maximize $\sum_{i=1}^r \frac{\bar{\lambda}_i}{n+1+M\bar{\lambda}_i^{-1}}$. It follows from Cauchy interlacing theorem that $\bar{\lambda}_j \leq \lambda_i$ where λ_i is the i'th largest eigenvalue of $\boldsymbol{\Sigma}$ since $\bar{\boldsymbol{\Sigma}}$ is an orthogonal projection of $\boldsymbol{\Sigma}$ on $\boldsymbol{U}$. Consequently, we find the desired bound where

$$\mathcal{L}_\star = M - n\sum_{i=1}^{r} \frac{\lambda_i}{n+1+M\lambda_i^{-1}}.$$

The equality holds by setting $\boldsymbol{U}$ to be the top-r eigenvectors of $\boldsymbol{\Sigma}$ and $\boldsymbol{E} = \boldsymbol{E}_\star(\boldsymbol{U})$ to be the diagonal matrix according to (D.38). ■

D.4 Additional Experiments

In this section, we present additional experiments demonstrating that the H3 model can outperform the linear attention model under different training or data settings. The implementation details are consistent with those outlined in Section 5.4.

• **H3 outperforms linear attention (Figure D.1).** Until now, our analysis has estab-

lished the equivalence between linear attention and H3 models in solving linear ICL problem. Furthermore, we also investigate settings where H3 could outperform linear attention due to its sample weighting ability. In Figs. D.1a and D.1b, instead of training separate models to fit the different context lengths, we train a single model with fixed max-length $n_{\max}$ and loss is evaluated as the average loss given samples from 1 to $n_{\max}$. Such setting has been wildly studied in the previous ICL work [59, 4, 113]. We generate data according to (5.7) with $\boldsymbol{\Sigma}_{\boldsymbol{x}} = \boldsymbol{\Sigma}_{\boldsymbol{\beta}} = \boldsymbol{I}_d$ and $\sigma = 0$, and train 1-layer linear attention (Fig. D.1a) and H3 (Fig. D.1b) models with different max-lengths $n_{\max} = 30, 50, 80$. Comparison between Fig. D.1a and D.1b shows that 1-layer attention and H3 implement different algorithms in solving the averaged linear regression problem and H3 is more consistent in generalizing to longer context lengths. In Fig. D.1c, we plot the averaged risks for each model and H3 outperforms linear attention. Furthermore, in Fig. D.1d, we focus on the setting where in-context examples are generated using evolving task vector $\boldsymbol{\beta}$. Specifically, consider that each sequence corresponds to two individual task parameters $\boldsymbol{\beta}^{(1)} \sim \mathcal{N}(0, \boldsymbol{I}_d)$ and $\boldsymbol{\beta}^{(2)} \sim \mathcal{N}(0, \boldsymbol{I}_d)$. Then the i'th sample is generated via $\boldsymbol{x}_i \sim \mathcal{N}(0, \boldsymbol{I}_d)$ and $y_i = \boldsymbol{\beta}_i^\top \boldsymbol{x}_i$ where $\boldsymbol{\beta}_i = \lambda_i \boldsymbol{\beta}^{(1)} + (1 - \lambda_i)\boldsymbol{\beta}^{(2)}$ and $\lambda_i = i/n$. The results are reported in Fig. D.1d which again shows that H3 achieves better performance compared to linear attention, as H3 may benefit from the additional convolutional filter (c.f. $\boldsymbol{f}$ in (5.2b)). Here, dotted curve represent the theoretical results under i.i.d. and noiseless setting, derived from Corollary 5.1.

APPENDIX E

Appendix for Chapter 6

E.1 Experimental setup

E.1.1 Implementation detail

Data generation. Consider ICL problem with input in the form of multi-task prompt as described in Section 6.5.1. In the experiments, we set $K = 2$, dimensions $d = 10$ and $p = 5$, uniform context length $n_1 = n_2 = \bar{n}$ where we have $n = K\bar{n}$, and vary $\bar{n}$ from 0 to 50. Let $(r_1, r_2) := \big(\mathbb{E}[\boldsymbol{\beta}_1^\top \boldsymbol{\beta}]/d, \mathbb{E}[\boldsymbol{\beta}_2^\top \boldsymbol{\beta}]/d\big)$ denote the correlations between in-context tasks $\boldsymbol{\beta}_1, \boldsymbol{\beta}_2$ and query task $\boldsymbol{\beta}$. We generate task vectors as follows:

$$\boldsymbol{\beta}_1, \boldsymbol{\beta}_2 \sim \mathcal{N}(0, \boldsymbol{I}_d), \quad \text{and} \quad \boldsymbol{\beta} \sim \mathcal{N}(r_1\boldsymbol{\beta}_1 + r_2\boldsymbol{\beta}_2, (1 - r_1^2 - r_2^2)\boldsymbol{I}_d).$$

Input features are randomly sampled $((\boldsymbol{x}_{ik})_{i=1}^{\bar{n}})_{k=1}^{K}, \boldsymbol{x}_{n+1} \sim \mathcal{N}(0, \boldsymbol{I}_d)$, and we have $y_{ik} = \boldsymbol{\beta}_k^\top \boldsymbol{x}_{ik}$ ($\sigma = 0$), $k \in \{1, 2\}$ and $y_{n+1} = \boldsymbol{\beta}^\top \boldsymbol{x}_{n+1}$. Additionally, delimiters $\bar{\boldsymbol{c}}_0, \cdots, \bar{\boldsymbol{c}}_K$ are randomly sampled from $\mathcal{N}(0, \boldsymbol{I}_p)$.

Implementation setting. We train 1-layer linear attention and GLA models for solving multi-prompt ICL problem as described in Section 6.5.1. For GLA model, we consider sigmoid-type gating function given by scalar gating: $G(\boldsymbol{z}) = \phi(\boldsymbol{w}_g^\top \boldsymbol{z})\mathbf{1}_{(d+p+1)\times(d+p+1)}$, or vector gating: $G(\boldsymbol{z}) = \phi(\boldsymbol{W}_g \boldsymbol{z})\mathbf{1}_{d+p+1}^\top$ where $\phi(z) = (1 + e^{-z})^{-1}$ is the activation function. Note that although the theoretical results are based on the model constructions (c.f. (6.9) and (6.23)), we do not restrict the attention weights in our implementation. We train each model for 10000 iterations with batch size 256 and Adam optimizer with learning rate 10^{-3}. Similar to the previous work [116], since our study focuses on the optimization landscape, ICL problems using linear attention/GLA models are non-convex, and experiments are implemented via gradient descent, we repeat 10 model trainings from different model initialization and

data sampling (e.g., different choice of delimiters) and results are presented as the minimal test risk among those 10 trails. Results presented have been normalized by d.

Experimental results. Based on the experimental setting, we can obtain the correlation matrix and vector following Definition 6.1

$$\boldsymbol{R} = \begin{bmatrix} \mathbf{1}_{\bar{n}}\mathbf{1}_{\bar{n}}^\top & \mathbf{0} \\ \mathbf{0} & \mathbf{1}_{\bar{n}}\mathbf{1}_{\bar{n}}^\top \end{bmatrix} \quad \text{and} \quad \boldsymbol{r} = \begin{bmatrix} r_1\mathbf{1}_{\bar{n}}^\top & r_2\mathbf{1}_{\bar{n}}^\top \end{bmatrix}^\top.$$

Then dotted curves display our theoretical results derive using $\boldsymbol{\Sigma} = \boldsymbol{I}$ and $\boldsymbol{R}, \boldsymbol{r}$ above. Specifically, in Figure 6.1, black dashed curves represent $\mathcal{L}^\star_{\text{wpgd}}$ following (6.20) and blues dashed curves represent $\mathcal{L}^\star_{\text{gla}}$ following (6.27). We consider scenarios where $(r_1, r_2) \in \{(0,1), (0.2, 0.8), (0.5, 0.5), (0.8, 0.2)\}$ and results are presented in Figures (6.1a), (6.1b), (6.1c) and (6.1d), respectively.

- `GLA-wo` achieves the worst performance among all the methods. We claim that it is due to the randomness of input tokens as discussed in Section 6.5.1. Thanks to the introduction of delimiters as described in (6.22), data and gating is decoupled and a task-dependent weighting is learnt. Hence, `GLA` is able to achieve comparable performance to the optimal one ($\mathcal{L}^\star_{\text{wpgd}}$, red dashed). Note that `GLA-wo` performs even worse than `LinAtt`. It comes from the fact the weighting induced by `GLA-wo` varies over different input prompts and it can not implement all ones weight.

- The alignments between `LinAtt` (blue solid) and blue dashed curves validate our Corollary 6.3. In Figures 6.1a, 6.1b and 6.1c, the alignments between `GLA` (red solid) and $\mathcal{L}_{\text{wpgd}}$ (black dashed) verify our Theorem 6.5, specifically, Equation 6.26. While in 6.1c and 6.1d, `GLA` achieves the same performance as `LinAtt`. It is due to the fact that `GLA` can not weight the history higher than its present. Then the equal-weighting, e.g., $\boldsymbol{\omega} = \mathbf{1}$, is the optimal weighting given such constraint. What's more, the alignment between `GLA-vector` (cyan curves) and red dashed in Figure 6.1d validates our vector gating theorem in Theorem 6.6.

E.1.2 Multi-layer experiments

In this section, we present additional experiments on multi-layer GLA models. We adopt the same experimental setup as described in Figure 6.1a and Appendix E.1, with parameters setting to $(r_1, r_2) = (0, 1)$. The results are displayed in Figure E.1, where the blue, red, and green curves correspond to the performance of one-, two-, and three-layer GLA models,

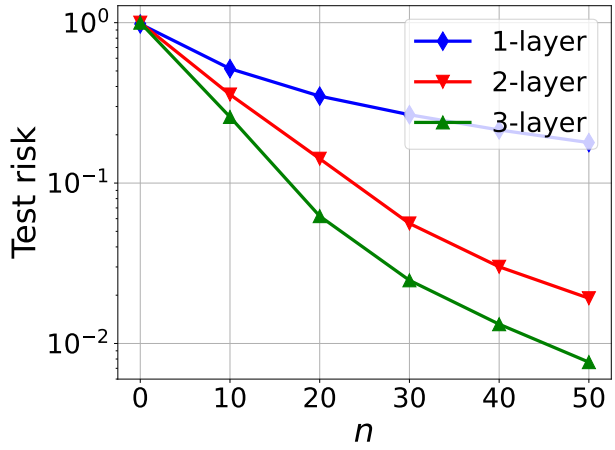


Figure E.1: Multi-layer GLA experiments with $(r_1, r_2) = (0, 1)$.

respectively, with the y-axis presented in log-scale. According to Theorem 6.2, an L-layer GLA performs L steps of WPGD, suggesting that deeper models should yield improved predictive performance. The experimental findings in Figure E.1 align with the theoretical predictions of Theorem 6.2.

E.2 GLA $\Leftrightarrow$ WPGD

E.2.1 Proof of Theroem 6.1

Proof. Recap the problem settings from Section 6.2 where in-context samples are given by

$$\boldsymbol{Z} = [\boldsymbol{z}_1 \ \cdots \ \boldsymbol{z}_n \ \boldsymbol{z}_{n+1}]^\top = \begin{bmatrix} \boldsymbol{x}_1 & \cdots & \boldsymbol{x}_n & \boldsymbol{x}_{n+1} \\ y_1 & \cdots & y_n & 0 \end{bmatrix}^\top$$

and let the value, key and query embeddings at time i be

$$\boldsymbol{v}_i = \boldsymbol{W}_v{}^\top \boldsymbol{z}_i, \quad \boldsymbol{k}_i = \boldsymbol{W}_k{}^\top \boldsymbol{z}_i, \quad \text{and} \quad \boldsymbol{q}_i = \boldsymbol{W}_q{}^\top \boldsymbol{z}_i.$$

Then we can rewrite the GLA output (cf. (GLA)) as follows:

$$\begin{aligned} \boldsymbol{o}_i = \boldsymbol{S}_i \boldsymbol{q}_i \quad \text{and} \quad \boldsymbol{S}_i &= \boldsymbol{G}_i \odot \boldsymbol{S}_{i-1} + \boldsymbol{v}_i \boldsymbol{k}_i^\top \\ &= \boldsymbol{G}_i \odot \cdots \boldsymbol{G}_1 \odot \boldsymbol{v}_1 \boldsymbol{k}_1^\top + \cdots + \boldsymbol{G}_i \odot \boldsymbol{v}_{i-1} \boldsymbol{k}_{i-1}^\top + \boldsymbol{v}_i \boldsymbol{k}_i^\top \\ &= \sum_{j=1}^{i} \boldsymbol{G}_{j:i} \odot \boldsymbol{v}_j \boldsymbol{k}_j^\top, \end{aligned}$$

where we define

$$\boldsymbol{G}_{j:i} = \boldsymbol{G}_{j+1} \odot \boldsymbol{G}_{j+2} \cdots \boldsymbol{G}_i, \quad j < i, \quad \text{and} \quad \boldsymbol{G}_{i:i} = \mathbf{1}_{(d+1)\times(d+1)}.$$

Consider the prediction based on the last token, then we obtain

$$\boldsymbol{o}_{n+1} = \boldsymbol{S}_{n+1}\boldsymbol{q}_{n+1} \quad \text{and} \quad \boldsymbol{S}_{n+1} = \sum_{j=1}^{n+1} \boldsymbol{G}_{j:n+1} \odot \boldsymbol{v}_j \boldsymbol{k}_j^\top.$$

Construction 1: Recall the model construction from (6.9) where

$$\boldsymbol{W}_k = \begin{bmatrix} \boldsymbol{P}_k & \mathbf{0} \\ \mathbf{0} & 0 \end{bmatrix}, \quad \boldsymbol{W}_q = \begin{bmatrix} \boldsymbol{P}_q & \mathbf{0} \\ \mathbf{0} & 0 \end{bmatrix} \quad \text{and} \quad \boldsymbol{W}_v = \begin{bmatrix} \mathbf{0}_{d\times d} & \mathbf{0} \\ \mathbf{0} & 1 \end{bmatrix}. \tag{E.1}$$

Then, given each token $\boldsymbol{z}_i = [\boldsymbol{x}_i^\top \; y_i]^\top$, $i \in [n]$, single-layer GLA returns

$$\boldsymbol{v}_i = \begin{bmatrix} \mathbf{0} \\ y_i \end{bmatrix}, \quad \boldsymbol{k}_i = \begin{bmatrix} \boldsymbol{P}_k^\top \boldsymbol{x}_i \\ 0 \end{bmatrix}, \quad \text{and} \quad \boldsymbol{q}_i = \begin{bmatrix} \boldsymbol{P}_q^\top \boldsymbol{x}_i \\ 0 \end{bmatrix},$$

and we obtain

$$\boldsymbol{v}_i \boldsymbol{k}_i^\top = \begin{bmatrix} \mathbf{0}_{d\times d} & \mathbf{0} \\ y_i \boldsymbol{x}_i^\top \boldsymbol{P}_k & 0 \end{bmatrix}, \quad i \le n, \quad \text{and} \quad \boldsymbol{v}_{n+1}\boldsymbol{k}_{n+1}^\top = \mathbf{0}_{(d+1)\times(d+1)}.$$

Therefore, since only d entries in $\boldsymbol{v}_i\boldsymbol{k}_i^\top$ matrix are nonzero, given $\odot$ as the Hadamard product, only the corresponding d entries in all $\boldsymbol{G}_i$ matrices are useful. Based on this observation, let

$$\boldsymbol{G}_i = \begin{bmatrix} * & * \\ \boldsymbol{g}_i^\top & * \end{bmatrix} \quad \text{and} \quad \boldsymbol{G}_{j:i} = \begin{bmatrix} * & * \\ \boldsymbol{g}_{j:i}^\top & *, \end{bmatrix}$$

where $\boldsymbol{g}_{j:i} = \boldsymbol{g}_{j+1} \odot \boldsymbol{g}_{j+2} \cdots \boldsymbol{g}_i \in \mathbb{R}^d$ for $j < i$ and $\boldsymbol{g}_{i:i} = \mathbf{1}_d$.

Combing all together, and letting $\boldsymbol{X}, \boldsymbol{y}$ follow the definitions in (6.6), we obtain

$$\begin{aligned}
\boldsymbol{o}_{n+1} &= \boldsymbol{S}_{n+1}\boldsymbol{q}_{n+1} \\
&= \left(\sum_{j=1}^{n+1} \boldsymbol{G}_{j:n+1} \odot \boldsymbol{v}_j \boldsymbol{k}_j^\top\right) \boldsymbol{q}_{n+1} \\
&= \begin{bmatrix} \boldsymbol{0}_{d\times d} & \boldsymbol{0} \\ \sum_{j=1}^{n} y_j \boldsymbol{x}_j^\top \boldsymbol{P}_k \odot \boldsymbol{g}_{j:n+1}^\top & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{P}_q^\top \boldsymbol{x} \\ 0 \end{bmatrix} \\
&= \begin{bmatrix} \boldsymbol{0} \\ \boldsymbol{x}^\top \boldsymbol{P}_q \left(\boldsymbol{X}\boldsymbol{P}_k \odot \boldsymbol{\Omega}\right)^\top \boldsymbol{y} \end{bmatrix}
\end{aligned}$$

where

$$\boldsymbol{\Omega} = \begin{bmatrix} \boldsymbol{g}_{1:n+1} & \boldsymbol{g}_{2:n+1} & \cdots & \boldsymbol{g}_{n:n+1} \end{bmatrix} \in \mathbb{R}^{n\times d}.$$

Then if taking the last entry of $\boldsymbol{o}_{n+1}$ as final prediction, we get

$$f_{\text{gla}}(\boldsymbol{Z}) = \boldsymbol{x}^\top \boldsymbol{P}_q \left(\boldsymbol{X}\boldsymbol{P}_k \odot \boldsymbol{\Omega}\right)^\top \boldsymbol{y}$$

which completes the proof of Theorem 6.1. ∎

Construction 2: Based on the construction given in (E.1), only d elements of $\boldsymbol{G}_i$ matrices are useful. One might ask about the effect of other entries of $\boldsymbol{G}_i$. Therefore, in the following, we introduce an other model construction showing that different row of $\boldsymbol{G}_i$ implements WPGD with different weighting. Similarly, let $\boldsymbol{W}_k, \boldsymbol{W}_q$ be the same as (E.1) but with $\boldsymbol{W}_v$ constructed by

$$\boldsymbol{W}_v = \begin{bmatrix} \boldsymbol{0}_{(d+1)\times d} & \boldsymbol{u} \end{bmatrix}^\top \quad \text{where} \quad \boldsymbol{u} = [u_1 \ u_2 \ \cdots \ u_{d+1}]^\top \in \mathbb{R}^{d+1}.$$

Then, the value embeddings have the form of $\boldsymbol{v}_i = y_i \boldsymbol{u}$, which gives

$$\boldsymbol{v}_i \boldsymbol{k}_i^\top = \boldsymbol{u} \begin{bmatrix} y_i \boldsymbol{x}_i^\top \boldsymbol{P}_k & \boldsymbol{0} \end{bmatrix}.$$

Next, let

$$\boldsymbol{G}_i = \begin{bmatrix} (\boldsymbol{g}_i^1)^\top & * \\ (\boldsymbol{g}_i^2)^\top & * \\ \vdots & \vdots \\ (\boldsymbol{g}_i^{d+1})^\top & * \end{bmatrix}, \quad \text{and} \quad \boldsymbol{G}_{j:i} = \begin{bmatrix} (\boldsymbol{g}_{j:i}^1)^\top & * \\ (\boldsymbol{g}_{j:i}^2)^\top & * \\ \vdots & \vdots \\ (\boldsymbol{g}_{j:i}^{d+1})^\top & * \end{bmatrix},$$

where $\boldsymbol{g}_i^{i'} \in \mathbb{R}^d$ corresponds to the i'-th row of $\boldsymbol{G}_i$ and $\boldsymbol{g}_{j:i}^{i'} = \boldsymbol{g}_{j+1}^{i'} \odot \boldsymbol{g}_{j+1}^{i'} \cdots \boldsymbol{g}_i^{i'}$. Then we get the output

$$\boldsymbol{o}_{n+1} = \begin{bmatrix} \sum_{j=1}^n u_1 y_j \boldsymbol{x}_j^\top \boldsymbol{P}_k \odot (\boldsymbol{g}_{j:n+1}^1)^\top & 0 \\ \sum_{j=1}^n u_2 y_j \boldsymbol{x}_j^\top \boldsymbol{P}_k \odot (\boldsymbol{g}_{j:n+1}^2)^\top & 0 \\ \vdots & \\ \sum_{j=1}^n u_{d+1} y_j \boldsymbol{x}_j^\top \boldsymbol{P}_k \odot (\boldsymbol{g}_{j:n+1}^{d+1})^\top & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{P}_q^\top \boldsymbol{x} \\ 0 \end{bmatrix} = \begin{bmatrix} \boldsymbol{x}^\top \boldsymbol{P}_q \left(\boldsymbol{X}\boldsymbol{P}_k \odot \boldsymbol{\Omega}_1\right)^\top \boldsymbol{y} \\ \boldsymbol{x}^\top \boldsymbol{P}_q \left(\boldsymbol{X}\boldsymbol{P}_k \odot \boldsymbol{\Omega}_2\right)^\top \boldsymbol{y} \\ \vdots \\ \boldsymbol{x}^\top \boldsymbol{P}_q \left(\boldsymbol{X}\boldsymbol{P}_k \odot \boldsymbol{\Omega}_{d+1}\right)^\top \boldsymbol{y} \end{bmatrix},$$

where

$$\boldsymbol{\Omega}_i = u_i \begin{bmatrix} \boldsymbol{g}_{1:n+1}^i & \boldsymbol{g}_{2:n+1}^i & \cdots & \boldsymbol{g}_{n:n+1}^i \end{bmatrix} \in \mathbb{R}^{n \times d}, \quad i \le d+1. \tag{E.2}$$

Therefore, consider $(d+1)$-dimensional output $\boldsymbol{o}_{n+1}$. Each entry implements a 1-step WPGD with same preconditioners $\boldsymbol{P}_k, \boldsymbol{P}_q$ and different weighting matrices $\boldsymbol{\Omega}$'s. The weighting matrix of i'th entry is determined by the i'th row of all gating matrices. Note that if consider the last entry of $\boldsymbol{o}_{n+1}$ as prediction, it returns the same result as **Construction 1** above, where only last rows of $\boldsymbol{G}_i$'s are useful.

Additionally, suppose that the final prediction is given after a linear head $\boldsymbol{h}$, that is, $f_{\text{gla}}(\boldsymbol{Z}) = \boldsymbol{h}^\top \boldsymbol{o}_{n+1}$, and let $\boldsymbol{h} = [h_1 \ h_2 \ \cdots \ h_{d+1}]^\top \in \mathbb{R}^{d+1}$. Then

$$f_{\text{gla}}(\boldsymbol{Z}) = \boldsymbol{h}^\top \boldsymbol{o}_{n+1} = \boldsymbol{x}^\top \boldsymbol{P}_q \left(\boldsymbol{X}\boldsymbol{P}_k \odot \bar{\boldsymbol{\Omega}}\right)^\top \boldsymbol{y} \tag{E.3}$$

where

$$\bar{\boldsymbol{\Omega}} = \sum_{i=1}^{d+1} h_i \boldsymbol{\Omega}_i = \sum_{i=1}^{d+1} h_i u_i \begin{bmatrix} \boldsymbol{g}_{1:n+1}^i & \boldsymbol{g}_{2:n+1}^i & \cdots & \boldsymbol{g}_{n:n+1}^i \end{bmatrix} \in \mathbb{R}^{n \times d}. \tag{E.4}$$

Then, single-layer GLA still returns one-step WPGD with updated weighting matrix.

E.2.2 Proof of Theorem 6.2

In this section, we first prove the following theorem, which differs from Theorem 6.2 only in that $\boldsymbol{W}_{q,\ell}$ is parameterized by $\boldsymbol{P}_{q,\ell}$ instead of $-\boldsymbol{P}_{q,\ell}$. Theorem 6.2 can then be directly obtained by setting $\boldsymbol{P}_{q,\ell}$ to $-\boldsymbol{P}_{q,\ell}$.

Theorem E.1 *Consider an L-layer GLA with residual connections, where each layer follows the weight construction specified in* (6.9)*. Let $\boldsymbol{W}_{k,\ell}$ and $\boldsymbol{W}_{q,\ell}$ in the ℓ'th layer parameterized by $\boldsymbol{P}_{k,\ell}, \boldsymbol{P}_{q,\ell} \in \mathbb{R}^{d \times d}$, for $\ell \in [L]$. Let the gating be a function of the features, e.g., $\boldsymbol{G}_i = G(\boldsymbol{x}_i)$, and define $\boldsymbol{\Omega}$ as in Theorem 6.1. Additionally, denote the masking as $\boldsymbol{M}_i = \begin{bmatrix} \boldsymbol{I}_i & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix} \in \mathbb{R}^{n \times n}$,*

and let $\hat{\boldsymbol{\beta}}_0 = \bigotimes i0 = \mathbf{0}$ *for all* $i \in [n]$.

Then, the last entry of the i*'th token output at the* ℓ*'th layer* ($o_{i,\ell}$) *is:*

- *For all* $i \le n$, $o_{i,\ell} = y_i - \boldsymbol{x}_i^\top \boldsymbol{\beta}_{i,\ell}$, *where* $\boldsymbol{\beta}_{i,\ell} = \boldsymbol{\beta}_{i,\ell-1} + \boldsymbol{P}_{q,\ell}\left(\nabla_{i,\ell} \oslash \boldsymbol{g}_{i:n+1}\right)$,
- $o_{n+1,\ell} = -\boldsymbol{x}^\top \hat{\boldsymbol{\beta}}_\ell$, *where* $\hat{\boldsymbol{\beta}}_\ell = (1+\alpha_\ell)\hat{\boldsymbol{\beta}}_{\ell-1} + \boldsymbol{P}_{q,\ell}\left(\nabla_{n,\ell} \oslash \boldsymbol{g}_{n+1}\right)$, *and* $\alpha_\ell = \boldsymbol{x}^\top \boldsymbol{P}_{q,\ell}\boldsymbol{P}_{k,\ell}^\top \boldsymbol{x}$.

Here, letting $\boldsymbol{B}_\ell = [\bigotimes 1\ell \ \cdots \ \bigotimes n\ell]^\top$, $\boldsymbol{X}_\ell = \boldsymbol{X}\boldsymbol{P}_{k,\ell} \odot \boldsymbol{\Omega}$, *and* $\boldsymbol{y}_\ell = diag(\boldsymbol{X}\boldsymbol{B}_\ell^\top)$, *we define*

$$\nabla_{i,\ell} = \boldsymbol{X}_\ell^\top \boldsymbol{M}_i \left(\boldsymbol{y}_\ell - \boldsymbol{y}\right).$$

Proof. Suppose that the input prompt for ℓ'th layer is

$$\boldsymbol{Z}_\ell = \begin{bmatrix} \tilde{\boldsymbol{x}}_1 & \dots & \tilde{\boldsymbol{x}}_n & \tilde{\boldsymbol{x}}_{n+1} \\ \tilde{y}_1 & \dots & \tilde{y}_n & \tilde{y}_{n+1} \end{bmatrix}^\top \in \mathbb{R}^{(n+1)\times(d+1)} \tag{E.5}$$

and let $\boldsymbol{k}_{i,\ell}, \boldsymbol{q}_{i,\ell}, \boldsymbol{v}_{i,\ell}$ be their corresponding key, query and value embeddings. Recap the model construction from (6.9) and follow the same analysis in Appendix E.2.1. Let $\boldsymbol{S}_i^\ell$, $i \in [n+1]$ be the corresponding states. We have

$$\begin{aligned} \boldsymbol{S}_i^\ell &= \sum_{j=1}^{i} \boldsymbol{G}_{j:i}^\ell \odot \boldsymbol{v}_{j,\ell}\boldsymbol{k}_{j,\ell}^\top \\ &= \begin{bmatrix} \mathbf{0}_{d\times d} & \mathbf{0} \\ \sum_{j=1}^{i} \tilde{y}_j \tilde{\boldsymbol{x}}_j^\top \boldsymbol{P}_{k,\ell} \odot (\boldsymbol{g}_{j:i}^\ell)^\top & 0 \end{bmatrix} \end{aligned}$$

where

$$\boldsymbol{G}_i^\ell = G(\tilde{\boldsymbol{x}}_i) = \begin{bmatrix} * & * \\ (\boldsymbol{g}_i^\ell)^\top & * \end{bmatrix} \quad \text{and} \quad \boldsymbol{G}_{j:i}^\ell = \begin{bmatrix} * & * \\ (\boldsymbol{g}_{j:i}^\ell)^\top & * \end{bmatrix}.$$

Additionally, recap that we have $\boldsymbol{M}_i = \begin{bmatrix} \boldsymbol{I}_i & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}$ and $\oslash$ denotes Hadamard division. Let

$$\boldsymbol{\Omega}^\ell = \begin{bmatrix} \boldsymbol{g}_{i:n+1}^\ell & \cdots & \boldsymbol{g}_{n:n+1}^\ell \end{bmatrix}.$$

Then, defining

$$\tilde{\boldsymbol{X}} = [\tilde{\boldsymbol{x}}_1 \ \ \tilde{\boldsymbol{x}}_2 \ \ \cdots \ \ \tilde{\boldsymbol{x}}_n]^\top \in \mathbb{R}^{n\times d} \quad \text{and} \quad \tilde{\boldsymbol{y}} = [\tilde{y}_1 \ \ \tilde{y}_2 \ \ \cdots \ \ \tilde{y}_n]^\top \in \mathbb{R}^n,$$

and letting $\boldsymbol{o}_{i,\ell}$, $i \in [n+1]$ be their outputs, we get:

- For $i \leq n$,

$$\sum_{j=1}^{i} \tilde{y}_j \boldsymbol{P}_{k,\ell}^\top \tilde{\boldsymbol{x}}_j \odot \boldsymbol{g}_{j:i}^\ell = \left(\sum_{j=1}^{i} \tilde{y}_j \boldsymbol{P}_{k,\ell}^\top \tilde{\boldsymbol{x}}_j \odot \boldsymbol{g}_{j:n+1}^\ell \right) \oslash \boldsymbol{g}_{i:n+1}^\ell$$
$$= (\tilde{\boldsymbol{X}} \boldsymbol{P}_{k,\ell} \odot \boldsymbol{\Omega}^\ell)^\top \boldsymbol{M}_i \tilde{\boldsymbol{y}} \oslash \boldsymbol{g}_{i:n+1}^\ell,$$

and

$$\boldsymbol{o}_{i,\ell} = \boldsymbol{S}_i^\ell \boldsymbol{q}_{i,\ell} = \begin{bmatrix} \boldsymbol{0}_{d\times d} & \boldsymbol{0} \\ \left((\tilde{\boldsymbol{X}} \boldsymbol{P}_{k,\ell} \odot \boldsymbol{\Omega}^\ell)^\top \boldsymbol{M}_i \tilde{\boldsymbol{y}} \oslash \boldsymbol{g}_{i:n+1}^\ell \right)^\top & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{P}_{q,\ell}^\top \tilde{\boldsymbol{x}}_i \\ 0 \end{bmatrix}$$
$$= \begin{bmatrix} \boldsymbol{0} \\ \tilde{\boldsymbol{x}}_i^\top \boldsymbol{P}_{q,\ell} \left((\tilde{\boldsymbol{X}} \boldsymbol{P}_{k,\ell} \odot \boldsymbol{\Omega}^\ell)^\top \boldsymbol{M}_i \tilde{\boldsymbol{y}} \oslash \boldsymbol{g}_{i:n+1}^\ell \right) \end{bmatrix}. \tag{E.6}$$

- For $i = n+1$,

$$\sum_{j=1}^{n+1} \tilde{y}_j \boldsymbol{P}_{k,\ell}^\top \tilde{\boldsymbol{x}}_j \odot \boldsymbol{g}_{j:i}^\ell = (\tilde{\boldsymbol{X}} \boldsymbol{P}_{k,\ell} \odot \boldsymbol{\Omega}^\ell)^\top \tilde{\boldsymbol{y}} + \tilde{y}_{n+1} \boldsymbol{P}_{k,\ell}^\top \tilde{\boldsymbol{x}}_{n+1},$$

and

$$\boldsymbol{o}_{n+1,\ell} = \begin{bmatrix} \boldsymbol{0}_{d\times d} & \boldsymbol{0} \\ \left((\tilde{\boldsymbol{X}} \boldsymbol{P}_{k,\ell} \odot \boldsymbol{\Omega}^\ell)^\top \tilde{\boldsymbol{y}} + \tilde{y}_{n+1} \boldsymbol{P}_{k,\ell}^\top \tilde{\boldsymbol{x}}_{n+1} \right)^\top & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{P}_{q,\ell}^\top \tilde{\boldsymbol{x}}_{n+1} \\ 0 \end{bmatrix}$$
$$= \begin{bmatrix} \boldsymbol{0} \\ \tilde{\boldsymbol{x}}_{n+1}^\top \boldsymbol{P}_{q,\ell} \left((\tilde{\boldsymbol{X}} \boldsymbol{P}_{k,\ell} \odot \boldsymbol{\Omega}^\ell)^\top \tilde{\boldsymbol{y}} + \tilde{y}_{n+1} \boldsymbol{P}_{k,\ell}^\top \tilde{\boldsymbol{x}}_{n+1} \right) \end{bmatrix}$$
$$= \begin{bmatrix} \boldsymbol{0} \\ \tilde{y}_{n+1} \tilde{\boldsymbol{x}}_{n+1}^\top \boldsymbol{P}_{q,\ell} \boldsymbol{P}_{k,\ell}^\top \tilde{\boldsymbol{x}}_{n+1} + \tilde{\boldsymbol{x}}_{n+1}^\top \boldsymbol{P}_{q,\ell} \left((\tilde{\boldsymbol{X}} \boldsymbol{P}_{k,\ell} \odot \boldsymbol{\Omega}^\ell)^\top \tilde{\boldsymbol{y}} \right). \end{bmatrix} \tag{E.7}$$

From (E.6) and (E.7), the first d entries of all the $(d+1)$-dimensional outputs (e.g, $\boldsymbol{o}_{i,\ell}$, $i \in [n+1]$) are zero. Given the multi-layer GLA as formulated in (6.11) where residual connections are applied, we have that the first d dimension of all the input tokens remain the same. That is, $\tilde{\boldsymbol{x}}_i \equiv \boldsymbol{x}_i$. Additionally, since gating only depends on the feature $\boldsymbol{x}_i$, then all the layers return the same gatings (e.g, $\boldsymbol{G}_i^\ell \equiv \boldsymbol{G}_i$).

Note that if the gating function varies across layers or depends on the entire token rather than just the feature $\boldsymbol{x}_i$, each layer will have its own gating, leading to $\boldsymbol{G}_i^\ell \neq \boldsymbol{G}_i$. In this theorem, we assume identical gating matrices across layers for simplicity and clarity. However, an extended version of this theorem, without this assumption, can be directly

derived.

Therefore, we obtain

$$\boldsymbol{o}_{i,\ell} = \begin{bmatrix} \mathbf{0} \\ \boldsymbol{x}_i^\top \boldsymbol{P}_{q,\ell} \left(\boldsymbol{X}_\ell^\top \boldsymbol{M}_i \tilde{\boldsymbol{y}} \oslash \boldsymbol{g}_{i:n+1} \right) \end{bmatrix}, \quad i \le n$$
$$\boldsymbol{o}_{n+1,\ell} = \begin{bmatrix} \mathbf{0} \\ \tilde{y}_{n+1} \boldsymbol{x}^\top \boldsymbol{P}_{q,\ell} \boldsymbol{P}_{k,\ell}^\top \boldsymbol{x} + \boldsymbol{x}^\top \boldsymbol{P}_{q,\ell} \left(\boldsymbol{X}_\ell^\top \tilde{\boldsymbol{y}} \oslash \boldsymbol{g}_{n+1} \right) \end{bmatrix}.$$

where we define $\boldsymbol{X}_\ell := \boldsymbol{X} \boldsymbol{P}_{k,\ell} \odot \boldsymbol{\Omega}$.

Since we focus on the last/$(d+1)$'th entry of each token output, we have

$$\begin{aligned} o_{i,\ell} &= \boldsymbol{x}_i^\top \boldsymbol{P}_{q,\ell} \left(\boldsymbol{X}_\ell^\top \boldsymbol{M}_i \tilde{\boldsymbol{y}} \oslash \boldsymbol{g}_{i:n+1} \right), \quad i \le n, \\ o_{n+1,\ell} &= \tilde{y}_{n+1} \boldsymbol{x}^\top \boldsymbol{P}_{q,\ell} \boldsymbol{P}_{k,\ell}^\top \boldsymbol{x} + \boldsymbol{x}^\top \boldsymbol{P}_{q,\ell} \left(\boldsymbol{X}_\ell^\top \tilde{\boldsymbol{y}} \oslash \boldsymbol{g}_{n+1} \right). \end{aligned} \tag{E.8}$$

It remains to get the $\tilde{y}_i$ for each layer's input. Let inputs of $(\ell-1)$'th and ℓ'th layer be

$$\boldsymbol{Z}_{\ell-1} = \begin{bmatrix} \boldsymbol{x}_1 & \dots & \boldsymbol{x}_n & \boldsymbol{x} \\ y_1^{\ell-1} & \dots & y_n^{\ell-1} & y^{\ell-1} \end{bmatrix} \quad \text{and} \quad \boldsymbol{Z}_\ell = \begin{bmatrix} \boldsymbol{x}_1 & \dots & \boldsymbol{x}_n & \boldsymbol{x} \\ y_1^{\ell} & \dots & y_n^{\ell} & y^{\ell} \end{bmatrix}.$$

Define

$$\boldsymbol{y}^{\ell-1} := [y_1^{\ell-1} \quad \cdots \quad y_n^{\ell-1}]^\top \in \mathbb{R}^n.$$

- For $i \le n$, from (E.8) and (6.11), we have

$$\begin{aligned} y_i^\ell &= y_i^{\ell-1} + \boldsymbol{x}_i^\top \boldsymbol{P}_{q,\ell} \left(\boldsymbol{X}_\ell^\top \boldsymbol{M}_i \boldsymbol{y}^{\ell-1} \oslash \boldsymbol{g}_{i:n+1} \right) \\ \Longrightarrow y_i^\ell - y_i^{\ell-1} &= \boldsymbol{x}_i^\top \boldsymbol{P}_{q,\ell} \left(\boldsymbol{X}_\ell^\top \boldsymbol{M}_i \boldsymbol{y}^{\ell-1} \oslash \boldsymbol{g}_{i:n+1} \right). \end{aligned}$$

  Suppose that $y_i - y_i^\ell = \boldsymbol{x}_i^\top \boldsymbol{\beta}_i^\ell$. Then

$$\boldsymbol{y}^\ell = \boldsymbol{y} - [\boldsymbol{x}_1^\top \boldsymbol{\beta}_1^\ell \quad \cdots \quad \boldsymbol{x}_n^\top \boldsymbol{\beta}_n^\ell]^\top = \boldsymbol{y} - \operatorname{diag}(\boldsymbol{X}\boldsymbol{B}^\top) \tag{E.9}$$

  and

$$\begin{aligned} & y_i - \boldsymbol{x}_i^\top \boldsymbol{\beta}_i^\ell - (y_i - \boldsymbol{x}_i^\top \boldsymbol{\beta}_i^{\ell-1}) = \boldsymbol{x}_i^\top \boldsymbol{P}_{q,\ell} \left(\boldsymbol{X}_\ell^\top \boldsymbol{M}_i (\boldsymbol{y} - \operatorname{diag}(\boldsymbol{X}\boldsymbol{B}^\top)) \oslash \boldsymbol{g}_{i:n+1} \right) \\ \Longrightarrow & \boldsymbol{x}_i^\top \boldsymbol{\beta}_i^\ell = \boldsymbol{x}_i^\top \boldsymbol{\beta}_i^{\ell-1} - \boldsymbol{x}_i^\top \boldsymbol{P}_{q,\ell} \left(\boldsymbol{X}_\ell^\top \boldsymbol{M}_i (\boldsymbol{y} - \operatorname{diag}(\boldsymbol{X}\boldsymbol{B}^\top)) \oslash \boldsymbol{g}_{i:n+1} \right). \end{aligned}$$

Hence,

$$y_i^\ell = y_i - \boldsymbol{x}_i^\top \boldsymbol{\beta}_i^\ell \quad \text{where} \quad \boldsymbol{\beta}_i^\ell = \boldsymbol{\beta}_i^{\ell-1} - \boldsymbol{P}_{q,\ell} \left(\boldsymbol{X}_\ell^\top \boldsymbol{M}_i (\boldsymbol{y} - \operatorname{diag}(\boldsymbol{X}\boldsymbol{B}^\top)) \oslash \boldsymbol{g}_{i:n+1} \right) \tag{E.10}$$

- For $i = n+1$, from (E.8) and (6.11), we have

$$y^\ell = y^{\ell-1} + y^{\ell-1} \boldsymbol{x}^\top \boldsymbol{P}_{q,\ell} \boldsymbol{P}_{k,\ell}^\top \boldsymbol{x} + \boldsymbol{x}^\top \boldsymbol{P}_{q,\ell} \left(\boldsymbol{X}_\ell^\top \boldsymbol{y}^{\ell-1} \oslash \boldsymbol{g}_{n+1} \right).$$

Setting $\alpha_\ell = \boldsymbol{x}^\top \boldsymbol{P}_{q,\ell} \boldsymbol{P}_{k,\ell}^\top \boldsymbol{x}$ and recalling $\boldsymbol{y}^\ell$ from (E.9), we get

$$y^\ell = (1+\alpha_\ell) y^{\ell-1} + \boldsymbol{x}^\top \boldsymbol{P}_{q,\ell} \left(\boldsymbol{X}_\ell^\top (\boldsymbol{y} - \operatorname{diag}(\boldsymbol{X}\boldsymbol{B}^\top)) \oslash \boldsymbol{g}_{n+1} \right).$$

Additionally, let $y^\ell = -\boldsymbol{x}^\top \boldsymbol{\beta}^\ell$. Then,

$$-\boldsymbol{x}^\top \boldsymbol{\beta}^\ell = -(1+\alpha_\ell) \boldsymbol{x}^\top \boldsymbol{\beta}^{\ell-1} + \boldsymbol{x}^\top \boldsymbol{P}_{q,\ell} \left(\boldsymbol{X}_\ell^\top (\boldsymbol{y} - \operatorname{diag}(\boldsymbol{X}\boldsymbol{B}^\top)) \oslash \boldsymbol{g}_{n+1} \right).$$

Hence,

$$y^\ell = -\boldsymbol{x}^\top \boldsymbol{\beta}^\ell \quad \text{where} \quad \boldsymbol{\beta}^\ell = (1+\alpha_\ell) \boldsymbol{\beta}^{\ell-1} - \boldsymbol{P}_{q,\ell} \left(\boldsymbol{X}_\ell^\top (\boldsymbol{y} - \operatorname{diag}(\boldsymbol{X}\boldsymbol{B}^\top)) \oslash \boldsymbol{g}_{n+1} \right) \tag{E.11}$$

Combining (E.10) and (E.11) together completes the proof. ■

E.3 Optimization landscape of WPGD

We first provide the following Lemma.

Lemma E.1 (Existence of Fixed Point) *Consider the functions $h_1(\bar{\gamma})$ and $h_2(\gamma)$ defined in* (6.15a) *and* (6.15b)*, respectively. The composite function $h_1(h_2(\gamma)) + 1$ has at least one fixed point $\gamma^\star \geq 1$, i.e., there exists $\gamma^\star \geq 1$ such that $h_1(h_2(\gamma^\star)) + 1 = \gamma^\star$.*

Proof. We prove existence using continuity and boundedness arguments along with the intermediate value theorem. Note that $h_1(\bar{\gamma})$ and $h_2(\gamma)$ are continuous functions. Thus, their composition $f(\gamma) := h_1(h_2(\gamma)) + 1$ is continuous for all $\gamma \geq 1$.

We examine the behavior of $f(\gamma)$ at the boundaries of its domain: At $\gamma = 1$, we have $h_2(1) = c_0 > 0$, where c_0 is a positive constant. Therefore, $f(1) = h_1(h_2(1)) + 1 = h_1(c_0) + 1 >$

0, which is finite and positive since h_1 is positive for all non-negative inputs. As $\gamma \to \infty$, $h_2(\gamma) \to c_\infty > 0$, where c_∞ is a positive constant.

For sufficiently large γ, we can establish that $f(\gamma) < \gamma$. To see this, note that we have shown that $h_2(\gamma) \to c_\infty$ as $\gamma \to \infty$, where c_∞ is a fixed positive constant. Further, for any fixed positive value $\bar{\gamma}$, the function $h_1(\bar{\gamma})$ is bounded, i.e., $h_1(\bar{\gamma}) \leq \max_{1 \leq i \leq n} \lambda_i$, which follows from the fact that $h_1(\bar{\gamma})$ is a weighted average of the λ_i values. Therefore, there exists a constant K such that $f(\gamma) \leq K$ for all $\gamma \geq 1$. Hence, it follows that for all $\gamma > K$, we have $f(\gamma) \leq K < \gamma$.

Now, consider the function $\bar{f}(\gamma) = f(\gamma) - \gamma$, which is continuous on $[1, \infty)$ since both $f(\gamma)$ and γ are continuous. We have $\bar{f}(1) = f(1) - 1 = h_1(c_0) > 0$ and for some $\gamma_1 > K$, we have $\bar{f}(\gamma_1) = f(\gamma_1) - \gamma_1 < 0$. Since $\bar{f}$ is continuous and changes sign from positive to negative on the interval $[1, \gamma_1]$, by the Intermediate Value Theorem, there exists at least one $\gamma^\star \in (1, \gamma_1)$ such that $\bar{f}(\gamma^\star) = 0$. This implies $f(\gamma^\star) - \gamma^\star = 0$, or equivalently, $f(\gamma^\star) = \gamma^\star$. Therefore, there exists at least one fixed point $\gamma^\star \geq 1$ for the function $h_1(h_2(\gamma)) + 1$. ■

E.3.1 Proof of Theorem 6.3

Proof. Recapping the objective from (6.2) and following Definitions 6.1 and 6.2, we have

$$\begin{aligned}
\mathcal{L}(\boldsymbol{P}, \boldsymbol{\omega}) &= \mathbb{E}\left[\left(y - \boldsymbol{x}^\top \boldsymbol{P} \boldsymbol{X}(\boldsymbol{\omega} \odot \boldsymbol{y})\right)^2\right] \\
&= \mathbb{E}\left[y^2\right] - 2\,\mathbb{E}\left[y \boldsymbol{x}^\top \boldsymbol{P} \boldsymbol{X}(\boldsymbol{\omega} \odot \boldsymbol{y})\right] + \mathbb{E}\left[\left(\boldsymbol{x}^\top \boldsymbol{P} \boldsymbol{X}(\boldsymbol{\omega} \odot \boldsymbol{y})\right)^2\right].
\end{aligned}$$

Let $y = \boldsymbol{x}^\top \boldsymbol{\beta} + \xi$ and $y_i = \boldsymbol{x}_i^\top \boldsymbol{\beta}_i + \xi_i$, for all $i \in [n]$, where $\xi, \xi_i \sim \mathcal{N}(0, \sigma^2)$ are i.i.d. Then,

$$\mathbb{E}[y^2] = \mathbb{E}[(\boldsymbol{x}^\top \boldsymbol{\beta} + \xi)^2] = \mathtt{tr}\left(\boldsymbol{\Sigma}\right) + \sigma^2,$$

and

$$\begin{aligned}
\mathbb{E}\left[y \boldsymbol{x}^\top \boldsymbol{P} \boldsymbol{X}(\boldsymbol{\omega} \odot \boldsymbol{y})\right] &= \mathbb{E}\left[(\boldsymbol{\beta}^\top \boldsymbol{x} + \xi) \boldsymbol{x}^\top \boldsymbol{P} \sum_{i=1}^n \omega_i \boldsymbol{x}_i (\boldsymbol{x}_i^\top \boldsymbol{\beta}_i + \xi_i)\right] \\
&= \mathbb{E}\left[\boldsymbol{\beta}^\top \boldsymbol{x} \boldsymbol{x}^\top \boldsymbol{P} \sum_{i=1}^n \omega_i \boldsymbol{x}_i \boldsymbol{x}_i^\top \boldsymbol{\beta}_i\right] \\
&= \mathtt{tr}\left(\boldsymbol{\Sigma} \boldsymbol{P} \boldsymbol{\Sigma} \sum_{i=1}^n \omega_i\, \mathbb{E}\left[\boldsymbol{\beta}_i \boldsymbol{\beta}^\top\right]\right) \\
&= \mathtt{tr}\left(\boldsymbol{\Sigma}^2 \boldsymbol{P}\right) \boldsymbol{\omega}^\top \boldsymbol{r}.
\end{aligned}$$

Here, the last equality comes from the fact that since $\boldsymbol{\beta}_i - r_{ij}\boldsymbol{\beta}_j$ is independent of $\boldsymbol{\beta}_j$ for $i, j \in [n+1]$ following Definition 6.1, we have $\mathbb{E}\left[\boldsymbol{\beta}_i\boldsymbol{\beta}^\top\right] = r_{i,n+1}\boldsymbol{I}_d$ and $\sum_{i=1}^n \omega_i \mathbb{E}\left[\boldsymbol{\beta}_i\boldsymbol{\beta}^\top\right]$ returns $\boldsymbol{\omega}^\top\boldsymbol{r}\cdot\boldsymbol{I}_d$.

Hence,

$$
\begin{aligned}
&\mathbb{E}\left[\left(\boldsymbol{x}^\top\boldsymbol{P}\boldsymbol{X}(\boldsymbol{\omega}\odot\boldsymbol{y})\right)^2\right] \\
&= \mathbb{E}\left[\boldsymbol{x}^\top\boldsymbol{P}\left(\sum_{i=1}^n \omega_i(\boldsymbol{x}_i^\top\boldsymbol{\beta}_i+\xi_i)\boldsymbol{x}_i\right)\left(\sum_{i=1}^n \omega_i\boldsymbol{x}_i^\top(\boldsymbol{x}_i^\top\boldsymbol{\beta}_i+\xi_i)\right)\boldsymbol{P}^\top\boldsymbol{x}\right] \\
&= \mathtt{tr}\left(\boldsymbol{P}^\top\boldsymbol{\Sigma}\boldsymbol{P}\,\mathbb{E}\left[\sum_{i=1}^n \omega_i^2(\boldsymbol{x}_i^\top\boldsymbol{\beta}_i+\xi_i)^2\boldsymbol{x}_i\boldsymbol{x}_i^\top\right]\right) \\
&+ \mathtt{tr}\left(\boldsymbol{P}^\top\boldsymbol{\Sigma}\boldsymbol{P}\,\mathbb{E}\left[\sum_{i\neq j} \omega_i\omega_j(\boldsymbol{x}_i^\top\boldsymbol{\beta}_i+\xi_i)\boldsymbol{x}_i\boldsymbol{x}_j^\top(\boldsymbol{x}_j^\top\boldsymbol{\beta}_j+\xi_j)\right]\right),
\end{aligned}
$$

where

$$
\begin{aligned}
&\mathtt{tr}\left(\boldsymbol{P}^\top\boldsymbol{\Sigma}\boldsymbol{P}\,\mathbb{E}\left[\sum_{i=1}^n \omega_i^2(\boldsymbol{x}_i^\top\boldsymbol{\beta}_i+\xi_i)^2\boldsymbol{x}_i\boldsymbol{x}_i^\top\right]\right) \\
&= \mathtt{tr}\left(\boldsymbol{P}^\top\boldsymbol{\Sigma}\boldsymbol{P}\,\mathbb{E}\left[\sum_{i=1}^n \omega_i^2(\boldsymbol{x}_i^\top\boldsymbol{\beta}_i\boldsymbol{\beta}_i^\top\boldsymbol{x}_i+\sigma^2)\boldsymbol{x}_i\boldsymbol{x}_i^\top\right]\right) \\
&= \|\boldsymbol{\omega}\|^2\mathtt{tr}\left(\boldsymbol{P}^\top\boldsymbol{\Sigma}\boldsymbol{P}\left(\mathbb{E}\left[\boldsymbol{x}\boldsymbol{x}^\top\boldsymbol{x}\boldsymbol{x}^\top\right]+\sigma^2\boldsymbol{\Sigma}\right)\right) \\
&= \|\boldsymbol{\omega}\|^2\left(\mathtt{tr}\left(\boldsymbol{\Sigma}\boldsymbol{P}^\top\boldsymbol{\Sigma}\boldsymbol{P}\right)\left(\mathtt{tr}\left(\boldsymbol{\Sigma}\right)+\sigma^2\right)+2\mathtt{tr}\left(\boldsymbol{\Sigma}^2\boldsymbol{P}^\top\boldsymbol{\Sigma}\boldsymbol{P}\right)\right),
\end{aligned}
$$

and

$$
\begin{aligned}
&\mathtt{tr}\left(\boldsymbol{P}^\top\boldsymbol{\Sigma}\boldsymbol{P}\,\mathbb{E}\left[\sum_{i\neq j} \omega_i\omega_j(\boldsymbol{x}_i^\top\boldsymbol{\beta}_i+\xi_i)\boldsymbol{x}_i\boldsymbol{x}_j^\top(\boldsymbol{x}_j^\top\boldsymbol{\beta}_j+\xi_j)\right]\right) \\
&= \mathtt{tr}\left(\boldsymbol{P}^\top\boldsymbol{\Sigma}\boldsymbol{P}\,\mathbb{E}\left[\sum_{i\neq j} \omega_i\omega_j\boldsymbol{x}_i\boldsymbol{x}_i^\top\boldsymbol{\beta}_i\boldsymbol{\beta}_j^\top\boldsymbol{x}_j\boldsymbol{x}_j^\top\right]\right) \\
&= \mathtt{tr}\left(\boldsymbol{P}^\top\boldsymbol{\Sigma}\boldsymbol{P}\right)\sum_{i\neq j}\omega_i\omega_j\,\mathbb{E}[\boldsymbol{\beta}_i^\top\boldsymbol{\beta}_j] \\
&= \mathtt{tr}\left(\boldsymbol{\Sigma}^2\boldsymbol{P}^\top\boldsymbol{\Sigma}\boldsymbol{P}\right)\boldsymbol{\omega}^\top(\boldsymbol{R}-\boldsymbol{I})\boldsymbol{\omega}.
\end{aligned}
$$

Combining all together and letting $M := \mathtt{tr}\left(\boldsymbol{\Sigma}\right) + \sigma^2$, we obtain

$$\begin{aligned}\mathcal{L}(\boldsymbol{P}, \boldsymbol{\omega}) = M - 2\mathtt{tr}\left(\boldsymbol{\Sigma}^2\boldsymbol{P}\right)\boldsymbol{\omega}^\top\boldsymbol{r} \\ + M\|\boldsymbol{\omega}\|^2\mathtt{tr}\left(\boldsymbol{\Sigma}\boldsymbol{P}^\top\boldsymbol{\Sigma}\boldsymbol{P}\right) + (\|\boldsymbol{\omega}\|^2 + \boldsymbol{\omega}^\top\boldsymbol{R}\boldsymbol{\omega})\mathtt{tr}\left(\boldsymbol{\Sigma}^2\boldsymbol{P}^\top\boldsymbol{\Sigma}\boldsymbol{P}\right).\end{aligned} \tag{E.12}$$

For simplicity, and without loss of generality, let

$$\tilde{\boldsymbol{P}} = \sqrt{\boldsymbol{\Sigma}}\boldsymbol{P}\sqrt{\boldsymbol{\Sigma}}. \tag{E.13}$$

Then, we obtain

$$\begin{aligned}\mathcal{L}(\tilde{\boldsymbol{P}}, \boldsymbol{\omega}) = M - 2\mathtt{tr}\left(\boldsymbol{\Sigma}\tilde{\boldsymbol{P}}\right)\boldsymbol{\omega}^\top\boldsymbol{r} \\ + M\|\boldsymbol{\omega}\|^2\mathtt{tr}\left(\tilde{\boldsymbol{P}}^\top\tilde{\boldsymbol{P}}\right) + (\|\boldsymbol{\omega}\|^2 + \boldsymbol{\omega}^\top\boldsymbol{R}\boldsymbol{\omega})\mathtt{tr}\left(\boldsymbol{\Sigma}\tilde{\boldsymbol{P}}^\top\tilde{\boldsymbol{P}}\right).\end{aligned} \tag{E.14}$$

Further, the gradients can be written as

$$\nabla_{\tilde{\boldsymbol{P}}}\mathcal{L}(\tilde{\boldsymbol{P}}, \boldsymbol{\omega}) = -2\boldsymbol{\omega}^\top\boldsymbol{r}\boldsymbol{\Sigma} + 2M\|\boldsymbol{\omega}\|^2\tilde{\boldsymbol{P}} + 2(\|\boldsymbol{\omega}\|^2 + \boldsymbol{\omega}^\top\boldsymbol{R}\boldsymbol{\omega})\boldsymbol{\Sigma}\tilde{\boldsymbol{P}}, \tag{E.15}$$

$$\nabla_{\boldsymbol{\omega}}\mathcal{L}(\tilde{\boldsymbol{P}}, \boldsymbol{\omega}) = -2\mathtt{tr}\left(\boldsymbol{\Sigma}\tilde{\boldsymbol{P}}\right)\boldsymbol{r} + 2M\mathtt{tr}\left(\tilde{\boldsymbol{P}}^\top\tilde{\boldsymbol{P}}\right)\boldsymbol{\omega} + 2\mathtt{tr}\left(\boldsymbol{\Sigma}\tilde{\boldsymbol{P}}^\top\tilde{\boldsymbol{P}}\right)(\boldsymbol{I}_n + \boldsymbol{R})\boldsymbol{\omega}. \tag{E.16}$$

Using the first-order optimality condition, and setting $\nabla_{\tilde{\boldsymbol{P}}}\mathcal{L}(\tilde{\boldsymbol{P}}, \boldsymbol{\omega}) = 0$ and $\nabla_{\boldsymbol{\omega}}\mathcal{L}(\tilde{\boldsymbol{P}}, \boldsymbol{\omega}) = 0$, we obtain

$$\begin{aligned}\tilde{\boldsymbol{P}} &= \left(M\|\boldsymbol{\omega}\|^2\boldsymbol{I} + (\|\boldsymbol{\omega}\|^2 + \boldsymbol{\omega}^\top\boldsymbol{R}\boldsymbol{\omega})\boldsymbol{\Sigma}\right)^{-1}\boldsymbol{\Sigma}\boldsymbol{\omega}^\top\boldsymbol{r} \\ &= \frac{\boldsymbol{\omega}^\top\boldsymbol{r}}{M\|\boldsymbol{\omega}\|^2}\left(\frac{\|\boldsymbol{\omega}\|^2 + \boldsymbol{\omega}^\top\boldsymbol{R}\boldsymbol{\omega}}{M\|\boldsymbol{\omega}\|^2}\boldsymbol{I} + \boldsymbol{\Sigma}^{-1}\right)^{-1} \\ &= \frac{\boldsymbol{\omega}^\top\boldsymbol{r}}{M\|\boldsymbol{\omega}\|^2}\left(\frac{\gamma}{M}\cdot\boldsymbol{I} + \boldsymbol{\Sigma}^{-1}\right)^{-1},\end{aligned} \tag{E.17a}$$

where

$$\gamma := \frac{\boldsymbol{\omega}^\top\boldsymbol{R}\boldsymbol{\omega}}{\|\boldsymbol{\omega}\|^2} + 1.$$

Further,

$$\begin{aligned}\boldsymbol{\omega} &= \left(\left(M\mathtt{tr}\left(\tilde{\boldsymbol{P}}^\top\tilde{\boldsymbol{P}}\right) + \mathtt{tr}\left(\boldsymbol{\Sigma}\tilde{\boldsymbol{P}}^\top\tilde{\boldsymbol{P}}\right)\right)\boldsymbol{I} + \mathtt{tr}\left(\boldsymbol{\Sigma}\tilde{\boldsymbol{P}}^\top\tilde{\boldsymbol{P}}\right)\boldsymbol{R}\right)^{-1}\mathtt{tr}\left(\boldsymbol{\Sigma}\tilde{\boldsymbol{P}}\right)\boldsymbol{r} \\ &= \frac{\mathtt{tr}\left(\boldsymbol{\Sigma}\tilde{\boldsymbol{P}}\right)}{M\mathtt{tr}\left(\tilde{\boldsymbol{P}}^\top\tilde{\boldsymbol{P}}\right) + \mathtt{tr}\left(\boldsymbol{\Sigma}\tilde{\boldsymbol{P}}^\top\tilde{\boldsymbol{P}}\right)}\left(\boldsymbol{I} + \frac{\mathtt{tr}\left(\boldsymbol{\Sigma}\tilde{\boldsymbol{P}}^\top\tilde{\boldsymbol{P}}\right)}{M\mathtt{tr}\left(\tilde{\boldsymbol{P}}^\top\tilde{\boldsymbol{P}}\right) + \mathtt{tr}\left(\boldsymbol{\Sigma}\tilde{\boldsymbol{P}}^\top\tilde{\boldsymbol{P}}\right)}\boldsymbol{R}\right)^{-1}\boldsymbol{r}.\end{aligned} \tag{E.17b}$$

Let

$$\boldsymbol{\Sigma}_\gamma := \frac{\gamma}{M} \cdot \boldsymbol{I} + \boldsymbol{\Sigma}^{-1}.$$

Then, we get

$$\begin{aligned}
\frac{\mathtt{tr}\left(\boldsymbol{\Sigma}\tilde{\boldsymbol{P}}^\top\tilde{\boldsymbol{P}}\right)}{M\mathtt{tr}\left(\tilde{\boldsymbol{P}}^\top\tilde{\boldsymbol{P}}\right)+\mathtt{tr}\left(\boldsymbol{\Sigma}\tilde{\boldsymbol{P}}^\top\tilde{\boldsymbol{P}}\right)} &= \left(1+M\frac{\mathtt{tr}\left(\tilde{\boldsymbol{P}}^\top\tilde{\boldsymbol{P}}\right)}{\mathtt{tr}\left(\boldsymbol{\Sigma}\tilde{\boldsymbol{P}}^\top\tilde{\boldsymbol{P}}\right)}\right)^{-1} \\
&= \left(1+M\frac{\mathtt{tr}\left(\boldsymbol{\Sigma}_\gamma^{-2}\right)}{\mathtt{tr}\left(\boldsymbol{\Sigma}\boldsymbol{\Sigma}_\gamma^{-2}\right)}\right)^{-1} \\
&= \left(1+M\sum_{i=1}^d \frac{s_i^2}{(M+\gamma s_i)^2}\left(\sum_{i=1}^d \frac{s_i^3}{(M+\gamma s_i)^2}\right)^{-1}\right)^{-1} \\
&=: h_2(\gamma).
\end{aligned}$$

Here, the last equality follows from eigen decomposition $\boldsymbol{\Sigma} = \boldsymbol{U}\mathrm{diag}(\mathbf{s})\boldsymbol{U}^\top$ with $\boldsymbol{s} = [s_1, \ldots, s_d]^\top \in \mathbb{R}^d_{++}$.

Now, plugging $\tilde{\boldsymbol{P}}$ defined in (E.17a) within $\boldsymbol{\omega}$ given in (E.17b), we obtain

$$\boldsymbol{\omega} = \frac{\mathtt{tr}\left(\boldsymbol{\Sigma}\tilde{\boldsymbol{P}}\right)}{M\mathtt{tr}\left(\tilde{\boldsymbol{P}}^\top\tilde{\boldsymbol{P}}\right)+\mathtt{tr}\left(\boldsymbol{\Sigma}\tilde{\boldsymbol{P}}^\top\tilde{\boldsymbol{P}}\right)} \cdot \left(h_2(\gamma)\cdot\boldsymbol{R}+\boldsymbol{I}\right)^{-1}\boldsymbol{r}. \tag{E.18}$$

Using the above formulae for $\boldsymbol{\omega}$, we rewrite $\gamma = \boldsymbol{\omega}^\top\boldsymbol{R}\boldsymbol{\omega}/\|\boldsymbol{\omega}\|^2 + 1$ as

$$\begin{aligned}
\gamma - 1 &= \frac{\boldsymbol{r}^\top(h_2(\gamma)\boldsymbol{R}+\boldsymbol{I})^{-1}\boldsymbol{R}(h_2(\gamma)\boldsymbol{R}+\boldsymbol{I})^{-1}\boldsymbol{r}}{\boldsymbol{r}^\top(h_2(\gamma)\boldsymbol{R}+\boldsymbol{I})^{-2}\boldsymbol{r}} \\
&= \sum_{i=1}^n \frac{\lambda_i a_i^2}{(1+h_2(\gamma)\lambda_i)^2}\left(\sum_{i=1}^n \frac{a_i^2}{(1+h_2(\gamma)\lambda_i)^2}\right)^{-1} \\
&=: h_1(h_2(\gamma)),
\end{aligned} \tag{E.19}$$

where the second equality follows from Assumption 6.1 where $\boldsymbol{r} = \boldsymbol{E}\boldsymbol{a}$ with $\boldsymbol{a} = [a_1, \cdots, a_n]^\top \in \mathbb{R}^n$, and the fact that $\boldsymbol{R} = \boldsymbol{E}\mathrm{diag}(\boldsymbol{\lambda})\boldsymbol{E}^\top$ denotes the eigen decomposition of $\boldsymbol{R}$, with $\boldsymbol{\lambda} = [\lambda_1, \ldots, \lambda_n]^\top \in \mathbb{R}^n_+$.

It follows from Lemma E.1 that there exists $\gamma^\star \geq 1$ such that $h_1(h_2(\gamma^\star)) + 1 = \gamma^\star$. From

(E.17a) and (E.18), we obtain

$$\begin{aligned}\tilde{\boldsymbol{P}} &= C(\boldsymbol{r},\boldsymbol{\omega},\boldsymbol{\Sigma})\cdot\left(\frac{\gamma^\star}{M}\cdot\boldsymbol{I}+\boldsymbol{\Sigma}^{-1}\right)^{-1}, \quad \text{and}\\ \boldsymbol{\omega} &= c(\boldsymbol{r},\boldsymbol{\omega},\boldsymbol{\Sigma})\cdot\left(h_2(\gamma^\star)\cdot\boldsymbol{R}+\boldsymbol{I}\right)^{-1}\boldsymbol{r}.\end{aligned} \tag{E.20}$$

for some $C(\boldsymbol{r},\boldsymbol{\omega},\boldsymbol{\Sigma}) = \frac{\boldsymbol{\omega}^\top\boldsymbol{r}}{M\|\boldsymbol{\omega}\|^2}$ and $c(\boldsymbol{r},\boldsymbol{\omega},\boldsymbol{\Sigma}) = \frac{\mathtt{tr}\left(\boldsymbol{\Sigma}\tilde{\boldsymbol{P}}\right)}{M\mathtt{tr}\left(\tilde{\boldsymbol{P}}^\top\tilde{\boldsymbol{P}}\right)+\mathtt{tr}\left(\boldsymbol{\Sigma}\tilde{\boldsymbol{P}}^\top\tilde{\boldsymbol{P}}\right)}$.

Now, using the our definition $\tilde{\boldsymbol{P}} = \sqrt{\boldsymbol{\Sigma}}\boldsymbol{P}\sqrt{\boldsymbol{\Sigma}}$, we obtain

$$\begin{aligned}\boldsymbol{P}(\gamma^\star) &= C(\boldsymbol{r},\boldsymbol{\omega},\boldsymbol{\Sigma})\cdot\boldsymbol{\Sigma}^{-\frac{1}{2}}\left(\frac{\gamma^\star}{M}\cdot\boldsymbol{\Sigma}+\boldsymbol{I}\right)^{-1}\boldsymbol{\Sigma}^{-\frac{1}{2}}, \quad \text{and}\\ \boldsymbol{\omega}(\gamma^\star) &= c(\boldsymbol{r},\boldsymbol{\omega},\boldsymbol{\Sigma})\cdot\Big(h_2(\gamma^\star)\cdot\boldsymbol{R}+\boldsymbol{I}\Big)^{-1}\boldsymbol{r}.\end{aligned}$$

This completes the proof. ■

E.3.2 Proof of Theorem 6.4

We first provide the following Lemma.

Lemma E.2 *Consider the functions $h_1(\bar{\gamma})$ and $h_2(\gamma)$ defined in* (6.15a) *and* (6.15b)*, respectively, where $\lambda_i \geq 0$, $a_i \neq 0$ for at least one i, $s_i > 0$, and $M = \sigma^2 + \sum_{i=1}^d s_i > 0$. Suppose $\Delta_{\boldsymbol{\Sigma}}\cdot\Delta_{\boldsymbol{R}} < M + s_{\min}$, where $\Delta_{\boldsymbol{\Sigma}}$ and $\Delta_{\boldsymbol{R}}$ denote the effective spectral gaps of $\boldsymbol{\Sigma}$ and $\boldsymbol{R}$, respectively, as given in* (6.14)*; and $s_{\min}$ is the smallest eigenvalue of $\boldsymbol{\Sigma}$. We have that*

$$\left|\frac{\partial h_2}{\partial\gamma}\cdot\frac{\partial h_1}{\partial h_2}\right| \leq \frac{\Delta_{\boldsymbol{\Sigma}}^2\cdot\Delta_{\boldsymbol{R}}^2}{(M+s_{\min})^2} < 1.$$

Proof. Let

$$B(\gamma) = \sum_{i=1}^d \frac{s_i^3}{(M+\gamma s_i)^2}, \quad C(\gamma) = \sum_{i=1}^d \frac{s_i^2}{(M+\gamma s_i)^2}, \quad A(\gamma) = 1 + M\frac{C(\gamma)}{B(\gamma)}.$$

The derivatives of $B(\gamma)$ and $C(\gamma)$ are

$$B'(\gamma) = -2\sum_{i=1}^d \frac{s_i^4}{(M+\gamma s_i)^3}, \quad C'(\gamma) = -2\sum_{i=1}^d \frac{s_i^3}{(M+\gamma s_i)^3}.$$

The gradient of $h_2(\gamma) = A(\gamma)^{-1}$ is

$$\frac{\partial h_2}{\partial \gamma} = -M \left(\frac{1}{A(\gamma)B(\gamma)} \right)^2 \left(C'(\gamma)B(\gamma) - C(\gamma)B'(\gamma) \right). \tag{E.21}$$

It can be seen that

$$\left(\frac{1}{A(\gamma)} \right)^2 \leq M^{-2} \left(\sum_{i=1}^{d} \frac{s_i^3}{(M+\gamma s_i)^2} \right)^2 \left(\sum_{i=1}^{d} \frac{s_i^2}{(M+\gamma s_i)^2} \right)^{-2},$$

which implies that

$$\left(\frac{1}{A(\gamma)B(\gamma)} \right)^2 \leq M^{-2} \left(\sum_{i=1}^{d} \frac{s_i^2}{(M+\gamma s_i)^2} \right)^{-2}. \tag{E.22a}$$

Further, we have

$$\begin{aligned} C'(\gamma)B(\gamma) - C(\gamma)B'(\gamma) &= -2 \sum_{i=1}^{d} \frac{s_i^3}{(M+\gamma s_i)^3} \sum_{i=1}^{d} \frac{s_i^3}{(M+\gamma s_i)^2} \\ &\quad + 2 \sum_{i=1}^{d} \frac{s_i^2}{(M+\gamma s_i)^2} \sum_{i=1}^{d} \frac{s_i^4}{(M+\gamma s_i)^3} \\ &= \sum_{i=1}^{d} \sum_{j=1}^{d} \frac{T_{ij}}{(M+\gamma s_i)^3 (M+\gamma s_j)^3} \\ &= M \cdot \sum_{i=1}^{d} \sum_{j=1}^{d} \frac{s_i^2 s_j^2 (s_i - s_i)^2}{(M+\gamma s_i)^3 (M+\gamma s_j)^3}, \end{aligned} \tag{E.22b}$$

where

$$\begin{aligned} T_{ij} &= s_i^2 (M+\gamma s_i) s_j^4 + s_i^4 s_j^2 (M+\gamma s_j) \\ &\quad - s_i^3 s_j^3 (M+\gamma s_j) - s_i^3 (M+\gamma s_i) s_j^3 \\ &= s_i^2 s_j^2 \left(M \cdot (s_j^2 + s_i^2 - 2 s_i s_j) + \gamma (s_i s_j^2 + s_i^2 s_j - s_i s_j^2 - s_i^2 s_j) \right). \end{aligned} \tag{E.22c}$$

Thus, substituting (E.22a) and (E.22b) into (E.21), we obtain

$$\left| \frac{\partial h_2}{\partial \gamma} \right| \leq M \cdot M^{-1} \left(\sum_{i=1}^{d} \frac{s_i^2}{(M+\gamma s_i)^2} \right)^{-2} \sum_{i,j=1}^{d} \frac{s_i^2 s_j^2 (s_i - s_j)^2}{(M+\gamma s_i)^3 (M+\gamma s_j)^3}. \tag{E.23}$$

Next, we derive $\frac{\partial h_1}{\partial \bar{\gamma}}$. Let

$$D(\bar{\gamma}) = \sum_{i=1}^{n} \frac{\lambda_i a_i^2}{(1+\bar{\gamma}\lambda_i)^2}, \quad E(\bar{\gamma}) = \sum_{i=1}^{n} \frac{a_i^2}{(1+\bar{\gamma}\lambda_i)^2}.$$

We have

$$D'(\bar{\gamma}) = -2\sum_{i=1}^{n} \frac{\lambda_i^2 a_i^2}{(1+\bar{\gamma}\lambda_i)^3}, \quad E'(\bar{\gamma}) = -2\sum_{i=1}^{n} \frac{\lambda_i a_i^2}{(1+\bar{\gamma}\lambda_i)^3}.$$

The derivative of h_1 with respect to $\bar{\gamma}$ is given by

$$\frac{\partial h_1}{\partial \bar{\gamma}} = -\left(\frac{1}{E(\bar{\gamma})}\right)^2 \left(E(\bar{\gamma})D'(\bar{\gamma}) - D(\bar{\gamma})E'(\bar{\gamma})\right). \tag{E.24}$$

Substituting into (E.24), we get

$$\left(\frac{1}{E(\bar{\gamma})}\right)^2 = \left(\sum_{i=1}^{n} \frac{a_i^2}{(1+\bar{\gamma}\lambda_i)^2}\right)^{-2}, \tag{E.25a}$$

and

$$\begin{aligned} E(\bar{\gamma})D'(\bar{\gamma}) - D(\bar{\gamma})E'(\bar{\gamma}) &= 2\sum_{i=1}^{n} \frac{\lambda_i^2 a_i^2}{(1+\bar{\gamma}\lambda_i)^3} \sum_{i=1}^{n} \frac{a_i^2}{(1+\bar{\gamma}\lambda_i)^2} \\ &\quad - 2\sum_{i=1}^{n} \frac{\lambda_i a_i^2}{(1+\bar{\gamma}\lambda_i)^2} \sum_{i=1}^{n} \frac{a_i^2 \lambda_i}{(1+\bar{\gamma}\lambda_i)^3} \\ &= \sum_{i=1}^{n}\sum_{j=1}^{n} \frac{\bar{T}_{ij}}{(1+\bar{\gamma}\lambda_i)^3(1+\bar{\gamma}\lambda_j)^3} \\ &= \sum_{i=1}^{n}\sum_{j=1}^{n} \frac{a_i^2 a_j^2 \left(\lambda_i^2 + \lambda_j^2 - 2\lambda_i\lambda_j\right)}{(1+\bar{\gamma}\lambda_i)^3(1+\bar{\gamma}\lambda_j)^3}. \end{aligned} \tag{E.25b}$$

Here,

$$\begin{aligned} \bar{T}_{ij} &= \lambda_i^2 a_i^2 a_j^2 (1+\bar{\gamma}\lambda_j) + a_i^2(1+\bar{\gamma}\lambda_i)\lambda_j^2 a_j^2 \\ &\quad - \lambda_i a_i^2 (1+\bar{\gamma}\lambda_i) a_j^2 \lambda_j - a_i^2 \lambda_i \lambda_j a_j^2 (1+\bar{\gamma}\lambda_j) \\ &= a_i^2 a_j^2 \left(\lambda_i^2(1+\bar{\gamma}\lambda_j) + (1+\bar{\gamma}\lambda_i)\lambda_j^2 - \lambda_i(1+\bar{\gamma}\lambda_i)\lambda_j - \lambda_i\lambda_j(1+\bar{\gamma}\lambda_j)\right). \end{aligned} \tag{E.25c}$$

Hence, substituting (E.25a) and (E.25b) into (E.24) gives

$$\frac{\partial h_1}{\partial \bar{\gamma}} = -\left(\sum_{i=1}^{n} \frac{a_i^2}{(1+\bar{\gamma}\lambda_i)^2}\right)^{-2} \sum_{i,j=1}^{n} \frac{a_i^2 a_j^2 (\lambda_i - \lambda_j)^2}{(1+\bar{\gamma}\lambda_i)^3 (1+\bar{\gamma}\lambda_j)^3}. \tag{E.26}$$

Now, for the combined derivative, we have

$$\begin{aligned}\left|\frac{\partial h_2}{\partial \gamma} \cdot \frac{\partial h_1}{\partial \bar{\gamma}}\right| \leq &\left(\sum_{i=1}^{d} \frac{s_i^2}{(M+\gamma s_i)^2}\right)^{-2} \sum_{i,j=1}^{d} \frac{s_i^2 s_j^2 (s_i - s_j)^2}{(M+\gamma s_i)^3 (M+\gamma s_j)^3} \\ &\cdot \left(\sum_{i=1}^{n} \frac{a_i^2}{(1+\bar{\gamma}\lambda_i)^2}\right)^{-2} \sum_{i,j=1}^{n} \frac{a_i^2 a_j^2 (\lambda_i - \lambda_j)^2}{(1+\bar{\gamma}\lambda_i)^3 (1+\bar{\gamma}\lambda_j)^3}.\end{aligned}$$

Note that $M + \gamma s_i$ and $1 + \bar{\gamma}\lambda_j$ are nonnegative for all i, j. Hence,

$$\begin{aligned}\left|\frac{\partial h_2}{\partial \gamma} \cdot \frac{\partial h_1}{\partial \bar{\gamma}}\right| \leq &\left(\sum_{i=1}^{d} \frac{s_i^2 (M+\gamma s_i)}{(M+\gamma s_i)^3}\right)^{-2} \\ &\cdot \sum_{i,j=1}^{d} \frac{s_i^2 s_j^2 \cdot \Delta_1 \cdot (M+\gamma s_j)(M+\gamma s_i)}{(M+\gamma s_i)^3 (M+\gamma s_j)^3} \\ &\cdot \left(\sum_{i=1}^{n} \frac{a_i^2 (1+\bar{\gamma}\lambda_i)}{(1+\bar{\gamma}\lambda_i)^3}\right)^{-2} \\ &\cdot \sum_{i,j=1}^{d} \frac{a_i^2 a_j^2 \cdot \Delta_2 \cdot (1+\bar{\gamma}\lambda_i)(1+\bar{\gamma}\lambda_j)}{(1+\bar{\gamma}\lambda_i)^3 (1+\bar{\gamma}\lambda_j)^3},\end{aligned}$$

where

$$\Delta_1 := \max_{i,j \in \mathcal{S}} \frac{(s_i - s_j)^2}{(M+\gamma s_j)(M+\gamma s_i)}, \quad \Delta_2 := \max_{i,j \in \mathcal{S}} \frac{(\lambda_i - \lambda_j)^2}{(1+\bar{\gamma}\lambda_i)(1+\bar{\gamma}\lambda_j)}. \tag{E.27}$$

Here, $\mathcal{S} = \{i \in [n] \mid \lambda_i \neq 0\} \subset [n]$.

Finally, setting $\bar{\gamma} = h_2(\gamma)$, since $\gamma \geq 1$ and by our assumption $\Delta_{\boldsymbol{\Sigma}} \cdot \Delta_{\boldsymbol{R}} < M + s_{\min}$, we obtain

$$|h_1'(h_2(\gamma)) \cdot h_2'(\gamma)| = \left|\frac{\partial h_2}{\partial \gamma} \cdot \frac{\partial h_1}{\partial h_2}\right| \leq \Delta_1 \cdot \Delta_2 \leq \frac{\Delta_{\boldsymbol{\Sigma}}^2 \cdot \Delta_{\boldsymbol{R}}^2}{(M+s_{\min})^2} < 1.$$

where $\Delta_{\boldsymbol{\Sigma}}$ and $\Delta_{\boldsymbol{R}}$ are the spectral gaps of $\boldsymbol{\Sigma}$ and $\boldsymbol{R}$; and $s_{\min}$ is the smallest eigenvalue of $\boldsymbol{\Sigma}$; and $M = \sigma^2 + \sum_{i=1}^{d} s_i$. ∎

Proof. [Proof of Theorem 6.4] It follows from Lemma E.1 that there exists $\gamma^\star \geq 1$ such that $h_1(h_2(\gamma^\star)) + 1 = \gamma^\star$. Further, Lemma E.2 establishes that the mapping $h_1(h_2(\gamma)) + 1$

is a contraction mapping, since it satisfies $|\partial h_1(h_2(\gamma))/\partial\gamma| < 1$. Therefore, by the Banach fixed-point theorem, there exists a *unique* fixed-point solution, denoted by $\gamma^\star$, satisfying $\gamma^\star = h_1(h_2(\gamma^\star)) + 1$. This completes the proof of statement **T1.**.

Next, we establish **T2.**. First, from Theorem 6.3, the stationary point $(\boldsymbol{P}^\star, \boldsymbol{\omega}^\star)$, up to rescaling, is defined as

$$\boldsymbol{P}^\star = \boldsymbol{\Sigma}^{-\frac{1}{2}} \left(\frac{\gamma^\star}{M} \cdot \boldsymbol{\Sigma} + \boldsymbol{I} \right)^{-1} \boldsymbol{\Sigma}^{-\frac{1}{2}} \quad \text{and} \quad \boldsymbol{\omega}^\star = (h_2(\gamma^\star) \cdot \boldsymbol{R} + \boldsymbol{I})^{-1} \boldsymbol{r},$$

where $\gamma^\star$ is a fixed point of composite function $h_1(h_2(\gamma)) + 1$.

Given the coercive nature of the loss function $\mathcal{L}(\boldsymbol{P}, \boldsymbol{\omega})$, a global minimum exists. Since the global minimum must satisfy first-order optimality conditions, it must lie within the set of stationary points characterized by Theorem 6.3.

Now, consider arbitrary global minimizers $(\hat{\boldsymbol{P}}, \hat{\boldsymbol{\omega}})$. Define scaling factors $\alpha, \beta > 0$ and write $(\hat{\boldsymbol{P}}, \hat{\boldsymbol{\omega}}) = (\alpha \boldsymbol{P}^\star, \beta \boldsymbol{\omega}^\star)$. Substituting this scaling into the loss function gives:

$$\begin{aligned} \mathcal{L}(\alpha \boldsymbol{P}^\star, \beta \boldsymbol{\omega}^\star) = M - 2\alpha\beta \mathtt{tr}\left(\boldsymbol{\Sigma}^2 \boldsymbol{P}^\star\right) \boldsymbol{\omega}^{\star\top} \boldsymbol{r} + M\alpha^2\beta^2 \|\boldsymbol{\omega}^\star\|^2 \mathtt{tr}\left(\boldsymbol{\Sigma} \boldsymbol{P}^{\star\top} \boldsymbol{\Sigma} \boldsymbol{P}^\star\right) \\ + \alpha^2\beta^2 (\|\boldsymbol{\omega}^\star\|^2 + \boldsymbol{\omega}^{\star\top} \boldsymbol{R} \boldsymbol{\omega}^\star) \mathtt{tr}\left(\boldsymbol{\Sigma}^2 \boldsymbol{P}^{\star\top} \boldsymbol{\Sigma} \boldsymbol{P}^\star\right). \end{aligned} \tag{E.28}$$

Differentiating this expression with respect to α and β and setting the derivatives equal to zero, we obtain the equation

$$-2A + 2\alpha\beta(MCB + (C + D)E) = 0, \tag{E.29a}$$

where we define

$$\begin{aligned} &A := \mathtt{tr}\left(\boldsymbol{\Sigma}^2 \boldsymbol{P}^\star\right) \boldsymbol{\omega}^{\star\top} \boldsymbol{r}, \quad B := \mathtt{tr}\left(\boldsymbol{\Sigma} \boldsymbol{P}^{\star\top} \boldsymbol{\Sigma} \boldsymbol{P}^\star\right), \\ &C := \|\boldsymbol{\omega}^\star\|^2, \quad D := \boldsymbol{\omega}^{\star\top} \boldsymbol{R} \boldsymbol{\omega}^\star, \quad E := \mathtt{tr}\left(\boldsymbol{\Sigma}^2 \boldsymbol{P}^{\star\top} \boldsymbol{\Sigma} \boldsymbol{P}^\star\right). \end{aligned} \tag{E.29b}$$

Equation (E.29) implies a unique constraint on the product $\alpha\beta$, ensuring no independent rescaling beyond a fixed ratio can further minimize the loss. Hence, no distinct minimizers exist other than scaling the original $(\boldsymbol{P}^\star, \boldsymbol{\omega}^\star)$ pair. Thus, the stationary point $(\boldsymbol{P}^\star, \boldsymbol{\omega}^\star)$ from Theorem 6.3 is indeed a unique minimizer up to rescaling.

■

E.3.3 Proof of Corollary 6.2

Proof. Since by assumption $\boldsymbol{\Sigma} = \boldsymbol{I}$, it follows from (6.15b) that

$$\begin{aligned} h_2(\gamma^\star) &= \left(1 + (d+\sigma^2)\sum_{i=1}^{d} \frac{1}{(d+\sigma^2+\gamma^\star+1)^2} \left(\sum_{i=1}^{d} \frac{1}{(d+\sigma^2+\gamma^\star+1)^2}\right)^{-1}\right)^{-1} \\ &= \frac{1}{d+\sigma^2+1}. \end{aligned}$$

Substituting this into (6.16) gives

$$\boldsymbol{P}^\star = \boldsymbol{I}, \quad \text{and} \quad \boldsymbol{\omega}^\star = \left(\boldsymbol{R} + (d+\sigma^2+1)\boldsymbol{I}\right)^{-1}\boldsymbol{r}.$$

Now, recall that the objective function is given by

$$\mathcal{L}(\boldsymbol{\omega}) = M - 2\mathtt{tr}\left(\boldsymbol{\Sigma}^2\boldsymbol{P}\right)\boldsymbol{\omega}^\top\boldsymbol{r} + M\|\boldsymbol{\omega}\|^2\mathtt{tr}\left(\boldsymbol{\Sigma}\boldsymbol{P}^\top\boldsymbol{\Sigma}\boldsymbol{P}\right) + (\|\boldsymbol{\omega}\|^2 + \boldsymbol{\omega}^\top\boldsymbol{R}\boldsymbol{\omega})\mathtt{tr}\left(\boldsymbol{\Sigma}^2\boldsymbol{P}^\top\boldsymbol{\Sigma}\boldsymbol{P}\right),$$

and, by assumption, $M = \sigma^2 + d$.

Substituting $\boldsymbol{P}^\star = \boldsymbol{I}$ and $\boldsymbol{\omega}^\star = \left(\boldsymbol{R} + (d+\sigma^2+1)\boldsymbol{I}\right)^{-1}\boldsymbol{r}$ into the objective (E.12), and using $\boldsymbol{\Sigma} = \boldsymbol{I}$, we get:

$$\mathcal{L}(\boldsymbol{\omega}^\star) = (\sigma^2+d) - 2\cdot d\cdot\boldsymbol{r}^\top\boldsymbol{\omega}^\star + (\sigma^2+d)\cdot\|\boldsymbol{\omega}^\star\|^2 d + d\left(\|\boldsymbol{\omega}^\star\|^2 + \boldsymbol{\omega}^{\star\top}\boldsymbol{R}\boldsymbol{\omega}^\star\right).$$

The expression simplifies as

$$\mathcal{L}(\boldsymbol{\omega}^\star) = (\sigma^2+d) - 2d\cdot\boldsymbol{r}^\top\left(\boldsymbol{R} + (d+\sigma^2+1)\boldsymbol{I}\right)^{-1}\boldsymbol{r} + (\sigma^2+d)d\|\boldsymbol{\omega}^\star\|^2 + d\left(\|\boldsymbol{\omega}^\star\|^2 + \boldsymbol{\omega}^{\star\top}\boldsymbol{R}\boldsymbol{\omega}^\star\right).$$

Next, we compute $\|\boldsymbol{\omega}^\star\|^2$ and $\boldsymbol{\omega}^{\star\top}\boldsymbol{R}\boldsymbol{\omega}^\star$. By definition, we have

$$\|\boldsymbol{\omega}^\star\|^2 = \boldsymbol{r}^\top\left(\boldsymbol{R} + (d+\sigma^2+1)\boldsymbol{I}\right)^{-2}\boldsymbol{r},$$

and

$$\boldsymbol{\omega}^{\star\top}\boldsymbol{R}\boldsymbol{\omega}^\star = \boldsymbol{r}^\top\left(\boldsymbol{R} + (d+\sigma^2+1)\boldsymbol{I}\right)^{-1}\boldsymbol{R}\left(\boldsymbol{R} + (d+\sigma^2+1)\boldsymbol{I}\right)^{-1}\boldsymbol{r}.$$

Thus,

$$\begin{aligned} &(d+\sigma^2+1)\|\boldsymbol{\omega}^\star\|^2 + \boldsymbol{\omega}^{\star\top}\boldsymbol{R}\boldsymbol{\omega}^\star \\ &= \boldsymbol{r}^\top\left(\boldsymbol{R} + (d+\sigma^2+1)\boldsymbol{I}\right)^{-1}\left((d+\sigma^2+1)\boldsymbol{I} + \boldsymbol{R}\right)\left(\boldsymbol{R} + (d+\sigma^2+1)\boldsymbol{I}\right)^{-1}\boldsymbol{r} \\ &= \boldsymbol{r}^\top\left(\boldsymbol{R} + (d+\sigma^2+1)\boldsymbol{I}\right)^{-1}\boldsymbol{r}. \end{aligned}$$

Substituting this result back into the objective function gives

$$\mathcal{L}(\boldsymbol{\omega}^\star) = (\sigma^2 + d) - d \cdot \boldsymbol{r}^\top \left(\boldsymbol{R} + (d + \sigma^2 + 1)\boldsymbol{I}\right)^{-1} \boldsymbol{r}.$$

■

E.4 Loss landscape of 1-layer GLA

E.4.1 Proof of Theorem 6.5

Proof.

We first prove that under Assumption 6.2, $\mathcal{L}^\star_{\text{gla}} = \min_{\boldsymbol{P}\in\mathbb{R}^{d\times d},\boldsymbol{\omega}\in\mathcal{W}} \mathcal{L}_{\text{wpgd}}(\boldsymbol{P},\boldsymbol{\omega})$ where $\mathcal{W}$ is the search space of weighting vector $\boldsymbol{\omega}\in\mathbb{R}^n$ defined as

$$\mathcal{W} := \left\{ \left[\omega_1 \mathbf{1}_{n_1}^\top \ \cdots \ \omega_K \mathbf{1}_{n_K}^\top\right]^\top \in \mathbb{R}^n \;\middle|\; 0 \le \omega_i \le \omega_j \le 1,\ \forall 1 \le i \le j \le K \right\}.$$

Step 1: Define a set $\bar{\mathcal{W}} := \left\{ \left[\omega_1 \ \cdots \ \omega_n\right]^\top \in \mathbb{R}^n \;\middle|\; 0 \le \omega_i \le \omega_j \le 1,\ \forall 1 \le i \le j \le n \right\}$ and we have $\mathcal{W} \in \bar{\mathcal{W}}$. Given scalar gating $\boldsymbol{G}_i = \begin{bmatrix} * & * \\ g_i \mathbf{1}^\top & * \end{bmatrix}$, following (6.12), the weighting vector returns

$$\boldsymbol{\omega} := \left[g_{1:n+1} \ \cdots \ g_{n:n+1}\right]^\top.$$

Since GLA with scalar gating valued in $[0,1]$ following Assumption 6.2, that is, $g_i \in [0,1]$, we have $g_{i:n+1} \le g_{j:n+1}$ for $1 \le i \le j \le n$. Therefore, any weighting vector $\boldsymbol{\omega}$ implemented by GLA gating should be inside $\bar{\mathcal{W}}$.

Step 2: Next, we will show that

$$\boldsymbol{\omega}^\star \in \mathcal{W} \quad \text{where} \quad \boldsymbol{\omega}^\star = \arg\min_{\boldsymbol{P},\boldsymbol{\omega}\in\bar{\mathcal{W}}} \mathcal{L}_{\text{wpgd}}(\boldsymbol{P},\boldsymbol{\omega}).$$

Define the weighting vector $\boldsymbol{\omega} = [\boldsymbol{\omega}_1^\top \ \cdots \ \boldsymbol{\omega}_K^\top]^\top \in \mathbb{R}^n$ where we have $\boldsymbol{\omega}_k = [\omega_1^{(k)} \ \cdots \ \omega_{n_k}^{(k)}]^\top \in \mathbb{R}^{n_k}$. For any $\boldsymbol{\omega} \notin \mathcal{W}$, there exist (i,j,k) with $i = j-1$ such that $\omega_i^{(k)} < \omega_j^{(k)}$. Given gradient in (E.16), we have that

$$\nabla_{\omega_i^{(k)}}\mathcal{L} - \nabla_{\omega_j^{(k)}}\mathcal{L} = 2\left(M\,\mathtt{tr}\left(\tilde{\boldsymbol{P}}^\top\tilde{\boldsymbol{P}}\right) + \mathtt{tr}\left(\boldsymbol{\Sigma}\tilde{\boldsymbol{P}}^\top\tilde{\boldsymbol{P}}\right)\right)(\omega_i^{(k)} - \omega_j^{(k)}).$$

Here, (E.17a) gives that (optimal) $\tilde{\boldsymbol{P}} \neq 0$ and therefore, we have that $\left(M\mathtt{tr}\left(\tilde{\boldsymbol{P}}^\top\tilde{\boldsymbol{P}}\right)+\mathtt{tr}\left(\boldsymbol{\Sigma}\tilde{\boldsymbol{P}}^\top\tilde{\boldsymbol{P}}\right)\right) > 0$ and $\nabla_{\omega_i^{(k)}}\mathcal{L} < \nabla_{\omega_j^{(k)}}\mathcal{L}$. Therefore either increasing $\omega_i^{(k)}$ (if $\nabla_{\omega_i^{(k)}}\mathcal{L} < 0$) or decreasing $\omega_j^{(k)}$ (if $\nabla_{\omega_j^{(k)}}\mathcal{L} > 0$) will reduce the loss. This results in showing that the optimal weighting vector $\boldsymbol{\omega}^\star$ satisfies $\omega_i^{(k)} = \omega_j^{(k)}$ for any $i, j \in [n_k]$ and $k \in [K]$. Hence, $\boldsymbol{\omega}^\star \in \mathcal{W}$.

Step 3: Finally, we will show that any $\boldsymbol{\omega} \in \mathcal{W}$ can be obtained via the GLA gating. Let $\boldsymbol{\omega} = [\omega_1\mathbf{1}_{n_1}^\top \ \cdots \ \omega_K\mathbf{1}_{n_K}^\top]^\top$ be any vector in $\mathcal{W}$ and assume that $\omega_K = \alpha < 1$ without loss of generality. Then such sample weighting can be achieved via the gating

$$\left[\mathbf{1}_{n_1}^\top \ \ \frac{\omega_1}{\omega_{1:K}} \ \ \cdots \ \ \mathbf{1}_{n_k}^\top \ \ \frac{\omega_k}{\omega_{k:K}} \ \ \cdots \ \ \mathbf{1}_{n_K}^\top \ \ \frac{\omega_K}{\omega_{K:K}}\right]^\top.$$

Let $\omega_k' := \frac{\omega_k}{\omega_{k:K}}$ and let $\boldsymbol{w}_g$ be in the form of

$$\boldsymbol{w}_g = \begin{bmatrix}\mathbf{0}_{d+1}\\ \tilde{\boldsymbol{w}}_g\end{bmatrix} \in \mathbb{R}^{d+p+1}.$$

Then, it remains to show that there exists $\tilde{\boldsymbol{w}}_g$ satisfying

$$\phi(\tilde{\boldsymbol{w}}_g^\top\bar{\boldsymbol{c}}_k) = \begin{cases}1, & k = 0,\\ \omega_k', & k \in [K].\end{cases}$$

The linear independence assumption in Assumption 6.2 implies that the problem is feasible, which completes the proof of (6.25).

Proof of (6.26)**:** Recap the optimal weighting from (6.16) which takes the form of

$$\boldsymbol{\omega}^\star = \left(h_2(\gamma^\star)\cdot\boldsymbol{R}+\boldsymbol{I}\right)^{-1}\boldsymbol{r},$$

where $h_2(\gamma^\star) > 0$. Since Assumption 6.3 holds and $n_1 = n_2 = \cdots = n_K := \bar{n}$, $\boldsymbol{R}$ is block diagonal matrix, where each block is an all-ones matrix. That is

$$\boldsymbol{R} = \begin{bmatrix}\mathbf{1}_{\bar{n}\times\bar{n}} & \mathbf{0} & \cdots & \mathbf{0}\\ \mathbf{0} & \mathbf{1}_{\bar{n}\times\bar{n}} & \cdots & \mathbf{0}\\ \vdots & \vdots & \ddots & \vdots\\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{1}_{\bar{n}\times\bar{n}}\end{bmatrix}.$$

Therefore, we get

$$(h_2(\gamma^\star)\cdot \boldsymbol{R}+\boldsymbol{I})^{-1}=\boldsymbol{I}-\frac{h_2(\gamma^\star)}{h_2(\gamma^\star)\cdot \bar{n}+1}\boldsymbol{R}.$$

Since $\boldsymbol{R}\boldsymbol{r}=\bar{n}\boldsymbol{r}$, then

$$\boldsymbol{\omega}^\star=\frac{1}{h_2(\gamma^\star)\cdot \bar{n}+1}\boldsymbol{r}.$$

Therefore, the optimal weighting (up to a scalar) is inside the set $\mathcal{W}$. Combining it with (6.25) completes the proof.

■

E.4.2 Proof of Theorem 6.6

Proof. Following the similar proof of Theorem 6.5, and letting $\tilde{\mathcal{W}} := \left\{\left[\omega_1\mathbf{1}_{n_1}^\top \ \cdots \ \omega_K\mathbf{1}_{n_K}^\top\right]^\top \in \mathbb{R}^n\right\}$, we obtain

$$\min_{\boldsymbol{P}\in\mathbb{R}^{d\times d},\boldsymbol{\omega}\in\tilde{\mathcal{W}}}\mathcal{L}_{\text{wpgd}}(\boldsymbol{P},\boldsymbol{\omega})=\min_{\boldsymbol{P}\in\mathbb{R}^{d\times d},\boldsymbol{\omega}\in\mathbb{R}^n}\mathcal{L}_{\text{wpgd}}(\boldsymbol{P},\boldsymbol{\omega}).$$

Therefore, it remains to show that any $\boldsymbol{\omega}\in\tilde{\mathcal{W}}$ can be implemented via some gating function. Let $\boldsymbol{\omega}=[\omega_1\mathbf{1}_{n_1}^\top \ \cdots \ \omega_K\mathbf{1}_{n_K}^\top]^\top$ be arbitrary weighting in $\tilde{\mathcal{W}}$. Theorem 6.5 has shown that if $\omega_1\le\omega_2\le\cdots\le\omega_K$, GLA with scalar function can implement such increasing weighting.

Now, inspired from Appendix E.2, **Construction 2** and (E.4) that all dimensions in the output implement individual WPGD, the weighting can be a composition of up to $d+1$ different weights ($\boldsymbol{u}$ is $(d+1)$-dimensional). Therefore, for any $\omega_1,\cdots,\omega_K\in\mathbb{R}$, we can get K separate weighting:

$$\begin{aligned}
&\boldsymbol{\omega}_1=\omega_1[\mathbf{1}_{n_1}^\top \ \cdots \ \mathbf{1}_{n_K}^\top]^\top,\\
&\boldsymbol{\omega}_2=(\omega_2-\omega_1)[\mathbf{0}_{n_1}^\top \ \mathbf{1}_{n_2}^\top \ \cdots \ \mathbf{1}_{n_K}^\top]^\top,\\
&\boldsymbol{\omega}_3=(\omega_3-\omega_2)[\mathbf{0}_{n_1}^\top \ \mathbf{0}_{n_2}^\top \ \mathbf{1}_{n_3}^\top \ \cdots \ \mathbf{1}_{n_K}^\top]^\top,\\
&\vdots\\
&\boldsymbol{\omega}_K=(\omega_K-\omega_{K-1})[\mathbf{0}_{n_1}^\top \ \mathbf{0}_{n_2}^\top \ \mathbf{0}_{n_3}^\top \ \cdots \ \mathbf{0}_{n_{K-1}}^\top \ \mathbf{1}_{n_K}^\top]^\top.
\end{aligned}$$

Recap from Appendix E.2 and (E.4), and consider the construction $\tilde{\boldsymbol{W}}_v = [\mathbf{0}_{(d+p+1)\times d} \ \ \boldsymbol{u} \ \ \mathbf{0}_{(d+p+1)\times p}]^\top$. Assumption 6.2 implies that $K\le p<d+p+1$.

From (E.3) and (E.4), let i'th dimension implements the weighting $\boldsymbol{\omega}_i$ for $i\in[K]$. Specifically, let i'th row of each gating matrix $\boldsymbol{G}_i$ implement weighting $[\mathbf{0}_{n_1}^\top \ \cdots \ \mathbf{0}_{n_{i-1}}^\top \ \mathbf{1}_{n_i}^\top \ \cdots \ \mathbf{1}_{n_k}^\top]$ (which is feasible due to Theorem 6.5) and set $u_ih_i=\omega_i-\omega_{i-1}$ with $\omega_0=0$. Then the

composed weighting following (E.4) returns $\boldsymbol{\omega} = \sum_{k=1}^{K} \boldsymbol{\omega}_k$, which completes the proof. ■

E.4.3 Proof of Corollary 6.3

Proof. Recap from (E.17a) that given $\boldsymbol{\Sigma} = \boldsymbol{I}$ and $\boldsymbol{\omega} = \mathbf{1}$,

$$\begin{aligned}\boldsymbol{P}^{\star} &= \left((d+\sigma^2)n\boldsymbol{I} + (n + \mathbf{1}^{\top}\boldsymbol{R}\mathbf{1})\boldsymbol{I}\right)^{-1}\mathbf{1}^{\top}\boldsymbol{r} \\ &= \frac{\mathbf{1}^{\top}\boldsymbol{r}}{n(d+\sigma^2+1) + \mathbf{1}^{\top}\boldsymbol{R}\mathbf{1}}\boldsymbol{I} := c\boldsymbol{I}.\end{aligned}$$

Then taking it back to the loss function (c.f. (E.12)) obtains

$$\begin{aligned}\mathcal{L}(\boldsymbol{P}^{\star}, \boldsymbol{\omega} = 1) &= d + \sigma^2 - 2cd\mathbf{1}^{\top}\boldsymbol{r} + (d+\sigma^2)c^2nd + (n + \mathbf{1}^{\top}\boldsymbol{R}\mathbf{1})c^2d \\ &= d + \sigma^2 - cd\mathbf{1}^{\top}\boldsymbol{r}.\end{aligned}$$

It completes the proof. ■

APPENDIX F

Appendix for Chapter 7

F.1 Lemmas

Lemma F.1 *Suppose $\boldsymbol{X}$ is a $n \times d$ matrix, each column of which is independently drawn from a d-variate Gaussian distribution with zero mean:*

$$\boldsymbol{X} = [\boldsymbol{x}_1 \ \ldots \ \boldsymbol{x}_n]^\top, \text{ where } \boldsymbol{x}_i \sim \mathcal{N}(0, \boldsymbol{\Sigma_x}) \in \mathbb{R}^d, i \in [n].$$

For a constant matrix $\boldsymbol{A} \in \mathbb{R}^{d \times d}$, the following expectation can be determined by:

$$\mathbb{E}\left[\boldsymbol{X}^\top \boldsymbol{X} \boldsymbol{A} \boldsymbol{X}^\top \boldsymbol{X}\right] = n\,\boldsymbol{tr}(\boldsymbol{\Sigma_x A})\,\boldsymbol{\Sigma_x} + n(n+1)\boldsymbol{\Sigma_x A \Sigma_x}.$$

Proof. We begin by expressing $\boldsymbol{X}^\top \boldsymbol{X}$ as a sum over its columns:

$$\boldsymbol{X}^\top \boldsymbol{X} = \sum_{i=1}^{n} \boldsymbol{x}_i \boldsymbol{x}_i^\top.$$

Therefore,

$$\boldsymbol{X}^\top \boldsymbol{X} \boldsymbol{A} \boldsymbol{X}^\top \boldsymbol{X} = \left(\sum_{i=1}^{n} \boldsymbol{x}_i \boldsymbol{x}_i^\top\right) \boldsymbol{A} \left(\sum_{j=1}^{n} \boldsymbol{x}_j \boldsymbol{x}_j^\top\right) = \sum_{i=1}^{n} \sum_{j=1}^{n} \boldsymbol{x}_i \boldsymbol{x}_i^\top \boldsymbol{A} \boldsymbol{x}_j \boldsymbol{x}_j^\top.$$

Taking expectations on both sides, we have:

$$\mathbb{E}[\boldsymbol{X}^\top \boldsymbol{X} \boldsymbol{A} \boldsymbol{X}^\top \boldsymbol{X}] = \sum_{i=1}^{n} \sum_{j=1}^{n} \mathbb{E}[\boldsymbol{x}_i \boldsymbol{x}_i^\top \boldsymbol{A} \boldsymbol{x}_j \boldsymbol{x}_j^\top].$$

Since the vectors $\boldsymbol{x}_i$ are independent and identically distributed, we can split the sum into

terms where $i = j$ and $i \neq j$:

$$\mathbb{E}[\boldsymbol{X}^\top \boldsymbol{X} \boldsymbol{A} \boldsymbol{X}^\top \boldsymbol{X}] = \sum_{i=1}^{n} \mathbb{E}[\boldsymbol{x}_i \boldsymbol{x}_i^\top \boldsymbol{A} \boldsymbol{x}_i \boldsymbol{x}_i^\top] + \sum_{i \neq j} \mathbb{E}[\boldsymbol{x}_i \boldsymbol{x}_i^\top \boldsymbol{A} \boldsymbol{x}_j \boldsymbol{x}_j^\top].$$

For $i \neq j$, independence implies:

$$\mathbb{E}[\boldsymbol{x}_i \boldsymbol{x}_i^\top \boldsymbol{A} \boldsymbol{x}_j \boldsymbol{x}_j^\top] = \mathbb{E}[\boldsymbol{x}_i \boldsymbol{x}_i^\top] \boldsymbol{A} \, \mathbb{E}[\boldsymbol{x}_j \boldsymbol{x}_j^\top] = \boldsymbol{\Sigma}_{\boldsymbol{x}} \boldsymbol{A} \boldsymbol{\Sigma}_{\boldsymbol{x}}.$$

There are $n(n-1)$ such terms. For $i = j$, we compute:

$$\mathbb{E}[\boldsymbol{x}_i \boldsymbol{x}_i^\top \boldsymbol{A} \boldsymbol{x}_i \boldsymbol{x}_i^\top] = \mathbb{E}[(\boldsymbol{x}_i^\top \boldsymbol{A} \boldsymbol{x}_i) \boldsymbol{x}_i \boldsymbol{x}_i^\top].$$

Using Isserlis' theorem for zero-mean Gaussian vectors, we have:

$$\mathbb{E}[(\boldsymbol{x}^\top \boldsymbol{A} \boldsymbol{x}) \boldsymbol{x} \boldsymbol{x}^\top] = \mathtt{tr}\left(\boldsymbol{A} \boldsymbol{\Sigma}_{\boldsymbol{x}}\right) \boldsymbol{\Sigma}_{\boldsymbol{x}} + 2 \boldsymbol{\Sigma}_{\boldsymbol{x}} \boldsymbol{A} \boldsymbol{\Sigma}_{\boldsymbol{x}}.$$

Thus, summing over n terms:

$$\sum_{i=1}^{n} \mathbb{E}[\boldsymbol{x}_i \boldsymbol{x}_i^\top \boldsymbol{A} \boldsymbol{x}_i \boldsymbol{x}_i^\top] = n \left(\mathtt{tr}\left(\boldsymbol{A} \boldsymbol{\Sigma}_{\boldsymbol{x}}\right) \boldsymbol{\Sigma}_{\boldsymbol{x}} + 2 \boldsymbol{\Sigma}_{\boldsymbol{x}} \boldsymbol{A} \boldsymbol{\Sigma}_{\boldsymbol{x}}\right).$$

Adding all terms together:

$$\begin{aligned}
\mathbb{E}[\boldsymbol{X}^\top \boldsymbol{X} \boldsymbol{A} \boldsymbol{X}^\top \boldsymbol{X}] &= n \left(\mathtt{tr}\left(\boldsymbol{A} \boldsymbol{\Sigma}_{\boldsymbol{x}}\right) \boldsymbol{\Sigma}_{\boldsymbol{x}} + 2 \boldsymbol{\Sigma}_{\boldsymbol{x}} \boldsymbol{A} \boldsymbol{\Sigma}_{\boldsymbol{x}}\right) + n(n-1) \boldsymbol{\Sigma}_{\boldsymbol{x}} \boldsymbol{A} \boldsymbol{\Sigma}_{\boldsymbol{x}} \\
&= n \mathtt{tr}\left(\boldsymbol{A} \boldsymbol{\Sigma}_{\boldsymbol{x}}\right) \boldsymbol{\Sigma}_{\boldsymbol{x}} + \left(2n + n(n-1)\right) \boldsymbol{\Sigma}_{\boldsymbol{x}} \boldsymbol{A} \boldsymbol{\Sigma}_{\boldsymbol{x}} \\
&= n \mathtt{tr}\left(\boldsymbol{A} \boldsymbol{\Sigma}_{\boldsymbol{x}}\right) \boldsymbol{\Sigma}_{\boldsymbol{x}} + n(n+1) \boldsymbol{\Sigma}_{\boldsymbol{x}} \boldsymbol{A} \boldsymbol{\Sigma}_{\boldsymbol{x}}.
\end{aligned}$$

This completes the proof. ∎

Lemma F.2 *Suppose $\boldsymbol{X}$ is a $n \times d$ matrix, each column of which is independently drawn from a d-variate Gaussian distribution with zero mean:*

$$\boldsymbol{X} = [\boldsymbol{x}_1 \ \ldots \ \boldsymbol{x}_n]^\top, \text{ where } \boldsymbol{x}_i \sim \mathcal{N}(0, \boldsymbol{\Sigma}_{\boldsymbol{x}}) \in \mathbb{R}^d, i \in [n].$$

For a zero-mean Gaussian variable sampled independently $\boldsymbol{\xi} \sim \mathcal{N}(0, \sigma^2 \boldsymbol{I}_n) \in \mathbb{R}^n$, the following expectation moment can be determined by:

$$\mathbb{E}\left[\boldsymbol{X}^\top \boldsymbol{\xi} \boldsymbol{\xi}^\top \boldsymbol{X}\right] = n \sigma^2 \boldsymbol{\Sigma}_{\boldsymbol{x}}.$$

Proof. Using the independence of $\boldsymbol{X}$ and the entries of $\boldsymbol{\xi} = [\xi_1 \ \cdots \ \xi_n]^\top$, $\mathbb{E}[\xi_i \xi_j] = \sigma^2 \delta_{ij}$ (where δ_{ij} is the Kronecker delta, equal to 1 if $i = j$ and 0 otherwise):

$$\begin{aligned}\mathbb{E}\left[\boldsymbol{X}^\top \boldsymbol{\xi}\boldsymbol{\xi}^\top \boldsymbol{X}\right] &= \mathbb{E}\left[\sum_{i=1}^{n}\sum_{j=1}^{n} \xi_i \xi_j \boldsymbol{x}_i \boldsymbol{x}_j^\top\right] \\ &= \sum_{i=1}^{n} \mathbb{E}[\xi_i \xi_i]\, \mathbb{E}[\boldsymbol{x}_i \boldsymbol{x}_i^\top] \\ &= n\sigma^2 \boldsymbol{\Sigma}_{\boldsymbol{x}}.\end{aligned}$$

■

Lemma F.3 *Let $\boldsymbol{W} \in \mathbb{R}^{d\times d}$, and $\boldsymbol{A}, \boldsymbol{B} \in \mathbb{R}^{d\times d}$ be constant matrices. Then,*

$$\frac{\partial}{\partial \boldsymbol{W}} \mathtt{tr}\left(\boldsymbol{W}\boldsymbol{A}\boldsymbol{W}^\top \boldsymbol{B}\right) = \boldsymbol{B}^\top \boldsymbol{W}\boldsymbol{A}^\top + \boldsymbol{B}\boldsymbol{W}\boldsymbol{A}.$$

Proof. We will compute the derivative $\frac{\partial}{\partial \boldsymbol{W}} \mathtt{tr}\left(\boldsymbol{W}\boldsymbol{A}\boldsymbol{W}^\top \boldsymbol{B}\right)$ using an element-wise approach.

First, expand the trace function:

$$\mathtt{tr}\left(\boldsymbol{W}\boldsymbol{A}\boldsymbol{W}^\top \boldsymbol{B}\right) = \sum_{i=1}^{d} (\boldsymbol{W}\boldsymbol{A}\boldsymbol{W}^\top \boldsymbol{B})_{ii}.$$

Using the definition of matrix multiplication, we have:

$$(\boldsymbol{W}\boldsymbol{A})_{ij} = \sum_{k=1}^{d} W_{ik} A_{kj},$$

$$(\boldsymbol{W}^\top \boldsymbol{B})_{ji} = \sum_{l=1}^{d} W_{lj} B_{li}.$$

Therefore,

$$(\boldsymbol{W}\boldsymbol{A}\boldsymbol{W}^\top \boldsymbol{B})_{ii} = \sum_{j=1}^{d} (\boldsymbol{W}\boldsymbol{A})_{ij} (\boldsymbol{W}^\top \boldsymbol{B})_{ji} = \sum_{j=1}^{d} \left(\sum_{k=1}^{d} W_{ik} A_{kj}\right)\left(\sum_{l=1}^{d} W_{lj} B_{li}\right).$$

Thus, the trace becomes:

$$\texttt{tr}\left(\boldsymbol{W}\boldsymbol{A}\boldsymbol{W}^\top\boldsymbol{B}\right) = \sum_{i=1}^{d}\sum_{j=1}^{d}\sum_{k=1}^{d}\sum_{l=1}^{d} W_{ik}A_{kj}W_{lj}B_{li}.$$

We need to compute the derivative with respect to W_{pq}:

$$\frac{\partial}{\partial W_{pq}}\texttt{tr}\left(\boldsymbol{W}\boldsymbol{A}\boldsymbol{W}^\top\boldsymbol{B}\right) = \sum_{i=1}^{d}\sum_{j=1}^{d}\sum_{k=1}^{d}\sum_{l=1}^{d} \frac{\partial}{\partial W_{pq}}\left(W_{ik}A_{kj}W_{lj}B_{li}\right).$$

Note that A_{kj} and B_{li} are constants.

We have:

$$\frac{\partial}{\partial W_{pq}}W_{ik} = \delta_{ip}\delta_{kq},$$

$$\frac{\partial}{\partial W_{pq}}W_{lj} = \delta_{lp}\delta_{jq},$$

where δ_{ij} is the Kronecker delta, equal to 1 if $i = j$ and 0 otherwise.

Therefore,

$$\frac{\partial}{\partial W_{pq}}\left(W_{ik}A_{kj}W_{lj}B_{li}\right) = \left(\delta_{ip}\delta_{kq}W_{lj} + W_{ik}\delta_{lp}\delta_{jq}\right)A_{kj}B_{li}.$$

Then:

$$\begin{aligned}
\frac{\partial}{\partial W_{pq}}\texttt{tr}\left(\boldsymbol{W}\boldsymbol{A}\boldsymbol{W}^\top\boldsymbol{B}\right) &= \sum_{i=1}^{d}\sum_{j=1}^{d}\sum_{k=1}^{d}\sum_{l=1}^{d}\frac{\partial}{\partial W_{pq}}\left(W_{ik}A_{kj}W_{lj}B_{li}\right)\\
&= \sum_{i=1}^{d}\sum_{j=1}^{d}\sum_{k=1}^{d}\sum_{l=1}^{d}\left(\delta_{ip}\delta_{kq}W_{lj} + W_{ik}\delta_{lp}\delta_{jq}\right)A_{kj}B_{li}\\
&= \sum_{j=1}^{d}\sum_{l=1}^{d}W_{lj}A_{qj}B_{lp} + \sum_{i=1}^{d}\sum_{k=1}^{d}W_{ik}A_{kq}B_{pi}\\
&= \left(\boldsymbol{A}\boldsymbol{W}^\top\boldsymbol{B}\right)_{qp} + \left(\boldsymbol{B}\boldsymbol{W}\boldsymbol{A}\right)_{pq}\\
&= \left(\boldsymbol{B}^\top\boldsymbol{W}\boldsymbol{A}^\top + \boldsymbol{B}\boldsymbol{W}\boldsymbol{A}\right)_{pq}
\end{aligned}$$

Since this holds for all elements (p, q), in matrix form, we have:

$$\frac{\partial}{\partial \boldsymbol{W}}\texttt{tr}\left(\boldsymbol{W}\boldsymbol{A}\boldsymbol{W}^\top\boldsymbol{B}\right) = \boldsymbol{B}^\top\boldsymbol{W}\boldsymbol{A}^\top + \boldsymbol{B}\boldsymbol{W}\boldsymbol{A}.$$

■

Lemma F.4 (Reduced form) *Denote the output sequence of the attention layer as* $att(\boldsymbol{Z})$.

Then under Assumption 7.1, the output becomes

Proof.

$$\begin{aligned}
\hat{y} &= \mathrm{att}(\boldsymbol{Z})_{(n+1,d+1)} = \boldsymbol{e}_{n+1}^\top \mathrm{att}(\boldsymbol{Z})\boldsymbol{e}_{d+1} \\
&= \boldsymbol{e}_{n+1}^\top (\boldsymbol{Z}\boldsymbol{W}_q\boldsymbol{W}_k^\top(\boldsymbol{Z})^\top)\boldsymbol{M}\boldsymbol{Z}\boldsymbol{W}_v\boldsymbol{e}_{d+1} \\
&= (\boldsymbol{e}_{n+1}^\top \boldsymbol{Z})\underbrace{(\boldsymbol{W}_q\boldsymbol{W}_k^\top)}_{=\boldsymbol{A}}(\boldsymbol{Z}^\top\boldsymbol{M}\boldsymbol{Z})\underbrace{\boldsymbol{W}_v\boldsymbol{e}_{d+1}}_{=\boldsymbol{a}} \\
&= (\boldsymbol{e}_{n+1}^\top \boldsymbol{Z})\boldsymbol{A}(\boldsymbol{Z}^\top\boldsymbol{M}\boldsymbol{Z})\boldsymbol{a} \\
&= \begin{bmatrix}\boldsymbol{x}\\ 0\end{bmatrix}^\top \boldsymbol{A}\begin{bmatrix}\boldsymbol{X}^\top\boldsymbol{X} & \boldsymbol{X}^\top\boldsymbol{y}\\ \boldsymbol{y}^\top\boldsymbol{X} & \boldsymbol{y}^\top\boldsymbol{y}\end{bmatrix}\boldsymbol{a}
\end{aligned}$$

Under Assumption 7.1

$$\boldsymbol{A} = \begin{bmatrix}\boldsymbol{W}_{d\times d} & \boldsymbol{0}_{d\times 1}\\ *_{1\times d} & *_{1\times 1}\end{bmatrix}, \boldsymbol{a} = \begin{bmatrix}\boldsymbol{0}_{d\times 1}\\ 1_{1\times 1}\end{bmatrix},$$

The output finally reduced to

$$\begin{aligned}
\hat{y} &= \begin{bmatrix}\boldsymbol{x}\\ 0\end{bmatrix}^\top \begin{bmatrix}\boldsymbol{W} & \boldsymbol{0}\\ * & *\end{bmatrix}\begin{bmatrix}\boldsymbol{X}^\top\boldsymbol{X} & \boldsymbol{X}^\top\boldsymbol{y}\\ \boldsymbol{y}^\top\boldsymbol{X} & \boldsymbol{y}^\top\boldsymbol{y}\end{bmatrix}\begin{bmatrix}\boldsymbol{0}\\ 1\end{bmatrix} \\
&= \boldsymbol{x}^\top\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{y}.
\end{aligned}$$

■

F.2 Proofs for Section 7.3

We consider an in-context learning (ICL) problem with demonstrations $(\boldsymbol{x}_i, y_i)_{i=1}^{n+1}$, and the input sequence $\boldsymbol{Z}$ is defined by removing y_{n+1} as follows:

$$\boldsymbol{Z} = [\boldsymbol{z}_1\ \ldots\ \boldsymbol{z}_n\ \boldsymbol{z}]^\top = \begin{bmatrix}\boldsymbol{x}_1 & \ldots & \boldsymbol{x}_n & \boldsymbol{x}\\ y_1 & \ldots & y_n & 0\end{bmatrix}^\top = \begin{bmatrix}\boldsymbol{X}^\top & \boldsymbol{x}\\ \boldsymbol{y}^\top & 0\end{bmatrix}^\top \in \mathbb{R}^{(n+1)\times(d+1)}. \tag{F.1}$$

Here, $\boldsymbol{z} = [\boldsymbol{x}^\top\ 0]^\top$ is the query token where $\boldsymbol{x} := \boldsymbol{x}_{n+1}$, and $\boldsymbol{X} = [\boldsymbol{x}_1\ \cdots\ \boldsymbol{x}_n]^\top \in \mathbb{R}^{n\times d}$, $\boldsymbol{y} = [y_1\ \cdots\ y_n]^\top \in \mathbb{R}^n$. Then, we aim for a sequence model to predict the associated label $y := y_{n+1}$ of the given input sequence $\boldsymbol{Z}$. In this work, we consider the following data generation of $(\boldsymbol{Z}, y)$. We will refer to $(\boldsymbol{X}, \boldsymbol{y})$, $\boldsymbol{x}$, and y as contexts, query feature and the label to predict, respectively.

Definition F.1 (Single-task ICL) *Given a task mean $\boldsymbol{\mu} \in \mathbb{R}^d$, and covariances $\boldsymbol{\Sigma}_{\boldsymbol{x}}, \boldsymbol{\Sigma}_{\boldsymbol{\beta}} \succ 0 \in \mathbb{R}^{d\times d}$. The input sequence and its associated label, i.e., $(\boldsymbol{Z}, y)$ with $\boldsymbol{Z}$ denoted in* (F.1)*,*

are generated as follows:

- *A task parameter* $\boldsymbol{\beta}$ *is generated from a Gaussian prior* $\boldsymbol{\beta} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma}_{\boldsymbol{\beta}})$.
- *Conditioned on* $\boldsymbol{\beta}$*, for* $i \in [n+1]$*,* $(\boldsymbol{x}_i, y_i)$ *is generated by* $\boldsymbol{x}_i \sim \mathcal{N}(0, \boldsymbol{\Sigma}_{\boldsymbol{x}})$ *and* $y_i \sim \mathcal{N}(\boldsymbol{x}_i^\top \boldsymbol{\beta}, \sigma^2)$.

Here, $\sigma \geq 0$ is the noise level.

In a noisy label setting, the labels y_i in the input sequence $\boldsymbol{Z}$ can be obtained by

$$y_i = \boldsymbol{x}_i^\top \boldsymbol{\beta} + \xi_i, \text{ where } \xi_i \sim \mathcal{N}(0, \sigma^2), i \in [n+1].$$

Thus, the labels in the contexts and the label to predict can be obtained by:

$$\begin{aligned} \boldsymbol{y} &= \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\xi}, \text{ where } \boldsymbol{\xi} = [\xi_1 \ \cdots \ \xi_n]^\top \in \mathbb{R}^n, \\ y &= \boldsymbol{x}^\top \boldsymbol{\beta} + \xi_{n+1}. \end{aligned}$$

Definition F.2 (Multi-task ICL) *Consider a multi-task ICL problem with* K *different tasks. Each task generates* $(\boldsymbol{Z}, y) \sim \mathcal{D}_k$ *following Definition F.1 using shared feature distribution* $\boldsymbol{x}_i \sim \mathcal{N}(0, \boldsymbol{\Sigma}_{\boldsymbol{x}})$*,* $i \in [n+1]$ *but distinct task distributions* $\boldsymbol{\beta}_k \sim \mathcal{N}(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_{\boldsymbol{\beta}_k})$ *with mean* $\boldsymbol{\mu}_k$ *and covariance* $\boldsymbol{\Sigma}_{\boldsymbol{\beta}_k}$ *for* $k \in [K]$.

Additionally, let $\{\pi_k\}_{k=1}^K$ *be the probabilities of each task, satisfying* $\sum_{k=1}^K \pi_k = 1$ *and* $\pi_k \geq 0$.

We consider a task-aware multi-task ICL setting. Specifically, when a task is selected according to π_k, its task index k is known. Let $\bar{\mathcal{D}} := \sum_{k=1}^K \pi_k \mathcal{D}_k$ be the mixture of distributions and given sequence model $f : \mathbb{R}^{(n+1)\times(d+1)} \to \mathbb{R}$, we define the multi-task ICL objective as follows:

$$\mathcal{L}(f) = \mathbb{E}_{(\boldsymbol{Z}, y)\sim\bar{\mathcal{D}}}[(y - f(\boldsymbol{Z}))^2]. \tag{F.2}$$

To start with, recap from Definition 7.2 where task k has probability π_k and its task vector follows distribution $\boldsymbol{\beta}_k \sim \mathcal{N}(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_{\boldsymbol{\beta}_k})$. Following (7.13) in the main paper, we define the debiased and biased mixed-task covariances (variant with $\boldsymbol{\Sigma}_{\boldsymbol{x}}$ prior) as follows:

$$\text{Debiased: } \bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} = \boldsymbol{\Sigma}_{\boldsymbol{x}} \sum_{k=1}^K \pi_k \, \mathbb{E}[(\boldsymbol{\beta}_k - \boldsymbol{\mu}_k)(\boldsymbol{\beta}_k - \boldsymbol{\mu}_k)^\top]; \tag{F.3a}$$

$$\text{Biased:} \quad \tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} = \boldsymbol{\Sigma}_{\boldsymbol{x}} \sum_{k=1}^K \pi_k \, \mathbb{E}[\boldsymbol{\beta}_k \boldsymbol{\beta}_k^\top]. \tag{F.3b}$$

Note that they satisfy $\bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} = \boldsymbol{\Sigma}_{\boldsymbol{x}} \sum_{k=1}^{K} \pi_k \boldsymbol{\Sigma}_{\boldsymbol{\beta}_k}$ and $\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} = \boldsymbol{\Sigma}_{\boldsymbol{x}} \sum_{k=1}^{K} \pi_k (\boldsymbol{\Sigma}_{\boldsymbol{\beta}_k} + \boldsymbol{\mu}_k \boldsymbol{\mu}_k^\top)$.

We first analyze the plain training setting where no additional task-specific parameters are introduced, and all K tasks are mixed together.

Under Assumption 7.1 in the main paper, let the prompt token for task k be $\boldsymbol{p}_k = \begin{bmatrix} \bar{\boldsymbol{p}}_k \\ 1 \end{bmatrix}$. The prediction of a single-layer linear attention model can then be written as:

$$\begin{aligned} f(\boldsymbol{Z}^{(k)}) &= \boldsymbol{x}^\top \boldsymbol{W} \begin{bmatrix} \bar{\boldsymbol{p}}_k & \boldsymbol{X}^\top \end{bmatrix} \begin{bmatrix} 1 \\ \boldsymbol{y} \end{bmatrix} \\ &= \boldsymbol{x}^\top \boldsymbol{W} (\boldsymbol{X}^\top \boldsymbol{y} + \bar{\boldsymbol{p}}_k) := g(\boldsymbol{x}, \boldsymbol{X}, \boldsymbol{y}; \boldsymbol{W}, \bar{\boldsymbol{p}}_k). \end{aligned} \tag{F.4}$$

Note that for the multi-task ICL with task-specific prompting, the optimization object (F.2) can be denoted as:

$$\begin{aligned} \mathcal{L}(f) &= \sum_{k=1}^{K} \pi_k \underbrace{\mathbb{E}_{(\boldsymbol{Z},y)\sim\mathcal{D}_k} \left[(y - f(\boldsymbol{Z}^{(k)}))^2 \right]}_{\text{Denoted as } \mathcal{L}_k(f)} \\ &= \sum_{k=1}^{K} \pi_k \mathcal{L}_k(f) = \sum_{k=1}^{K} \pi_k \, \mathbb{E}_{(\boldsymbol{Z},y)\sim\mathcal{D}_k} \left[(y - g(\boldsymbol{x}, \boldsymbol{X}, \boldsymbol{y}; \boldsymbol{W}, \bar{\boldsymbol{p}}_k))^2 \right] := \sum_{k=1}^{K} \pi_k \mathcal{L}_k(\boldsymbol{W}, \bar{\boldsymbol{p}}_k), \end{aligned}$$

where $\boldsymbol{Z} = \begin{bmatrix} \boldsymbol{X}^\top & \boldsymbol{x} \\ \boldsymbol{y}^\top & 0 \end{bmatrix}^\top \in \mathbb{R}^{(n+1)\times(d+1)}$, $\boldsymbol{Z}^{(k)} = \begin{bmatrix} \bar{\boldsymbol{p}}_k & \boldsymbol{X}^\top & \boldsymbol{x} \\ 1 & \boldsymbol{y}^\top & 0 \end{bmatrix}^\top \in \mathbb{R}^{(n+2)\times(d+1)}$.

Note that the multi-task ICL loss is equivalent to calculating the weighted sum of the task-k ICL losses over all tasks $k \in [K]$.

We begin by deriving the single-task ICL loss $\mathcal{L}_k(f)$ for task k, and then generalize it to the multi-task ICL loss by taking a weighted sum. Since the derivation is similar for all tasks, the task index k is omitted in the following derivation for simplicity. Unless otherwise specified, $\mathcal{L}_k(\boldsymbol{W}, \bar{\boldsymbol{p}}_k)$ and $\mathcal{L}(\boldsymbol{W}, \bar{\boldsymbol{p}})$ will represent the same meaning. This convention similarly applies to other task-specific parameters, e.g., $\boldsymbol{\Sigma}_{\boldsymbol{\beta}} \leftrightarrow \boldsymbol{\Sigma}_{\boldsymbol{\beta}_k}$, $\boldsymbol{\mu} \leftrightarrow \boldsymbol{\mu}_k$, etc.

The loss on a certain task with a trainable prompt can be determined by:

$$\begin{aligned}
\mathcal{L}(\boldsymbol{W}, \bar{\boldsymbol{p}}) &= \mathbb{E}\left[(y - g(\boldsymbol{x}, \boldsymbol{X}, \boldsymbol{y}; \boldsymbol{W}, \bar{\boldsymbol{p}}))^2\right] \\
&= \mathbb{E}\left[\left(\boldsymbol{x}^\top \boldsymbol{W}\boldsymbol{X}^\top \boldsymbol{y} + \boldsymbol{x}^\top \boldsymbol{W}\bar{\boldsymbol{p}} - \boldsymbol{x}^\top \boldsymbol{\beta} - \xi_{n+1}\right)^2\right] \\
&= \mathbb{E}\left[\left(\boldsymbol{x}^\top \boldsymbol{W}\boldsymbol{X}^\top (\boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\xi}) + \boldsymbol{x}^\top \boldsymbol{W}\bar{\boldsymbol{p}} - \boldsymbol{x}^\top \boldsymbol{\beta} - \xi_{n+1}\right)^2\right] \\
&= \mathbb{E}\left[\left(\boldsymbol{x}^\top \underbrace{\left((\boldsymbol{W}\boldsymbol{X}^\top \boldsymbol{X} - \boldsymbol{I})(\bar{\boldsymbol{\theta}} + \boldsymbol{\mu}) + \boldsymbol{W}\boldsymbol{X}^\top \boldsymbol{\xi} + \boldsymbol{W}\bar{\boldsymbol{p}}\right)}_{\text{Denoted as a (task-specific) vector } \boldsymbol{c}}\right)^2\right] + \sigma^2 \\
&= \mathbb{E}\left[\left(\boldsymbol{x}^\top \boldsymbol{c}\right)^2\right] + \sigma^2 = \mathtt{tr}\left(\mathbb{E}[\boldsymbol{c}\boldsymbol{c}^\top]\boldsymbol{\Sigma}_{\boldsymbol{x}}\right) + \sigma^2, \qquad \text{(F.5)}
\end{aligned}$$

where $\bar{\boldsymbol{\theta}} = \boldsymbol{\beta} - \boldsymbol{\mu} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\Sigma}_{\boldsymbol{\beta}})$ is a centralized variable.

F.2.1 Proof of Theorem 7.1

Theorem F.1 (Plain training) *Consider training a single-layer linear attention model in solving multi-task ICL problem with dataset defined in Definition 7.2 and model construction as described in Assumption 7.1. Let the optimal solution $\boldsymbol{W}_{pt}^\star$ (c.f. (7.10) in the main paper) and the minimal plain training loss $\mathcal{L}_{pt}^\star$ as defined in Section 7.2.3. Additionally, let $\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}$ be defined in (7.14) in the main paper and $\bar{\boldsymbol{W}}_{pt}^\star = \boldsymbol{\Sigma}_{\boldsymbol{x}} \boldsymbol{W}_{pt}^\star$.*

Then the solution $\bar{\boldsymbol{W}}_{pt}^\star$ and optimal loss $\mathcal{L}_{pt}^\star$ satisfy

$$\bar{\boldsymbol{W}}_{pt}^\star = \tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} \left((n+1)\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} + (\mathtt{tr}\left(\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}\right) + \sigma^2)\boldsymbol{I}\right)^{-1},$$

$$\mathcal{L}_{pt}^\star = \mathtt{tr}\left(\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}\right) + \sigma^2 - n\,\mathtt{tr}\left(\bar{\boldsymbol{W}}_{pt}^\star \tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}\right).$$

Proof. As previously stated, the following derivation applies to all tasks $k \in [K]$. Therefore, for simplicity, we omit the index k in the notation unless otherwise specified.

In the plain training setting, $\bar{\boldsymbol{p}} = \boldsymbol{0}$ for all tasks, and only the attention model, parameterized by $\boldsymbol{W}$ under Assumption 7.1, is updated. Hence, the loss in (F.5) is:

$$\mathcal{L}(\boldsymbol{W}, \bar{\boldsymbol{p}} = \boldsymbol{0}) = \mathtt{tr}\left(\mathbb{E}[\boldsymbol{c}\boldsymbol{c}^\top]\boldsymbol{\Sigma}_{\boldsymbol{x}}\right) + \sigma^2, \text{ where } \boldsymbol{c} = \left((\boldsymbol{W}\boldsymbol{X}^\top \boldsymbol{X} - \boldsymbol{I})(\bar{\boldsymbol{\theta}} + \boldsymbol{\mu}) + \boldsymbol{W}\boldsymbol{X}^\top \boldsymbol{\xi}\right).$$

The expansion of $\boldsymbol{c}\boldsymbol{c}^\top$ is (there are $3 \times 3 = 9$ terms in total):

$$
\begin{aligned}
\boldsymbol{c}\boldsymbol{c}^\top =& \left((\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{X} - \boldsymbol{I})(\bar{\boldsymbol{\theta}} + \boldsymbol{\mu}) + \boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{\xi}\right)\left((\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{X} - \boldsymbol{I})(\bar{\boldsymbol{\theta}} + \boldsymbol{\mu}) + \boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{\xi}\right)^\top \\
=& \left((\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{X})(\bar{\boldsymbol{\theta}} + \boldsymbol{\mu}) - (\bar{\boldsymbol{\theta}} + \boldsymbol{\mu}) + \boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{\xi}\right)\left((\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{X})(\bar{\boldsymbol{\theta}} + \boldsymbol{\mu}) - (\bar{\boldsymbol{\theta}} + \boldsymbol{\mu}) + \boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{\xi}\right)^\top \\
=& \left[(\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{X})(\bar{\boldsymbol{\theta}} + \boldsymbol{\mu})\right]\left[(\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{X})(\bar{\boldsymbol{\theta}} + \boldsymbol{\mu})\right]^\top + \left[(\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{X})(\bar{\boldsymbol{\theta}} + \boldsymbol{\mu})\right]\left[-(\bar{\boldsymbol{\theta}} + \boldsymbol{\mu})\right]^\top + \left[(\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{X})(\bar{\boldsymbol{\theta}} + \boldsymbol{\mu})\right]\left[\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{\xi}\right]^\top \\
&+ \left[-(\bar{\boldsymbol{\theta}} + \boldsymbol{\mu})\right]\left[(\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{X})(\bar{\boldsymbol{\theta}} + \boldsymbol{\mu})\right]^\top + \left[-(\bar{\boldsymbol{\theta}} + \boldsymbol{\mu})\right]\left[-(\bar{\boldsymbol{\theta}} + \boldsymbol{\mu})\right]^\top + \left[-(\bar{\boldsymbol{\theta}} + \boldsymbol{\mu})\right]\left[\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{\xi}\right]^\top \\
&+ \left[\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{\xi}\right]\left[(\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{X})(\bar{\boldsymbol{\theta}} + \boldsymbol{\mu})\right]^\top + \left[\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{\xi}\right]\left[-(\bar{\boldsymbol{\theta}} + \boldsymbol{\mu})\right]^\top + \left[\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{\xi}\right]\left[\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{\xi}\right]^\top.
\end{aligned}
$$

Take expectation of it,

$$
\begin{aligned}
\mathbb{E}[\boldsymbol{c}\boldsymbol{c}^\top] =& \left[\boldsymbol{W} \quad \underbrace{\mathbb{E}[\boldsymbol{X}^\top\boldsymbol{X}(\bar{\boldsymbol{\theta}} + \boldsymbol{\mu})(\bar{\boldsymbol{\theta}} + \boldsymbol{\mu})^\top\boldsymbol{X}^\top\boldsymbol{X}]}_{\text{Lemma F.1, denoted as a (task-specific) matrix } \boldsymbol{C}} \quad \boldsymbol{W}^\top\right] + \left[-n\boldsymbol{W}\boldsymbol{\Sigma}_{\boldsymbol{x}}(\boldsymbol{\Sigma}_{\boldsymbol{\beta}} + \boldsymbol{\mu}\boldsymbol{\mu}^\top)\right] + 0 \\
&+ \left[-n(\boldsymbol{\Sigma}_{\boldsymbol{\beta}} + \boldsymbol{\mu}\boldsymbol{\mu}^\top)\boldsymbol{\Sigma}_{\boldsymbol{x}}\boldsymbol{W}^\top\right] + (\boldsymbol{\Sigma}_{\boldsymbol{\beta}} + \boldsymbol{\mu}\boldsymbol{\mu}^\top) + 0 \\
&+ 0 + 0 + \underbrace{n\sigma^2\boldsymbol{W}\boldsymbol{\Sigma}_{\boldsymbol{x}}\boldsymbol{W}^\top}_{\text{Lemma F.2}} \\
=& \boldsymbol{W}(\boldsymbol{C} + n\sigma^2\boldsymbol{\Sigma}_{\boldsymbol{x}})\boldsymbol{W}^\top - n\boldsymbol{W}\boldsymbol{\Sigma}_{\boldsymbol{x}}(\boldsymbol{\Sigma}_{\boldsymbol{\beta}} + \boldsymbol{\mu}\boldsymbol{\mu}^\top) - n(\boldsymbol{\Sigma}_{\boldsymbol{\beta}} + \boldsymbol{\mu}\boldsymbol{\mu}^\top)\boldsymbol{\Sigma}_{\boldsymbol{x}}\boldsymbol{W}^\top + (\boldsymbol{\Sigma}_{\boldsymbol{\beta}} + \boldsymbol{\mu}\boldsymbol{\mu}^\top),
\end{aligned}
$$

where $\boldsymbol{C} = n\mathtt{tr}\left(\boldsymbol{\Sigma}_{\boldsymbol{x}}(\boldsymbol{\Sigma}_{\boldsymbol{\beta}} + \boldsymbol{\mu}\boldsymbol{\mu}^\top)\right)\boldsymbol{\Sigma}_{\boldsymbol{x}} + n(n+1)\boldsymbol{\Sigma}_{\boldsymbol{x}}(\boldsymbol{\Sigma}_{\boldsymbol{\beta}} + \boldsymbol{\mu}\boldsymbol{\mu}^\top)\boldsymbol{\Sigma}_{\boldsymbol{x}}$.

Substitute back into the loss $\mathcal{L}(\boldsymbol{W}, \bar{\boldsymbol{p}} = \boldsymbol{0})$:

$$
\mathcal{L}(\boldsymbol{W}, \bar{\boldsymbol{p}} = \boldsymbol{0}) = \mathtt{tr}\left(\boldsymbol{W}(\boldsymbol{C} + n\sigma^2\boldsymbol{\Sigma}_{\boldsymbol{x}})\boldsymbol{W}^\top\boldsymbol{\Sigma}_{\boldsymbol{x}}\right) - 2n\mathtt{tr}\left(\boldsymbol{\Sigma}_{\boldsymbol{x}}\boldsymbol{W}\boldsymbol{\Sigma}_{\boldsymbol{x}}(\boldsymbol{\Sigma}_{\boldsymbol{\beta}} + \boldsymbol{\mu}\boldsymbol{\mu}^\top)\right) + \mathtt{tr}\left(\boldsymbol{\Sigma}_{\boldsymbol{x}}(\boldsymbol{\Sigma}_{\boldsymbol{\beta}} + \boldsymbol{\mu}\boldsymbol{\mu}^\top)\right) + \sigma^2.
$$

Use the following definition:

$$
\begin{aligned}
\text{Debiased: } \bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} &= \boldsymbol{\Sigma}_{\boldsymbol{x}}\sum_{k=1}^{K}\pi_k\,\mathbb{E}[(\boldsymbol{\beta}_k - \boldsymbol{\mu}_k)(\boldsymbol{\beta}_k - \boldsymbol{\mu}_k)^\top] = \boldsymbol{\Sigma}_{\boldsymbol{x}}\sum_{k=1}^{K}\pi_k\boldsymbol{\Sigma}_{\boldsymbol{\beta}_k}; \\
\text{Biased: } \quad \tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} &= \boldsymbol{\Sigma}_{\boldsymbol{x}}\sum_{k=1}^{K}\pi_k\,\mathbb{E}[\boldsymbol{\beta}_k\boldsymbol{\beta}_k^\top] = \boldsymbol{\Sigma}_{\boldsymbol{x}}\sum_{k=1}^{K}\pi_k(\boldsymbol{\Sigma}_{\boldsymbol{\beta}_k} + \boldsymbol{\mu}_k\boldsymbol{\mu}_k^\top).
\end{aligned}
$$

Denote

$$
\bar{\boldsymbol{C}} = \sum_{k=1}^{K}\pi_k\boldsymbol{C}_k = n\mathtt{tr}\left(\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}\right)\boldsymbol{\Sigma}_{\boldsymbol{x}} + n(n+1)\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}\boldsymbol{\Sigma}_{\boldsymbol{x}}.
$$

The weighted sum of the k-th task plain training loss over all tasks $k \in [K]$ is:

$$\begin{aligned}\mathcal{L}_{\text{pt}}(\boldsymbol{W}, \boldsymbol{P} = \boldsymbol{0}) &= \sum_{k=1}^{K} \pi_k \mathcal{L}_k(\boldsymbol{W}, \bar{\boldsymbol{p}}_k = \boldsymbol{0}) \\ &= \texttt{tr}\left(\boldsymbol{W}(\bar{\boldsymbol{C}} + n\sigma^2 \boldsymbol{\Sigma}_{\boldsymbol{x}})\boldsymbol{W}^\top \boldsymbol{\Sigma}_{\boldsymbol{x}}\right) - 2n\texttt{tr}\left(\boldsymbol{W}\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}\boldsymbol{\Sigma}_{\boldsymbol{x}}\right) + \texttt{tr}\left(\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}\right) + \sigma^2.\end{aligned}$$

Take derivative w.r.t. $\boldsymbol{W}$, using Lemma F.3. Since $\boldsymbol{C}$ is symmetric

$$\frac{\partial \mathcal{L}_{\text{pt}}(\boldsymbol{W}, \boldsymbol{P} = \boldsymbol{0})}{\partial \boldsymbol{W}} = 2\boldsymbol{\Sigma}_{\boldsymbol{x}}\boldsymbol{W}(\bar{\boldsymbol{C}} + n\sigma^2\boldsymbol{\Sigma}_{\boldsymbol{x}}) - 2n\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}\boldsymbol{\Sigma}_{\boldsymbol{x}}$$

Let $\bar{\boldsymbol{W}}^\star_{\text{pt}} = \boldsymbol{\Sigma}_{\boldsymbol{x}}\boldsymbol{W}^\star_{\text{pt}}$, and set the derivative to zero:

$$\begin{aligned}\bar{\boldsymbol{W}}^\star_{\text{pt}} &= n\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}\boldsymbol{\Sigma}_{\boldsymbol{x}}(\bar{\boldsymbol{C}} + n\sigma^2\boldsymbol{\Sigma}_{\boldsymbol{x}})^{-1} \\ &= \tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}\boldsymbol{\Sigma}_{\boldsymbol{x}}\left((\texttt{tr}\left(\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}\right) + \sigma^2)\boldsymbol{\Sigma}_{\boldsymbol{x}} + (n+1)\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}\boldsymbol{\Sigma}_{\boldsymbol{x}}\right)^{-1} \\ &\boxed{= \tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}\left((n+1)\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} + (\texttt{tr}\left(\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}\right) + \sigma^2)\boldsymbol{I}\right)^{-1}.}\end{aligned}$$

Substitute back into $\mathcal{L}_{\text{pt}}(\boldsymbol{W}, \boldsymbol{P} = \boldsymbol{0})$:

$$\boxed{\mathcal{L}^\star_{\text{pt}} = \texttt{tr}\left(\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}\right) + \sigma^2 - n\texttt{tr}\left(\bar{\boldsymbol{W}}^\star_{\text{pt}}\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}\right).}$$

■

F.2.2 Proof of Theorem 7.2

Theorem F.2 (Fine-tuning) *Suppose a pretrained model as described in Theorem 7.1 is given with $\boldsymbol{W}^\star_{pt}$ being its optimal solution. Consider fine-tuning this model with task-specific prompts as defined in Definition 7.3, and let the optimal prompt matrix $\boldsymbol{P}^\star_{ft}$ (c.f. (7.11) in the main paper) and the minimal fine-tuning loss $\mathcal{L}^\star_{ft}$ be defined in Section 7.2.3. Additionally, let $\bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}, \tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}$ be defined in (7.14) in the main paper and $\bar{\boldsymbol{W}}^\star_{pt} = \boldsymbol{\Sigma}_{\boldsymbol{x}}\boldsymbol{W}^\star_{pt}$, and define the mean matrix*

$$\boldsymbol{M}_{\boldsymbol{\mu}} = [\boldsymbol{\mu}_1 \ \cdots \ \boldsymbol{\mu}_K]^\top \in \mathbb{R}^{K\times d}.$$

Then the solution $\boldsymbol{P}^\star_{ft}$ and optimal loss $\mathcal{L}^\star_{ft}$ satisfy

$$\boldsymbol{P}_{ft}^{\star} = \boldsymbol{M}_{\boldsymbol{\mu}} \left((\bar{\boldsymbol{W}}_{pt}^{\star})^{-1} - n\boldsymbol{I} \right) \boldsymbol{\Sigma}_{\boldsymbol{x}},$$
$$\mathcal{L}_{ft}^{\star} = \mathcal{L}_{pt}^{\star} - \mathtt{tr}\left((\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} - \bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}})(n\bar{\boldsymbol{W}}_{pt}^{\star} - \boldsymbol{I})^{\top}(n\bar{\boldsymbol{W}}_{pt}^{\star} - \boldsymbol{I}) \right).$$

Proof. As previously stated, the following derivation applies to all tasks $k \in [K]$. Therefore, for simplicity, we omit the index k in the notation unless otherwise specified.

1. Determining the optimal task-specific prompts

In the fine-tuning setting, the attention model is pretrained and parameterized by a fixed $\boldsymbol{W} = \boldsymbol{W}_{\mathrm{pt}}^{\star}$, and only the task-specific prompts are fine-tuned. Then recap from (F.5):

$$\begin{aligned} &\mathcal{L}(\boldsymbol{W}, \bar{\boldsymbol{p}}) = \mathtt{tr}\left(\mathbb{E}[\boldsymbol{c}\boldsymbol{c}^{\top}] \boldsymbol{\Sigma}_{\boldsymbol{x}} \right) + \sigma^2, \\ &\text{where} \quad \boldsymbol{c} = \left((\boldsymbol{W}\boldsymbol{X}^{\top}\boldsymbol{X} - \boldsymbol{I})(\bar{\boldsymbol{\theta}} + \boldsymbol{\mu}) + \boldsymbol{W}\boldsymbol{X}^{\top}\boldsymbol{\xi} + \boldsymbol{W}\bar{\boldsymbol{p}} \right). \end{aligned}$$

The optimal task-specific prompt is determined by taking the derivative and setting it to zero:

$$\begin{aligned} \frac{\partial \mathcal{L}(\boldsymbol{W}, \bar{\boldsymbol{p}})}{\partial \bar{\boldsymbol{p}}} &= 2\,\mathbb{E}\left[\frac{\partial \boldsymbol{c}}{\partial \bar{\boldsymbol{p}}}^{\top} \boldsymbol{\Sigma}_{\boldsymbol{x}} \boldsymbol{c} \right] \\ &= 2\boldsymbol{W}^{\top}\boldsymbol{\Sigma}_{\boldsymbol{x}}\,\mathbb{E}[\boldsymbol{c}] = 2\boldsymbol{W}^{\top}\boldsymbol{\Sigma}_{\boldsymbol{x}} \left[(n\boldsymbol{W}\boldsymbol{\Sigma}_{\boldsymbol{x}} - \boldsymbol{I})\boldsymbol{\mu} + \boldsymbol{W}\bar{\boldsymbol{p}} \right] = \boldsymbol{0}, \\ \Rightarrow \quad \bar{\boldsymbol{p}}^{\star} &= \left(\boldsymbol{W}^{-1} - n\boldsymbol{\Sigma}_{\boldsymbol{x}} \right) \boldsymbol{\mu}, \end{aligned}$$

which is equivalent to (by substituting $\boldsymbol{W} = \boldsymbol{W}_{\mathrm{pt}}^{\star}$):

$$\boxed{\boldsymbol{P}_{\mathrm{ft}}^{\star} = \boldsymbol{M}_{\boldsymbol{\mu}} \left((\bar{\boldsymbol{W}}_{\mathrm{pt}}^{\star})^{-1} - n\boldsymbol{\Sigma}_{\boldsymbol{x}} \right).}$$

2. Determining the fine-tuning loss

Substituting back $\bar{\boldsymbol{p}}^{\star} = (\boldsymbol{W}^{-1} - n\boldsymbol{\Sigma}_{\boldsymbol{x}})\,\boldsymbol{\mu}$:

$$\begin{aligned} &\mathcal{L}(\boldsymbol{W}, \bar{\boldsymbol{p}}) = \mathtt{tr}\left(\mathbb{E}[\boldsymbol{c}\boldsymbol{c}^{\top}] \boldsymbol{\Sigma}_{\boldsymbol{x}} \right) + \sigma^2, \\ &\text{where} \quad \boldsymbol{c} = \left((\boldsymbol{W}\boldsymbol{X}^{\top}\boldsymbol{X} - \boldsymbol{I})(\bar{\boldsymbol{\theta}} + \boldsymbol{\mu}) + \boldsymbol{W}\boldsymbol{X}^{\top}\boldsymbol{\xi} + \boldsymbol{W}\bar{\boldsymbol{p}} \right) \\ &\qquad\quad = \boldsymbol{W}\boldsymbol{X}^{\top}\boldsymbol{X}(\bar{\boldsymbol{\theta}} + \boldsymbol{\mu}) - \bar{\boldsymbol{\theta}} + \boldsymbol{W}\boldsymbol{X}^{\top}\boldsymbol{\xi} - n\boldsymbol{W}\boldsymbol{\Sigma}_{\boldsymbol{x}}\boldsymbol{\mu}. \end{aligned}$$

The expansion of $\boldsymbol{c}\boldsymbol{c}^\top$ is (there are $4 \times 4 = 16$ terms in total):

$$
\begin{aligned}
\boldsymbol{c}\boldsymbol{c}^\top =& \left[\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{X}(\bar{\boldsymbol{\theta}}+\boldsymbol{\mu}) - \bar{\boldsymbol{\theta}} + \boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{\xi} - n\boldsymbol{W}\boldsymbol{\Sigma}_{\boldsymbol{x}}\boldsymbol{\mu}\right]\left[\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{X}(\bar{\boldsymbol{\theta}}+\boldsymbol{\mu}) - \bar{\boldsymbol{\theta}} + \boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{\xi} - n\boldsymbol{W}\boldsymbol{\Sigma}_{\boldsymbol{x}}\boldsymbol{\mu}\right]^\top \\
=& \left[\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{X}(\bar{\boldsymbol{\theta}}+\boldsymbol{\mu})\right]\left[\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{X}(\bar{\boldsymbol{\theta}}+\boldsymbol{\mu})\right]^\top + \left[\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{X}(\bar{\boldsymbol{\theta}}+\boldsymbol{\mu})\right]\left[-\bar{\boldsymbol{\theta}}\right]^\top \\
&+ \left[\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{X}(\bar{\boldsymbol{\theta}}+\boldsymbol{\mu})\right]\left[\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{\xi}\right]^\top + \left[\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{X}(\bar{\boldsymbol{\theta}}+\boldsymbol{\mu})\right]\left[-n\boldsymbol{W}\boldsymbol{\Sigma}_{\boldsymbol{x}}\boldsymbol{\mu}\right]^\top \\
&+ \quad \left[-\bar{\boldsymbol{\theta}}\right]\left[\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{X}(\bar{\boldsymbol{\theta}}+\boldsymbol{\mu})\right]^\top \quad + \quad \left[-\bar{\boldsymbol{\theta}}\right]\left[-\bar{\boldsymbol{\theta}}\right]^\top \quad + \quad \left[-\bar{\boldsymbol{\theta}}\right]\left[\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{\xi}\right]^\top \quad + \quad \left[-\bar{\boldsymbol{\theta}}\right]\left[-n\boldsymbol{W}\boldsymbol{\Sigma}_{\boldsymbol{x}}\boldsymbol{\mu}\right]^\top \\
&+ \left[\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{\xi}\right]\left[\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{X}(\bar{\boldsymbol{\theta}}+\boldsymbol{\mu})\right]^\top + \left[\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{\xi}\right]\left[-\bar{\boldsymbol{\theta}}\right]^\top + \left[\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{\xi}\right]\left[\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{\xi}\right]^\top + \left[\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{\xi}\right]\left[-n\boldsymbol{W}\boldsymbol{\Sigma}_{\boldsymbol{x}}\boldsymbol{\mu}\right]^\top \\
&+ \left[-n\boldsymbol{W}\boldsymbol{\Sigma}_{\boldsymbol{x}}\boldsymbol{\mu}\right]\left[\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{X}(\bar{\boldsymbol{\theta}}+\boldsymbol{\mu})\right]^\top + \left[-n\boldsymbol{W}\boldsymbol{\Sigma}_{\boldsymbol{x}}\boldsymbol{\mu}\right]\left[-\bar{\boldsymbol{\theta}}\right]^\top + \left[-n\boldsymbol{W}\boldsymbol{\Sigma}_{\boldsymbol{x}}\boldsymbol{\mu}\right]\left[\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{\xi}\right]^\top + \left[-n\boldsymbol{W}\boldsymbol{\Sigma}_{\boldsymbol{x}}\boldsymbol{\mu}\right]\left[-n\boldsymbol{W}\boldsymbol{\Sigma}_{\boldsymbol{x}}\boldsymbol{\mu}\right]^\top.
\end{aligned}
$$

Take expectation of it,

$$
\begin{aligned}
\mathbb{E}[\boldsymbol{c}\boldsymbol{c}^\top] =& \left[\boldsymbol{W} \quad \underbrace{\mathbb{E}[\boldsymbol{X}^\top\boldsymbol{X}(\bar{\boldsymbol{\theta}}+\boldsymbol{\mu})(\bar{\boldsymbol{\theta}}+\boldsymbol{\mu})^\top\boldsymbol{X}^\top\boldsymbol{X}]}_{\text{Lemma F.1, denoted as a (task-specific) matrix } \boldsymbol{C}} \quad \boldsymbol{W}^\top\right] + \left[-n\boldsymbol{W}\boldsymbol{\Sigma}_{\boldsymbol{x}}\boldsymbol{\Sigma}_{\boldsymbol{\beta}}\right] \\
&+ \quad 0 \quad + \left[-n^2\boldsymbol{W}\boldsymbol{\Sigma}_{\boldsymbol{x}}\boldsymbol{\mu}\boldsymbol{\mu}^\top\boldsymbol{\Sigma}_{\boldsymbol{x}}\boldsymbol{W}^\top\right] \\
&+ \left[-n\boldsymbol{\Sigma}_{\boldsymbol{\beta}}\boldsymbol{\Sigma}_{\boldsymbol{x}}\boldsymbol{W}^\top\right] \quad + \quad \boldsymbol{\Sigma}_{\boldsymbol{\beta}} \quad + \quad 0 \quad + \quad 0 \\
&+ \quad 0 \quad + \quad 0 \quad + \underbrace{n\sigma^2\boldsymbol{W}\boldsymbol{\Sigma}_{\boldsymbol{x}}\boldsymbol{W}^\top}_{\text{Lemma F.2}} + \quad 0 \\
&+ \left[-n^2\boldsymbol{W}\boldsymbol{\Sigma}_{\boldsymbol{x}}\boldsymbol{\mu}\boldsymbol{\mu}^\top\boldsymbol{\Sigma}_{\boldsymbol{x}}\boldsymbol{W}^\top\right] + \quad 0 \quad + \quad 0 \quad + \quad \left[n^2\boldsymbol{W}\boldsymbol{\Sigma}_{\boldsymbol{x}}\boldsymbol{\mu}\boldsymbol{\mu}^\top\boldsymbol{\Sigma}_{\boldsymbol{x}}\boldsymbol{W}^\top\right] \\
=& \boldsymbol{W}(\boldsymbol{C} + n\sigma^2\boldsymbol{\Sigma}_{\boldsymbol{x}} - n^2\boldsymbol{\Sigma}_{\boldsymbol{x}}\boldsymbol{\mu}\boldsymbol{\mu}^\top\boldsymbol{\Sigma}_{\boldsymbol{x}})\boldsymbol{W}^\top - n\boldsymbol{W}\boldsymbol{\Sigma}_{\boldsymbol{x}}\boldsymbol{\Sigma}_{\boldsymbol{\beta}} - n\boldsymbol{\Sigma}_{\boldsymbol{\beta}}\boldsymbol{\Sigma}_{\boldsymbol{x}}\boldsymbol{W}^\top + \boldsymbol{\Sigma}_{\boldsymbol{\beta}}
\end{aligned}
$$

where $\boldsymbol{C} = n\mathtt{tr}\left(\boldsymbol{\Sigma}_{\boldsymbol{x}}(\boldsymbol{\Sigma}_{\boldsymbol{\beta}} + \boldsymbol{\mu}\boldsymbol{\mu}^\top)\right)\boldsymbol{\Sigma}_{\boldsymbol{x}} + n(n+1)\boldsymbol{\Sigma}_{\boldsymbol{x}}(\boldsymbol{\Sigma}_{\boldsymbol{\beta}} + \boldsymbol{\mu}\boldsymbol{\mu}^\top)\boldsymbol{\Sigma}_{\boldsymbol{x}}$.

Substitute back into the loss $\mathcal{L}(\boldsymbol{W}, \bar{\boldsymbol{p}}^\star(\boldsymbol{W}))$:

$$
\begin{aligned}
&\mathcal{L}(\boldsymbol{W}, \bar{\boldsymbol{p}}^\star(\boldsymbol{W})) \\
&= \mathtt{tr}\left(\mathbb{E}[\boldsymbol{c}\boldsymbol{c}^\top]\boldsymbol{\Sigma}_{\boldsymbol{x}}\right) + \sigma^2 \\
&= \mathtt{tr}\left(\boldsymbol{W}(\boldsymbol{C} + n\sigma^2\boldsymbol{\Sigma}_{\boldsymbol{x}} - n^2\boldsymbol{\Sigma}_{\boldsymbol{x}}\boldsymbol{\mu}\boldsymbol{\mu}^\top\boldsymbol{\Sigma}_{\boldsymbol{x}})\boldsymbol{W}^\top\boldsymbol{\Sigma}_{\boldsymbol{x}}\right) - 2n\mathtt{tr}\left(\boldsymbol{W}\boldsymbol{\Sigma}_{\boldsymbol{x}}\boldsymbol{\Sigma}_{\boldsymbol{\beta}}\boldsymbol{\Sigma}_{\boldsymbol{x}}\right) + \mathtt{tr}\left(\boldsymbol{\Sigma}_{\boldsymbol{\beta}}\boldsymbol{\Sigma}_{\boldsymbol{x}}\right) + \sigma^2
\end{aligned}
$$

Use the following definition:

$$
\begin{aligned}
\text{Debiased: } \bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} &= \boldsymbol{\Sigma}_{\boldsymbol{x}} \sum_{k=1}^K \pi_k \, \mathbb{E}[(\boldsymbol{\beta}_k - \boldsymbol{\mu}_k)(\boldsymbol{\beta}_k - \boldsymbol{\mu}_k)^\top] = \boldsymbol{\Sigma}_{\boldsymbol{x}} \sum_{k=1}^K \pi_k \boldsymbol{\Sigma}_{\boldsymbol{\beta}_k}; \\
\text{Biased: } \quad \tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} &= \boldsymbol{\Sigma}_{\boldsymbol{x}} \sum_{k=1}^K \pi_k \, \mathbb{E}[\boldsymbol{\beta}_k\boldsymbol{\beta}_k^\top] = \boldsymbol{\Sigma}_{\boldsymbol{x}} \sum_{k=1}^K \pi_k (\boldsymbol{\Sigma}_{\boldsymbol{\beta}_k} + \boldsymbol{\mu}_k\boldsymbol{\mu}_k^\top).
\end{aligned}
$$

Denote:

$$\bar{\boldsymbol{C}} = \sum_{k=1}^{K} \pi_k \boldsymbol{C}_k = n\mathtt{tr}\left(\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}\right)\boldsymbol{\Sigma}_{\boldsymbol{x}} + n(n+1)\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}\boldsymbol{\Sigma}_{\boldsymbol{x}}.$$

The weighted sum of the k-th task fine-tuning loss over all tasks $k \in [K]$ is:

$$\mathcal{L}_{\mathrm{ft}}(\boldsymbol{W}, \boldsymbol{P}_{\mathrm{ft}}^{\star}(\boldsymbol{W})) = \mathtt{tr}\left(\boldsymbol{W}\left(\bar{\boldsymbol{C}} + n\sigma^2\boldsymbol{\Sigma}_{\boldsymbol{x}} - n^2(\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} - \bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}})\boldsymbol{\Sigma}_{\boldsymbol{x}}\right)\boldsymbol{W}^{\top}\boldsymbol{\Sigma}_{\boldsymbol{x}}\right) - 2n\mathtt{tr}\left(\boldsymbol{W}\bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}\boldsymbol{\Sigma}_{\boldsymbol{x}}\right) + \mathtt{tr}\left(\bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}\right) + \sigma^2$$

Note that for plain training,

$$\mathcal{L}_{\mathrm{pt}}(\boldsymbol{W}, \boldsymbol{P} = \boldsymbol{0}) = \mathtt{tr}\left(\boldsymbol{W}(\bar{\boldsymbol{C}} + n\sigma^2\boldsymbol{\Sigma}_{\boldsymbol{x}})\boldsymbol{W}^{\top}\boldsymbol{\Sigma}_{\boldsymbol{x}}\right) - 2n\mathtt{tr}\left(\boldsymbol{W}\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}\boldsymbol{\Sigma}_{\boldsymbol{x}}\right) + \mathtt{tr}\left(\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}\right) + \sigma^2,$$

and their optimal loss share the same attention weight parameterization $\boldsymbol{W} = \boldsymbol{W}_{\mathrm{pt}}^{\star}$. Let $\bar{\boldsymbol{W}}_{\mathrm{pt}}^{\star} = \boldsymbol{\Sigma}_{\boldsymbol{x}}\boldsymbol{W}_{\mathrm{pt}}^{\star}$,

$$\begin{aligned}
\mathcal{L}_{\mathrm{ft}}^{\star} &= \mathcal{L}_{\mathrm{pt}}^{\star} - \mathtt{tr}\left(\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} - \bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}\right) + 2n\mathtt{tr}\left(\bar{\boldsymbol{W}}_{\mathrm{pt}}^{\star}(\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} - \bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}})\right) - n^2\mathtt{tr}\left(\bar{\boldsymbol{W}}_{\mathrm{pt}}^{\star}(\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} - \bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}})\bar{\boldsymbol{W}}_{\mathrm{pt}}^{\star\top}\right) \\
&\boxed{= \mathcal{L}_{\mathrm{pt}}^{\star} - \mathtt{tr}\left((\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} - \bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}})(n\bar{\boldsymbol{W}}_{\mathrm{pt}}^{\star} - \boldsymbol{I})^{\top}(n\bar{\boldsymbol{W}}_{\mathrm{pt}}^{\star} - \boldsymbol{I})\right).}
\end{aligned}$$

■

F.2.3 Proof of Theorem 7.3

Theorem F.3 (Joint training) *Consider training a single-layer linear attention model in solving multi-task ICL problem with dataset defined in Definition 7.2 and model construction as described in Assumption 7.1 in the main paper. Let $\boldsymbol{W}_{jt}^{\star}, \boldsymbol{P}_{jt}^{\star}$ (c.f. (7.12) in the main paper) be the optimal solutions and $\mathcal{L}_{jt}^{\star}$ is the optimal joint training loss defined in Section 7.2.3. Additionally, let $\bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}, \tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}, \boldsymbol{M}_{\boldsymbol{\mu}}$ follow the same definitions as in Theorem 7.2 and define $\bar{\boldsymbol{W}}_{jt}^{\star} = \boldsymbol{\Sigma}_{\boldsymbol{x}}\boldsymbol{W}_{jt}^{\star}$. Then the solution $(\boldsymbol{W}_{jt}^{\star}, \boldsymbol{P}_{jt}^{\star})$ and optimal loss $\mathcal{L}_{jt}^{\star}$ satisfy*

$$\begin{aligned}
\bar{\boldsymbol{W}}_{jt}^{\star} &= \bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}\left((n+1)\bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} + (tr\left(\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}\right) + \sigma^2)\boldsymbol{I} + \boldsymbol{\Sigma}_{\boldsymbol{x}}\sum_{k=1}^{K}\pi_k\boldsymbol{\mu}_k\boldsymbol{\mu}_k^{\top}\right)^{-1}, \\
\boldsymbol{P}_{jt}^{\star} &= \boldsymbol{M}_{\boldsymbol{\mu}}\left((\bar{\boldsymbol{W}}_{jt}^{\star})^{-1} - n\boldsymbol{I}\right)\boldsymbol{\Sigma}_{\boldsymbol{x}}, \\
\mathcal{L}_{jt}^{\star} &= tr\left(\bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}\right) + \sigma^2 - n\,tr\left(\bar{\boldsymbol{W}}_{jt}^{\star}\bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}\right).
\end{aligned}$$

Here, $\boldsymbol{\Sigma}_{\boldsymbol{x}}\sum_{k=1}^{K}\pi_k\boldsymbol{\mu}_k\boldsymbol{\mu}_k^{\top} \sim \mathcal{O}(1)$ is a $d \times d$-sized constant matrix.

Proof. As previously stated, the following derivation applies to all tasks $k \in [K]$. Therefore,

for simplicity, we omit the index k in the notation unless otherwise specified.

1. Determining the optimal task-specific prompts

In the joint training setting, the attention model is pretrained and parameterized by a trainable $\boldsymbol{W}$, and the task-specific prompts are fine-tuned accordingly (see (F.5)):

$$\begin{aligned}&\mathcal{L}(\boldsymbol{W}, \bar{\boldsymbol{p}}) = \mathtt{tr}\left(\mathbb{E}[\boldsymbol{c}\boldsymbol{c}^\top]\boldsymbol{\Sigma}_{\boldsymbol{x}}\right) + \sigma^2,\\ &\text{where} \quad \boldsymbol{c} = (\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{X} - \boldsymbol{I})(\bar{\boldsymbol{\theta}} + \boldsymbol{\mu}) + \boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{\xi} + \boldsymbol{W}\bar{\boldsymbol{p}}.\end{aligned}$$

The optimal task-specific prompt is determined by taking the derivative and setting it to zero:

$$\begin{aligned}\frac{\partial \mathcal{L}(\boldsymbol{W}, \bar{\boldsymbol{p}})}{\partial \bar{\boldsymbol{p}}} &= 2\,\mathbb{E}\left[\frac{\partial \boldsymbol{c}}{\partial \bar{\boldsymbol{p}}}^\top \boldsymbol{\Sigma}_{\boldsymbol{x}}\boldsymbol{c}\right]\\ &= 2\boldsymbol{W}^\top\boldsymbol{\Sigma}_{\boldsymbol{x}}\,\mathbb{E}[\boldsymbol{c}] = 2\boldsymbol{W}^\top\boldsymbol{\Sigma}_{\boldsymbol{x}}\left[(n\boldsymbol{W}\boldsymbol{\Sigma}_{\boldsymbol{x}} - \boldsymbol{I})\boldsymbol{\mu} + \boldsymbol{W}\bar{\boldsymbol{p}}\right] = \boldsymbol{0},\\ \Rightarrow \quad \bar{\boldsymbol{p}}^\star &= \left(\boldsymbol{W}^{-1} - n\boldsymbol{\Sigma}_{\boldsymbol{x}}\right)\boldsymbol{\mu},\end{aligned}$$

which is equivalent to:

$$\boxed{\boldsymbol{P}^\star_{\mathrm{jt}} = \boldsymbol{M}_{\boldsymbol{\mu}}\left((\bar{\boldsymbol{W}}^\star_{\mathrm{jt}})^{-1} - n\boldsymbol{I}\right)\boldsymbol{\Sigma}_{\boldsymbol{x}}.}$$

2. Determining the fine-tuning loss

It is worth noting that fine-tuning and joint training share a functional relationship between the tuned prompts $\bar{\boldsymbol{p}}^\star$ and the current attention model parameterization $\boldsymbol{W}$. Thus, substituting the tuned prompt back into the joint training loss will result in the same expression as the fine-tuning loss:

$$\begin{aligned}&\mathcal{L}(\boldsymbol{W}, \bar{\boldsymbol{p}}^\star(\boldsymbol{W}))\\ &= \mathtt{tr}\left(\mathbb{E}[\boldsymbol{c}\boldsymbol{c}^\top]\boldsymbol{\Sigma}_{\boldsymbol{x}}\right) + \sigma^2\\ &= \mathtt{tr}\left(\boldsymbol{W}(\boldsymbol{C} + n\sigma^2\boldsymbol{\Sigma}_{\boldsymbol{x}} - n^2\boldsymbol{\Sigma}_{\boldsymbol{x}}\boldsymbol{\mu}\boldsymbol{\mu}^\top\boldsymbol{\Sigma}_{\boldsymbol{x}})\boldsymbol{W}^\top\boldsymbol{\Sigma}_{\boldsymbol{x}}\right) - 2n\mathtt{tr}\left(\boldsymbol{W}\boldsymbol{\Sigma}_{\boldsymbol{x}}\boldsymbol{\Sigma}_{\boldsymbol{\beta}}\boldsymbol{\Sigma}_{\boldsymbol{x}}\right) + \mathtt{tr}\left(\boldsymbol{\Sigma}_{\boldsymbol{\beta}}\boldsymbol{\Sigma}_{\boldsymbol{x}}\right)\end{aligned}$$

Use the following definition:

$$\begin{aligned}&\text{Debiased: } \bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} = \boldsymbol{\Sigma}_{\boldsymbol{x}}\sum_{k=1}^K \pi_k\,\mathbb{E}[(\boldsymbol{\beta}_k - \boldsymbol{\mu}_k)(\boldsymbol{\beta}_k - \boldsymbol{\mu}_k)^\top] = \boldsymbol{\Sigma}_{\boldsymbol{x}}\sum_{k=1}^K \pi_k\boldsymbol{\Sigma}_{\boldsymbol{\beta}_k};\\ &\text{Biased:} \quad \tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} = \boldsymbol{\Sigma}_{\boldsymbol{x}}\sum_{k=1}^K \pi_k\,\mathbb{E}[\boldsymbol{\beta}_k\boldsymbol{\beta}_k^\top] = \boldsymbol{\Sigma}_{\boldsymbol{x}}\sum_{k=1}^K \pi_k(\boldsymbol{\Sigma}_{\boldsymbol{\beta}_k} + \boldsymbol{\mu}_k\boldsymbol{\mu}_k^\top).\end{aligned}$$

Denote:

$$\bar{\boldsymbol{C}} = \sum_{k=1}^K \pi_k\boldsymbol{C}_k = n\mathtt{tr}\left(\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}\right)\boldsymbol{\Sigma}_{\boldsymbol{x}} + n(n+1)\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}\boldsymbol{\Sigma}_{\boldsymbol{x}}.$$

The weighted sum of the k-th task fine-tuning loss over all tasks $k \in [K]$ is:

$$\mathcal{L}_{\text{jt}}(\boldsymbol{W}, \boldsymbol{P}^\star_{\text{jt}}(\boldsymbol{W})) = \texttt{tr}\left(\boldsymbol{W}\left(\bar{\boldsymbol{C}} + n\sigma^2\boldsymbol{\Sigma}_{\boldsymbol{x}} - n^2(\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} - \bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}})\boldsymbol{\Sigma}_{\boldsymbol{x}}\right)\boldsymbol{W}^\top\boldsymbol{\Sigma}_{\boldsymbol{x}}\right) - 2n\texttt{tr}\left(\boldsymbol{W}\bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}\boldsymbol{\Sigma}_{\boldsymbol{x}}\right) + \texttt{tr}\left(\bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}\right) + \sigma^2.$$

In joint training, the attention model parameterization $\boldsymbol{W}$ is no longer fixed (as it is in the fine-tuning setting), but is instead optimized. Due to the functional relationship between the tuned prompts and parameterization $\bar{\boldsymbol{p}}^\star = \bar{\boldsymbol{p}}^\star(\boldsymbol{W})$, they will be optimized jointly until reaching their optimal values.

Take derivative w.r.t. $\boldsymbol{W}$, using Lemma F.3:

$$\frac{\partial \mathcal{L}_{\text{jt}}(\boldsymbol{W}, \boldsymbol{P}^\star_{\text{jt}}(\boldsymbol{W}))}{\partial \boldsymbol{W}} = 2\boldsymbol{\Sigma}_{\boldsymbol{x}}\boldsymbol{W}\left(\bar{\boldsymbol{C}} + n\sigma^2\boldsymbol{\Sigma}_{\boldsymbol{x}} - n^2(\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} - \bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}})\boldsymbol{\Sigma}_{\boldsymbol{x}}\right) - 2n\bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}\boldsymbol{\Sigma}_{\boldsymbol{x}}$$

Let $\bar{\boldsymbol{W}}^\star_{\text{jt}} = \boldsymbol{\Sigma}_{\boldsymbol{x}}\boldsymbol{W}^\star_{\text{jt}}$, and set the derivative to zero (note that $\sum_{k=1}^K \boldsymbol{\mu}_k\boldsymbol{\mu}_k^\top \sim \mathcal{O}(1)$):

$$\begin{aligned}
\bar{\boldsymbol{W}}^\star_{\text{jt}} &= n\bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}\boldsymbol{\Sigma}_{\boldsymbol{x}}\left(\bar{\boldsymbol{C}} + n\sigma^2\boldsymbol{\Sigma}_{\boldsymbol{x}} - n^2(\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} - \bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}})\boldsymbol{\Sigma}_{\boldsymbol{x}})\right)^{-1} \\
&= \bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}\boldsymbol{\Sigma}_{\boldsymbol{x}}\left((\texttt{tr}\left(\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}\right) + \sigma^2)\boldsymbol{\Sigma}_{\boldsymbol{x}} - n(\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} - \bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}})\boldsymbol{\Sigma}_{\boldsymbol{x}} + (n+1)\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}\boldsymbol{\Sigma}_{\boldsymbol{x}}\right)^{-1} \\
&= \bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}\left((\texttt{tr}\left(\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}\right) + \sigma^2)\boldsymbol{I} + (n+1)\bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} + (\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} - \bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}})\right)^{-1} \\
&\boxed{= \bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}\left((n+1)\bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} + (\texttt{tr}\left(\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}\right) + \sigma^2)\boldsymbol{I} + \boldsymbol{\Sigma}_{\boldsymbol{x}}\sum_{k=1}^K \pi_k\boldsymbol{\mu}_k\boldsymbol{\mu}_k^\top\right)^{-1}.}
\end{aligned}$$

Substitute back into $\mathcal{L}_{\text{jt}}(\boldsymbol{W}, \boldsymbol{P}^\star(\boldsymbol{W}))$:

$$\boxed{\mathcal{L}^\star_{\text{jt}} = \texttt{tr}\left(\bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}\right) + \sigma^2 - n\texttt{tr}\left(\bar{\boldsymbol{W}}^\star_{\text{jt}}\bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}\right).}$$

■

F.2.4 Proof of Corollary 7.1

Corollary F.1 *Let $\mathcal{L}^*_{pt}$, $\mathcal{L}^*_{ft}$, and $\mathcal{L}^*_{jt}$ denote the optimal losses for plain training, fine-tuning, and joint training, as described in Theorems 1, 2 and 3, respectively. These losses satisfy:*

$$\mathcal{L}^*_{jt} \leq \mathcal{L}^*_{ft} \leq \mathcal{L}^*_{pt}. \tag{F.6}$$

The equalities hold if and only if $\bar{\boldsymbol{\Sigma}}_\beta = \tilde{\boldsymbol{\Sigma}}_\beta$ (c.f. (14)), which occurs when all task means $\boldsymbol{\mu}_k = \boldsymbol{0}$ for $k \in [K]$. Furthermore, the loss gaps satisfy the following:

1. *The loss gaps scale quadratically with task mean:* $\mathcal{L}^*_{pt} - \mathcal{L}^*_{ft} \sim \mathcal{O}\left(\frac{1}{n^2}\right) \|\Delta\|_F$, $\mathcal{L}^*_{ft} - \mathcal{L}^*_{jt} \sim \mathcal{O}\left(\frac{1}{n}\right) \|\Delta\|_F$, *where* $\Delta := \tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} - \bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} = \boldsymbol{\Sigma}_x \sum_{k=1}^K \pi_k \boldsymbol{\mu}_k \boldsymbol{\mu}_k^\top$.

2. *The ratio between gaps is:* $\frac{\mathcal{L}^*_{pt} - \mathcal{L}^*_{ft}}{\mathcal{L}^*_{ft} - \mathcal{L}^*_{jt}} \sim \mathcal{O}\left(\frac{1}{n}\right)$, *indicating that fine-tuning provides most of the benefit in few-shot regimes (small* n*), while joint training benefits more for larger* n*.*

Proof. **1.** $\mathcal{L}^\star_{\text{ft}} \leq \mathcal{L}^\star_{\text{pt}}$**:**

From Theorem 7.2, we have:

$$\mathcal{L}^\star_{\text{ft}} = \mathcal{L}^\star_{\text{pt}} - \texttt{tr}\left((\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} - \bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}})(n\bar{\boldsymbol{W}}^\star_{\text{pt}} - \boldsymbol{I})^\top (n\bar{\boldsymbol{W}}^\star_{\text{pt}} - \boldsymbol{I})\right).$$

Use the following definition:

$$\text{Debiased: } \bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} = \boldsymbol{\Sigma}_{\boldsymbol{x}} \sum_{k=1}^K \pi_k \, \mathbb{E}[(\boldsymbol{\beta}_k - \boldsymbol{\mu}_k)(\boldsymbol{\beta}_k - \boldsymbol{\mu}_k)^\top] = \boldsymbol{\Sigma}_{\boldsymbol{x}} \sum_{k=1}^K \pi_k \boldsymbol{\Sigma}_{\boldsymbol{\beta}_k};$$

$$\text{Biased: } \quad \tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} = \boldsymbol{\Sigma}_{\boldsymbol{x}} \sum_{k=1}^K \pi_k \, \mathbb{E}[\boldsymbol{\beta}_k \boldsymbol{\beta}_k^\top] = \boldsymbol{\Sigma}_{\boldsymbol{x}} \sum_{k=1}^K \pi_k (\boldsymbol{\Sigma}_{\boldsymbol{\beta}_k} + \boldsymbol{\mu}_k \boldsymbol{\mu}_k^\top).$$

We define an auxiliray variable:

$$\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} - \bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} = \boldsymbol{\Sigma}_x \sum_{k=1}^K \pi_k \boldsymbol{\mu}_k \boldsymbol{\mu}_k^\top = \Delta$$

It can be seen that $(\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} - \bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}) = \boldsymbol{\Sigma}_{\boldsymbol{x}} (\sum_{k=1}^K \pi_k \boldsymbol{\mu}_k \boldsymbol{\mu}_k^\top) \succeq \boldsymbol{0}, (n\bar{\boldsymbol{W}}^\star_{\text{pt}} - \boldsymbol{I})^\top (n\bar{\boldsymbol{W}}^\star_{\text{pt}} - \boldsymbol{I}) \succ \boldsymbol{0}$, which leads to

$$\texttt{tr}\left((\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} - \bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}})(n\bar{\boldsymbol{W}}^\star_{\text{pt}} - \boldsymbol{I})^\top (n\bar{\boldsymbol{W}}^\star_{\text{pt}} - \boldsymbol{I})\right) \geq 0 \Rightarrow \mathcal{L}^\star_{\text{ft}} \leq \mathcal{L}^\star_{\text{pt}}.$$

The equality holds if and only if $\boxed{\boldsymbol{\mu}_k = \boldsymbol{0}, k \in [K] \iff \bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} = \tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}.}$

The loss gap between $\mathcal{L}^\star_{\text{ft}}$ and $\mathcal{L}^\star_{\text{pt}}$ can be written as:

$$\mathcal{L}^\star_{\text{pt}} - \mathcal{L}^\star_{\text{ft}} = \texttt{tr}\left(\Delta (n\bar{\boldsymbol{W}}^\star_{\text{pt}} - \boldsymbol{I})^\top (n\bar{\boldsymbol{W}}^\star_{\text{pt}} - \boldsymbol{I})\right)$$

To analyze the asymptotic behavior of the gap, we need to analyze the asymptotic behavior of:

$$\bar{\boldsymbol{W}}^\star_{\text{pt}} = \tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} ((n+1)\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} + \texttt{tr}\left(\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}\right) \boldsymbol{I})^{-1}.$$

First, for large n, we can factor out n from the inverse term:

$$\bar{\boldsymbol{W}}_{\text{pt}}^{\star} = \frac{1}{n}\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} \left(\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} \left(1 + \frac{1}{n} \right) + \frac{\texttt{tr}\left(\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} \right)}{n} \boldsymbol{I} \right)^{-1}$$

Using a matrix Taylor expansion, with $\boldsymbol{A} = \tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}$ and $\boldsymbol{B} = \frac{\texttt{tr}\left(\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} \right)}{n} \boldsymbol{I} + \frac{1}{n}\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}$:

$$(\boldsymbol{A} + \boldsymbol{B})^{-1} = \boldsymbol{A}^{-1} - \boldsymbol{A}^{-1}\boldsymbol{B}\boldsymbol{A}^{-1} + \mathcal{O}(\|\boldsymbol{B}\|^2)$$

Applying this to our expression:

$$\left(\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} \left(1 + \frac{1}{n} \right) + \frac{\texttt{tr}\left(\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} \right)}{n} \boldsymbol{I} \right)^{-1} = \tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}^{-1} - \frac{1}{n}\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}^{-1} \left(\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} + \texttt{tr}\left(\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} \right) \boldsymbol{I} \right) \tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}^{-1} + \mathcal{O}\left(\frac{1}{n^2} \right)$$

Therefore:

$$\bar{\boldsymbol{W}}_{\text{pt}}^{\star} = \frac{1}{n}\boldsymbol{I} - \frac{1}{n^2} \left(\boldsymbol{I} + \texttt{tr}\left(\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} \right) \tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}^{-1} \right) + \mathcal{O}\left(\frac{1}{n^3} \right)$$

The omitted $\mathcal{O}(\frac{1}{n^3})$ terms include higher-order expansion terms from the matrix Taylor series. These terms involve powers of $\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}$ and its inverse. They grow increasingly small as n increases and don't affect the dominant $\mathcal{O}(\frac{1}{n^2})$ behavior of the loss gap.

Using the result above:

$$n\bar{\boldsymbol{W}}_{\text{pt}}^{\star} - \boldsymbol{I} = n \left(\frac{1}{n}\boldsymbol{I} - \frac{1}{n^2} \left(\boldsymbol{I} + \texttt{tr}\left(\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} \right) \tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}^{-1} \right) + \mathcal{O}\left(\frac{1}{n^3} \right) \right) - \boldsymbol{I}$$

$$n\bar{\boldsymbol{W}}_{\text{pt}}^{\star} - \boldsymbol{I} = \boldsymbol{I} - \frac{1}{n} \left(\boldsymbol{I} + \texttt{tr}\left(\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} \right) \tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}^{-1} \right) + \mathcal{O}\left(\frac{1}{n^2} \right) - \boldsymbol{I}$$

$$n\bar{\boldsymbol{W}}_{\text{pt}}^{\star} - \boldsymbol{I} = -\frac{1}{n} \left(\boldsymbol{I} + \texttt{tr}\left(\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} \right) \tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}^{-1} \right) + \mathcal{O}\left(\frac{1}{n^2} \right)$$

The omitted terms at order $\mathcal{O}(\frac{1}{n^2})$ include additional matrix products that become negligible for large n.

Therefore:

$$\|n\bar{\boldsymbol{W}}_{\text{pt}}^{\star} - \boldsymbol{I}\|_F \sim \mathcal{O}\left(\frac{1}{n} \right)$$

Now, substituting this into the expression for the gap:

$$\mathcal{L}_{\text{pt}}^{\star} - \mathcal{L}_{\text{ft}}^{\star} = \texttt{tr}\left(\Delta (n\bar{\boldsymbol{W}}_{\text{pt}}^{\star} - \boldsymbol{I})^{\top} (n\bar{\boldsymbol{W}}_{\text{pt}}^{\star} - \boldsymbol{I}) \right)$$

$$\mathcal{L}^\star_{\text{pt}} - \mathcal{L}^\star_{\text{ft}} \sim \texttt{tr}\left(\Delta \cdot \mathcal{O}\left(\frac{1}{n^2}\right)\right)$$

$$\mathcal{L}^\star_{\text{pt}} - \mathcal{L}^\star_{\text{ft}} \sim \mathcal{O}\left(\frac{1}{n^2}\right)\|\Delta\|_F$$

Given that $\|\Delta\|_F \leq M^2 \sum_{k=1}^K \pi_k^2 \cdot \lambda_{\max}(\boldsymbol{\Sigma}_x)$, we have:

$$\boxed{\mathcal{L}^\star_{\text{pt}} - \mathcal{L}^\star_{\text{ft}} \sim \mathcal{O}\left(\frac{1}{n^2}\right) M^2 \sum_{k=1}^K \pi_k^2 \cdot \lambda_{\max}(\boldsymbol{\Sigma}_x)}$$

2. $\mathcal{L}^\star_{\text{jt}} \leq \mathcal{L}^\star_{\text{ft}}$:

From the proof of Theorem 7.3, it can be seen that joint training and fine-tuning shares a functional relationship between $\mathcal{L}$ and $\boldsymbol{W}$:

$$\mathcal{L}_{\text{jt}}(\boldsymbol{W}, \boldsymbol{P}^\star_{\text{jt}}(\boldsymbol{W})) = \mathcal{L}_{\text{ft}}(\boldsymbol{W}, \boldsymbol{P}^\star_{\text{ft}}(\boldsymbol{W}))$$

However, the only minimizer of this funtion is derived from $\frac{\partial \mathcal{L}_{\text{jt}}(\boldsymbol{W}, \boldsymbol{P}^\star(\boldsymbol{W}))}{\partial \boldsymbol{W}} = 0 \Rightarrow \boldsymbol{W}^\star_{\text{jt}}$, which leads to:

$$\mathcal{L}^\star_{\text{jt}} \leq \mathcal{L}^\star_{\text{ft}}.$$

The equality holds if and only if $\boxed{\boldsymbol{W}^\star_{\text{pt}} = \boldsymbol{W}^\star_{\text{jt}} \iff \boldsymbol{\mu}_k = \boldsymbol{0}, k \in [K] \iff \bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} = \tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}.}$

Recall that $\bar{\boldsymbol{W}}^\star_{\text{jt}}$ is given by:

$$\bar{\boldsymbol{W}}^\star_{\text{jt}} = \bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}((n+1)\bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} + \texttt{tr}\left(\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}\right)\boldsymbol{I} + \boldsymbol{\Sigma}_x \sum_{k=1}^K \pi_k \boldsymbol{\mu}_k \boldsymbol{\mu}_k^\top)^{-1}$$

Note that we denote $\Delta = \boldsymbol{\Sigma}_x \sum_{k=1}^K \pi_k \boldsymbol{\mu}_k \boldsymbol{\mu}_k^\top$, so:

$$\begin{aligned}\bar{\boldsymbol{W}}^\star_{\text{jt}} &= \bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}((n+1)\bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} + \texttt{tr}\left(\bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} + \Delta\right)\boldsymbol{I} + \Delta)^{-1} \\ &= \bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}((n+1)\bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} + \texttt{tr}\left(\bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}\right)\boldsymbol{I} + \texttt{tr}\left(\Delta\right)\boldsymbol{I} + \Delta)^{-1}\end{aligned}$$

First, we rewrite this for large n:

$$\bar{\boldsymbol{W}}^\star_{\text{jt}} = \frac{1}{n}\bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}\left(\bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}\left(1 + \frac{1}{n}\right) + \frac{\texttt{tr}\left(\bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}\right)}{n}\boldsymbol{I} + \frac{\texttt{tr}\left(\Delta\right)}{n}\boldsymbol{I} + \frac{\Delta}{n}\right)^{-1}$$

Let $\boldsymbol{A} = \bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}$ and $\boldsymbol{B} = \frac{1}{n}\bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} + \frac{\texttt{tr}\left(\bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}\right)}{n}\boldsymbol{I} + \frac{\texttt{tr}(\Delta)}{n}\boldsymbol{I} + \frac{\Delta}{n}$.

Using the matrix Taylor expansion for $(\boldsymbol{A}+\boldsymbol{B})^{-1}$:

$$(\boldsymbol{A}+\boldsymbol{B})^{-1} = \boldsymbol{A}^{-1} - \boldsymbol{A}^{-1}\boldsymbol{B}\boldsymbol{A}^{-1} + \boldsymbol{A}^{-1}\boldsymbol{B}\boldsymbol{A}^{-1}\boldsymbol{B}\boldsymbol{A}^{-1} + \mathcal{O}(\|\boldsymbol{B}\|^3)$$

Applying this to our expression and noting that $\|\boldsymbol{B}\| = \mathcal{O}(1/n)$:

$$\begin{aligned}
&\left(\bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}\left(1+\frac{1}{n}\right) + \frac{\mathtt{tr}\left(\bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}\right)}{n}\boldsymbol{I} + \frac{\mathtt{tr}\left(\Delta\right)}{n}\boldsymbol{I} + \frac{\Delta}{n}\right)^{-1} \\
&= \bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}^{-1} - \frac{1}{n}\bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}^{-1}\left(\bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} + \mathtt{tr}\left(\bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}\right)\boldsymbol{I} + \mathtt{tr}\left(\Delta\right)\boldsymbol{I} + \Delta\right)\bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}^{-1} + \mathcal{O}\left(\frac{1}{n^2}\right)
\end{aligned}$$

Therefore:

$$\bar{\boldsymbol{W}}_{\mathrm{jt}}^{\star} = \frac{1}{n}\boldsymbol{I} - \frac{1}{n^2}\left(\boldsymbol{I} + \mathtt{tr}\left(\bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}\right)\bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}^{-1} + \mathtt{tr}\left(\Delta\right)\bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}^{-1} + \bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}^{-1}\Delta\right) + \mathcal{O}\left(\frac{1}{n^3}\right)$$

The omitted $\mathcal{O}(\frac{1}{n^3})$ terms include higher-order matrix products involving powers of $\bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}$, Δ, and their inverses. These become negligible as n grows.

Using the result above:

$$\begin{aligned}
n\bar{\boldsymbol{W}}_{\mathrm{jt}}^{\star} - \boldsymbol{I} &= n\left(\frac{1}{n}\boldsymbol{I} - \frac{1}{n^2}\left(\boldsymbol{I} + \mathtt{tr}\left(\bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}\right)\bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}^{-1} + \mathtt{tr}\left(\Delta\right)\bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}^{-1} + \bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}^{-1}\Delta\right) + \mathcal{O}\left(\frac{1}{n^3}\right)\right) - \boldsymbol{I} \\
&= -\frac{1}{n}\left(\boldsymbol{I} + \mathtt{tr}\left(\bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}\right)\bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}^{-1} + \mathtt{tr}\left(\Delta\right)\bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}^{-1} + \bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}^{-1}\Delta\right) + \mathcal{O}\left(\frac{1}{n^2}\right)
\end{aligned}$$

Therefore:

$$\|n\bar{\boldsymbol{W}}_{\mathrm{jt}}^{\star} - \boldsymbol{I}\|_F = \mathcal{O}\left(\frac{1}{n}\right)$$

Recall that:

$$\bar{\boldsymbol{W}}_{\mathrm{pt}}^{\star} = \frac{1}{n}\boldsymbol{I} - \frac{1}{n^2}\left(\boldsymbol{I} + \mathtt{tr}\left(\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}\right)\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}^{-1}\right) + \mathcal{O}\left(\frac{1}{n^3}\right)$$

$$\bar{\boldsymbol{W}}_{\mathrm{jt}}^{\star} = \frac{1}{n}\boldsymbol{I} - \frac{1}{n^2}\left(\boldsymbol{I} + \mathtt{tr}\left(\bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}\right)\bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}^{-1} + \mathtt{tr}\left(\Delta\right)\bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}^{-1} + \bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}^{-1}\Delta\right) + \mathcal{O}\left(\frac{1}{n^3}\right)$$

The leading terms $(1/n)\boldsymbol{I}$ cancel, and the difference appears in the second-order terms:

$$\bar{\boldsymbol{W}}_{\mathrm{pt}}^{\star} - \bar{\boldsymbol{W}}_{\mathrm{jt}}^{\star} = \frac{1}{n^2}\boldsymbol{C} + \mathcal{O}\left(\frac{1}{n^3}\right)$$

Where $\boldsymbol{C}$ is a matrix that depends on $\bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}$ and Δ. Therefore:

$$\|\bar{\boldsymbol{W}}_{\mathrm{pt}}^{\star} - \bar{\boldsymbol{W}}_{\mathrm{jt}}^{\star}\|_F \sim \mathcal{O}\left(\frac{1}{n^2}\right) \|\Delta\|_F$$

Starting from the loss function definition:

$$\mathcal{L}(\boldsymbol{W}) = \mathtt{tr}\left(\bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}\right) - n\mathtt{tr}\left(\boldsymbol{W}\bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}\right)$$

The loss gap becomes:

$$\mathcal{L}_{\mathrm{ft}}^{\star} - \mathcal{L}_{\mathrm{jt}}^{\star} = n\|\mathtt{tr}\left((\bar{\boldsymbol{W}}_{\mathrm{pt}}^{\star} - \bar{\boldsymbol{W}}_{\mathrm{jt}}^{\star})\bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}\right)\|$$

This gives us:

$$\boxed{\mathcal{L}_{\mathrm{ft}}^{\star} - \mathcal{L}_{\mathrm{jt}}^{\star} \sim \mathcal{O}\left(\frac{1}{n}\right) M^2 \sum_{k=1}^{K} \pi_k^2 \cdot \lambda_{\max}(\boldsymbol{\Sigma}_x)}$$

3. Comparative Analysis

1. Plain Training vs Fine-tuning:

$$\mathcal{L}_{\mathrm{pt}}^{\star} - \mathcal{L}_{\mathrm{ft}}^{\star} \sim \mathcal{O}\left(\frac{1}{n^2}\right) M^2 \sum_{k=1}^{K} \pi_k^2 \cdot \lambda_{\max}(\boldsymbol{\Sigma}_x)$$

2. Fine-tuning vs Joint Training:

$$\mathcal{L}_{\mathrm{ft}}^{\star} - \mathcal{L}_{\mathrm{jt}}^{\star} \sim \mathcal{O}\left(\frac{1}{n}\right) M^2 \sum_{k=1}^{K} \pi_k^2 \cdot \lambda_{\max}(\boldsymbol{\Sigma}_x)$$

3. Ratio of gaps:

$$\frac{\mathcal{L}_{\mathrm{pt}}^{\star} - \mathcal{L}_{\mathrm{ft}}^{\star}}{\mathcal{L}_{\mathrm{ft}}^{\star} - \mathcal{L}_{\mathrm{jt}}^{\star}} \sim \mathcal{O}\left(\frac{1}{n}\right)$$

This aligns with our experimental results in Section 6, indicating that for few-shot multi-task in-context learning settings (e.g., when $n = 1$), fine-tuning task-specific prompts alone is sufficiently effective at improving performance. However, for many-shot (large n) settings, joint training of both the attention weight and task-specific prompts is necessary to achieve further performance improvements. ■

F.3 Proofs for Section 7.4

F.3.1 Proof of Proposition 7.1

Proposition F.1 *Consider the multi-task ICL data as described in Definition 7.2 and let* $\tilde{\mathcal{L}}^{\star}_{att}$ *and* $\tilde{\mathcal{L}}^{\star}_{\mathrm{pgd}}$ *be the optimal linear attention and debiased preconditioned gradient descent losses as presented in* (7.18) *and* (7.19) *in the main paper, respectively. Then,* $\tilde{\mathcal{L}}^{\star}_{att} = \tilde{\mathcal{L}}^{\star}_{\mathrm{pgd}}$.

Proof. To begin with, let attention weights be

$$\boldsymbol{W}_q\boldsymbol{W}_k^\top = \begin{bmatrix} \bar{\boldsymbol{W}}_1 & \boldsymbol{w}_1 \\ * & * \end{bmatrix} \quad \text{and} \quad \boldsymbol{W}_v = \begin{bmatrix} \bar{\boldsymbol{W}}_2 & \boldsymbol{w}_2 \\ \boldsymbol{w}_3^\top & w \end{bmatrix},$$

where $\bar{\boldsymbol{W}}_{1,2} \in \mathbb{R}^{d\times d}, \boldsymbol{w}_{1,2,3} \in \mathbb{R}^d$ and $w \in \mathbb{R}$. Additionally, let task-specific prompts and heads be

$$\boldsymbol{p}_k = \begin{bmatrix} \bar{\boldsymbol{p}}_k \\ p_k \end{bmatrix} \quad \text{and} \quad \boldsymbol{h}_k = \begin{bmatrix} \bar{\boldsymbol{h}}_k \\ h_k \end{bmatrix} \quad \text{for } k \in [K],$$

where $\bar{\boldsymbol{p}}_k, \bar{\boldsymbol{h}}_k \in \mathbb{R}^d$ and $p_k, h_k \in \mathbb{R}$. Recapping the prediction from (7.17) in the main paper and input sequence $\boldsymbol{Z}^{(k)}$ from (7.5) in the main paper, we obtain

$$\begin{aligned} \tilde{f}_{\mathrm{att}}(\boldsymbol{Z}^{(k)}) &= (\boldsymbol{z}^\top \boldsymbol{W}_q\boldsymbol{W}_k^\top(\boldsymbol{Z}^{(k)})^\top)\boldsymbol{M}\boldsymbol{Z}^{(k)}\boldsymbol{W}_v\boldsymbol{h}_k \\ &= \begin{bmatrix} \boldsymbol{x}^\top\bar{\boldsymbol{W}}_1 & \boldsymbol{x}^\top\boldsymbol{w}_1 \end{bmatrix} \left(\begin{bmatrix} \bar{\boldsymbol{p}}_k\bar{\boldsymbol{p}}_k^\top & p_k\bar{\boldsymbol{p}}_k \\ p_k\bar{\boldsymbol{p}}_k^\top & p_k^2 \end{bmatrix} + \begin{bmatrix} \boldsymbol{X}^\top\boldsymbol{X} & \boldsymbol{X}^\top\boldsymbol{y} \\ \boldsymbol{y}^\top\boldsymbol{X} & \|\boldsymbol{y}\|^2 \end{bmatrix} \right) \begin{bmatrix} \bar{\boldsymbol{W}}_2\bar{\boldsymbol{h}}_k + h_k\boldsymbol{w}_2 \\ \boldsymbol{w}_3^\top\bar{\boldsymbol{h}}_k + h_k w \end{bmatrix}. \end{aligned}$$

Recap the task distribution from Definition 7.2. Let $\boldsymbol{y}_0 = \boldsymbol{y} - \boldsymbol{X}\boldsymbol{\mu}_k$. For cleaner notation and without loss of generality, we remove the subscription k and set

$$\begin{bmatrix} \bar{\boldsymbol{W}}_2\bar{\boldsymbol{h}}_k + h_k\boldsymbol{w}_2 \\ \boldsymbol{w}_3^\top\bar{\boldsymbol{h}}_k + h_k w \end{bmatrix} = \begin{bmatrix} \boldsymbol{v} \\ v \end{bmatrix}$$

where $\boldsymbol{v} \in \mathbb{R}^d$ and $v \in \mathbb{R}$.

$$\begin{aligned}
\tilde{f}_{\mathrm{att}}(\boldsymbol{Z}^{(k)}) &= \begin{bmatrix} \boldsymbol{x}^\top \bar{\boldsymbol{W}}_1 & \boldsymbol{x}^\top \boldsymbol{w}_1 \end{bmatrix} \left(\begin{bmatrix} \bar{\boldsymbol{p}}\bar{\boldsymbol{p}}^\top & p\bar{\boldsymbol{p}} \\ p\bar{\boldsymbol{p}}^\top & p^2 \end{bmatrix} + \begin{bmatrix} \boldsymbol{X}^\top \boldsymbol{X} & \boldsymbol{X}^\top \boldsymbol{y} \\ \boldsymbol{y}^\top \boldsymbol{X} & \|\boldsymbol{y}\|^2 \end{bmatrix} \right) \begin{bmatrix} \boldsymbol{v} \\ v \end{bmatrix} \\
&= \begin{bmatrix} \boldsymbol{x}^\top \bar{\boldsymbol{W}}_1 & \boldsymbol{x}^\top \boldsymbol{w}_1 \end{bmatrix} \begin{bmatrix} \bar{\boldsymbol{p}}\bar{\boldsymbol{p}}^\top & p\bar{\boldsymbol{p}} \\ p\bar{\boldsymbol{p}}^\top & p^2 \end{bmatrix} \begin{bmatrix} \boldsymbol{v} \\ v \end{bmatrix} + \begin{bmatrix} \boldsymbol{x}^\top \bar{\boldsymbol{W}}_1 & \boldsymbol{x}^\top \boldsymbol{w}_1 \end{bmatrix} \begin{bmatrix} \boldsymbol{X}^\top \boldsymbol{X} & \boldsymbol{X}^\top \boldsymbol{y} \\ \boldsymbol{y}^\top \boldsymbol{X} & \|\boldsymbol{y}\|^2 \end{bmatrix} \begin{bmatrix} \boldsymbol{v} \\ v \end{bmatrix} \\
&= \boldsymbol{x}^\top \tilde{\boldsymbol{p}} + \boldsymbol{x}^\top \left(\bar{\boldsymbol{W}}_1 \boldsymbol{X}^\top \boldsymbol{X} \boldsymbol{v} + (\boldsymbol{w}_1 \boldsymbol{v}^\top + v \bar{\boldsymbol{W}}_1) \boldsymbol{X}^\top \boldsymbol{y} + v \|\boldsymbol{y}\|^2 \boldsymbol{w}_1 \right) \\
&= \boldsymbol{x}^\top \tilde{\boldsymbol{W}} \boldsymbol{X}^\top \boldsymbol{y}_0 + \boldsymbol{x}^\top \tilde{\boldsymbol{p}} + \boldsymbol{x}^\top \underbrace{\left(\bar{\boldsymbol{W}}_1 \boldsymbol{X}^\top \boldsymbol{X} \boldsymbol{v} + (\tilde{\boldsymbol{W}} - v \boldsymbol{w}_1 \boldsymbol{\mu}^\top) \boldsymbol{X}^\top \boldsymbol{X} \boldsymbol{\mu} + v \boldsymbol{w}_1 \|\boldsymbol{y}_0\|^2 \right)}_{\epsilon(\boldsymbol{X}, \boldsymbol{y}_0)}
\end{aligned}$$

where

$$\begin{aligned}
\tilde{\boldsymbol{p}} &:= \bar{\boldsymbol{W}}_1 \bar{\boldsymbol{p}} \bar{\boldsymbol{p}}^\top \boldsymbol{v} + p \boldsymbol{w}_1 \bar{\boldsymbol{p}}^\top \boldsymbol{v} + p v \bar{\boldsymbol{W}}_1 \bar{\boldsymbol{p}} + p^2 v \boldsymbol{w}_1 \\
\tilde{\boldsymbol{W}} &:= \boldsymbol{w}_1 \boldsymbol{v}^\top + v \boldsymbol{W}_1 + 2 v \boldsymbol{w}_1 \boldsymbol{\mu}^\top.
\end{aligned}$$

Then letting $y_0 = y - \boldsymbol{x}^\top \boldsymbol{\mu}_k$, the expected risk of task k obeys

$$\begin{aligned}
\mathbb{E}_{\boldsymbol{Z}, y \sim \mathcal{D}_k} [(\tilde{f}_{\mathrm{att}}(\boldsymbol{Z}^{(k)}) - y)^2] &= \mathbb{E} \left[\left(\boldsymbol{x}^\top \tilde{\boldsymbol{W}} \boldsymbol{X}^\top \boldsymbol{y}_0 - y_0 + \boldsymbol{x}^\top \tilde{\boldsymbol{p}} - \boldsymbol{x}^\top \boldsymbol{\mu} + \boldsymbol{x}^\top \epsilon(\boldsymbol{X}, \boldsymbol{y}_0) \right)^2 \right] \\
&= \mathbb{E} \left[\left(\boldsymbol{x}^\top \tilde{\boldsymbol{W}} \boldsymbol{X}^\top \boldsymbol{y}_0 - y_0 \right)^2 \right] + \mathbb{E} \left[\left(\boldsymbol{x}^\top \tilde{\boldsymbol{p}} - \boldsymbol{x}^\top \boldsymbol{\mu} + \boldsymbol{x}^\top \epsilon(\boldsymbol{X}, \boldsymbol{y}_0) \right)^2 \right] \\
&\quad + 2 \, \mathbb{E} \left[\left(\boldsymbol{x}^\top \tilde{\boldsymbol{W}} \boldsymbol{X}^\top \boldsymbol{y}_0 - y_0 \right) \left(\boldsymbol{x}^\top \tilde{\boldsymbol{p}} - \boldsymbol{x}^\top \boldsymbol{\mu} + \boldsymbol{x}^\top \epsilon(\boldsymbol{X}, \boldsymbol{y}_0) \right) \right]
\end{aligned}$$

Note that, letting $\boldsymbol{\beta}_0 = \boldsymbol{\beta} - \boldsymbol{\mu}_k$, we have $\boldsymbol{y}_0 = \boldsymbol{X} \boldsymbol{\beta}_0 + \boldsymbol{\xi}$, $y = \boldsymbol{x}^\top \boldsymbol{\beta}_0 + \xi_{n+1}$ and $\boldsymbol{\beta}_0 \sim \mathcal{N}(0, \boldsymbol{\Sigma}_{\boldsymbol{\beta}_k})$. Therefore,

$$\begin{aligned}
&\mathbb{E} \left[\left(\boldsymbol{x}^\top \tilde{\boldsymbol{W}} \boldsymbol{X}^\top \boldsymbol{y}_0 - y_0 \right) \left(\boldsymbol{x}^\top \tilde{\boldsymbol{p}} - \boldsymbol{x}^\top \boldsymbol{\mu} + \boldsymbol{x}^\top \epsilon(\boldsymbol{X}, \boldsymbol{y}_0) \right) \right] \\
&= \mathbb{E} \left[\boldsymbol{x}^\top \left(\tilde{\boldsymbol{W}} \boldsymbol{X}^\top \boldsymbol{X} \boldsymbol{\beta}_0 - \boldsymbol{\beta}_0 \right) \left(\tilde{\boldsymbol{p}} - \boldsymbol{\mu} + \epsilon(\boldsymbol{X}, \boldsymbol{y}_0) \right)^\top \boldsymbol{x} \right] \\
&= \mathbb{E} \left[\boldsymbol{x}^\top \left(\tilde{\boldsymbol{W}} \boldsymbol{X}^\top \boldsymbol{X} - \boldsymbol{I} \right) \boldsymbol{\beta}_0 \, \epsilon(\boldsymbol{X}, \boldsymbol{y}_0)^\top \boldsymbol{x} \right] \\
&= \mathbb{E} \left[\boldsymbol{x}^\top \left(\tilde{\boldsymbol{W}} \boldsymbol{X}^\top \boldsymbol{X} - \boldsymbol{I} \right) \boldsymbol{\beta}_0 \left(v \|\boldsymbol{X} \boldsymbol{\beta}_0\|^2 \right) \boldsymbol{w}_1^\top \boldsymbol{x} \right] \\
&= 0.
\end{aligned}$$

Then the risk satisfies

$$\mathbb{E}_{\boldsymbol{Z},y\sim\mathcal{D}_k}[(\tilde{f}_{\text{att}}(\boldsymbol{Z}^{(k)})-y)^2] = \mathbb{E}\left[\left(\boldsymbol{x}^\top\tilde{\boldsymbol{W}}\boldsymbol{X}^\top\boldsymbol{y}_0-y_0\right)^2\right] + \mathbb{E}\left[\left(\boldsymbol{x}^\top\tilde{\boldsymbol{p}}-\boldsymbol{x}^\top\boldsymbol{\mu}+\boldsymbol{x}^\top\epsilon(\boldsymbol{X},\boldsymbol{y}_0)\right)^2\right]$$
$$\geq \mathbb{E}\left[\left(\boldsymbol{x}^\top\tilde{\boldsymbol{W}}\boldsymbol{X}^\top\boldsymbol{y}_0-y_0\right)^2\right].$$

We next prove that the equality is achievable for any $\tilde{\boldsymbol{W}}$. Consider the following constructions:

$$\boldsymbol{W}_q\boldsymbol{W}_k^\top = \begin{bmatrix}\boldsymbol{W} & \boldsymbol{0}\\ \boldsymbol{0}^\top & 0\end{bmatrix}, \quad \boldsymbol{W}_v=\boldsymbol{I}, \quad \boldsymbol{p}_k=\begin{bmatrix}\boldsymbol{W}^{-1}\boldsymbol{\mu}_k\\ \boldsymbol{\mu}_k^\top\boldsymbol{W}^{-\top}\boldsymbol{\mu}_k+1\end{bmatrix} \quad \text{and} \quad \boldsymbol{h}_k=\begin{bmatrix}-\boldsymbol{\mu}_k\\ 1\end{bmatrix}.$$

Then

$$\begin{bmatrix}\boldsymbol{v}\\ v\end{bmatrix}=\begin{bmatrix}-\boldsymbol{\mu}_k\\ 1\end{bmatrix}, \quad \tilde{\boldsymbol{p}}=\boldsymbol{\mu}_k, \quad \text{and} \quad \tilde{\boldsymbol{W}}=\boldsymbol{W}.$$

Using above construction, we obtain that for any $\boldsymbol{W}\in\mathbb{R}^{d\times d}$, there exist $\boldsymbol{p}_k$'s and $\boldsymbol{h}_k$'s such that

$$\mathbb{E}_{\boldsymbol{Z},y\sim\mathcal{D}_k}\left[\left(\boldsymbol{x}^\top\tilde{\boldsymbol{p}}-\boldsymbol{x}^\top\boldsymbol{\mu}+\boldsymbol{x}^\top\epsilon(\boldsymbol{X},\boldsymbol{y}_0)\right)^2\right]=0$$

and hence,

$$\mathbb{E}_{\boldsymbol{Z},y\sim\mathcal{D}_k}[(\tilde{f}_{\text{att}}(\boldsymbol{Z}^{(k)})-y)^2] = \mathbb{E}\left[\left(\boldsymbol{x}^\top\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{y}_0-y_0\right)^2\right].$$

Next, consider the preconditioned gradient descent problem defined in (7.19) in the main paper. Recapping the PGD prediction where we have

$$\tilde{f}_{\text{pgd}}(\boldsymbol{Z}^{(k)}) = \boldsymbol{x}^\top\boldsymbol{W}\boldsymbol{X}^\top(\boldsymbol{y}-\boldsymbol{X}\boldsymbol{\mu}_k)+\boldsymbol{x}^\top\boldsymbol{\mu}_k.$$

Then

$$\mathbb{E}_{\boldsymbol{Z},y\sim\mathcal{D}_k}[(\tilde{f}_{\text{pgd}}(\boldsymbol{Z}^{(k)})-y)^2] = \mathbb{E}_{\boldsymbol{Z},y\sim\mathcal{D}_k}[(\boldsymbol{x}^\top\boldsymbol{W}\boldsymbol{X}^\top(\boldsymbol{y}-\boldsymbol{X}\boldsymbol{\mu}_k)+\boldsymbol{x}^\top\boldsymbol{\mu}_k-y)^2]$$
$$= \mathbb{E}_{\boldsymbol{Z},y\sim\mathcal{D}_k}[(\boldsymbol{x}^\top\boldsymbol{W}\boldsymbol{X}^\top\boldsymbol{y}_0-y_0)^2].$$

Combining the results together completes the proof. ∎

F.3.2 Proof of Theorem 7.4

Theorem F.4 *Consider the multi-task ICL problem with dataset defined in Definition 7.2. Let* $\boldsymbol{W}^\star_{\text{pgd}} := \arg\min_{\boldsymbol{W}}\mathcal{L}(\tilde{f}_{\text{pgd}})$ *following* (7.19) *in the main paper. Define* $\bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}$ *in* (7.14) *in the main paper and let* $\bar{\boldsymbol{W}}^\star_{\text{pgd}}=\boldsymbol{\Sigma}_{\boldsymbol{x}}\boldsymbol{W}^\star_{\text{pgd}}$. *Then the solution* $\bar{\boldsymbol{W}}^\star_{\text{pgd}}$ *and optimal loss* $\tilde{\mathcal{L}}^\star_{\text{pgd}}$

(c.f. (7.19) *in the main paper) satisfy*

$$\bar{\boldsymbol{W}}_{\mathrm{pgd}}^{\star} = \bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} \left((n+1)\bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}} + (\boldsymbol{tr}\left(\bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}\right) + \sigma^2)\boldsymbol{I} \right)^{-1},$$

$$\tilde{\mathcal{L}}_{\mathrm{pgd}}^{\star} = \boldsymbol{tr}\left(\bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}\right) - n\,\boldsymbol{tr}\left(\bar{\boldsymbol{W}}_{\mathrm{pgd}}^{\star}\bar{\boldsymbol{\Sigma}}_{\boldsymbol{\beta}}\right).$$

Proof. From the proof of Proposition 7.1, it can be seen that the optimal multi-task ICL learning performance of a 1-layer linear attention model can be rigorously calculated as: $\tilde{\mathcal{L}}_{\mathrm{att}}^{\star} = \tilde{\mathcal{L}}_{\mathrm{pgd}}^{\star}$. Moreover, in this case, with the help of task-specific heads $\boldsymbol{h}_k$, the optimal loss $\tilde{\mathcal{L}}_{\mathrm{pgd}}^{\star}$ is equivalent to the optimal plain training loss in a **zero task mean** multi-task ICL setting.

By applying Theorem 7.1 with all task means $\boldsymbol{\mu}_k = \mathbf{0}$ for $k \in [K]$, Theorem 7.4 can be proven.

■

APPENDIX G

Appendix for Chapter 8

G.1 Analysis of Single-layer Linear Attention

G.1.1 Supporting Lemmas

Recap the SPI estimator from (SPI). Given a semi-supervised dataset $(\boldsymbol{x}_i, y_i)_{i=1}^n$ as described in Section 8.2.1, let $\mathcal{I}$ denote the token indices set corresponding to the labeled demonstrations, that is, we have

$$y_i = \begin{cases} y_i^c, & i \in \mathcal{I} \\ 0, & otherwise. \end{cases} \tag{G.1}$$

Then, the SPI estimates the task mean via

$$\hat{\boldsymbol{\mu}}_s = \frac{1}{|\mathcal{I}|} \sum_{i \in \mathcal{I}} y_i \boldsymbol{x}_i.$$

Let $\boldsymbol{W} \in \mathbb{R}^{d \times d}$ be the preconditioning matrix. We define the following objective:

$$\boldsymbol{W}^\star := \arg \min_{\boldsymbol{W} \in \mathbb{R}^{d \times d}} \tilde{\mathcal{L}}(\boldsymbol{W}) \quad \text{where} \quad \tilde{\mathcal{L}}(\boldsymbol{W}) = \mathbb{E}\left[\left(\boldsymbol{x}^\top \boldsymbol{W} \sum_{i \in \mathcal{I}} y_i \boldsymbol{x}_i - y \right)^2 \right]. \tag{G.2}$$

Here, we set $(\boldsymbol{x}, y)$ to be the query feature and its corresponding true label. The expectation subsumes the randomness in $(\boldsymbol{x}_i, y_i), (\boldsymbol{x}, y)$ as described in Section 8.2.1.

In the following, we provide a lemma that establishes equivalence between optimizing $\mathcal{L}_{\text{att-1}}(\mathcal{W}^{(1)}, \boldsymbol{h})$ (cf. (8.7) and choosing $L = 1$) and $\tilde{\mathcal{L}}(\boldsymbol{W})$.

Lemma G.1 *Consider ICL problem described in Section 8.2.2 with prompt defined in* (8.3). *Consider training a single-layer linear attention with squared loss, that is,* $L = 1$ *and* $\ell(y, \hat{y}) =$

$(y-\hat{y})^2$. *Recall the objectives from* (8.7) *and* (G.2)*, and let* $\mathcal{L}^\star_{att\text{-}1}$ *and* $\tilde{\mathcal{L}}^\star := \tilde{\mathcal{L}}(\boldsymbol{W}^\star)$ *be their corresponding optimal losses where* $\boldsymbol{W}^\star$ *is defined in* (G.2)*. Then, we have*

$$\mathcal{L}^\star_{att\text{-}1} = \tilde{\mathcal{L}}^\star. \tag{G.3}$$

Additionally, let $f^\star_{att\text{-}1} : \mathbb{R}^{(n+1)\times(d+1)} \to \mathbb{R}$ *denote the optimal prediction (associated with the optimal loss* $\mathcal{L}^\star_{att\text{-}1}$*). We have that* $f^\star_{att\text{-}1}$ *is unique and for any prompt* $\boldsymbol{Z}$ *(cf.* (8.3)*)*

$$f^\star_{att\text{-}1}(\boldsymbol{Z}) = \boldsymbol{x}^\top \boldsymbol{W}^\star \sum_{i\in\mathcal{I}} y_i \boldsymbol{x}_i. \tag{G.4}$$

Proof. Recap the single-layer linear attention model and its prediction from (8.4) and (8.6). We have

$$f_{\text{att-1}}(\boldsymbol{Z}) = \boldsymbol{h}^\top \text{att}(\boldsymbol{Z};\mathcal{W})_{[n+1]} \quad \text{where} \quad \text{att}(\boldsymbol{Z};\mathcal{W}) = (\boldsymbol{Z}\boldsymbol{W}_q\boldsymbol{W}_k^\top\boldsymbol{Z}^\top)\boldsymbol{M}\boldsymbol{Z}\boldsymbol{W}_v \tag{G.5}$$

with $\mathcal{W} := \{\boldsymbol{W}_q, \boldsymbol{W}_k, \boldsymbol{W}_v\}$ being the set of the query, key and value matrices of the attention. Since $\mathcal{W}$ and $\boldsymbol{h}$ are tunable parameters, without loss of generality and for simplicity, let

$$\boldsymbol{W} := \boldsymbol{W}_q\boldsymbol{W}_k^\top \quad \text{and} \quad \bar{\boldsymbol{h}} := \boldsymbol{W}_v\boldsymbol{h}.$$

Following the proof of 116, Proposition 1, similarly, we denote

$$\boldsymbol{W} = \begin{bmatrix} \bar{\boldsymbol{W}} & \boldsymbol{w}_1 \\ \boldsymbol{w}_2^\top & w \end{bmatrix} \qquad \text{and} \qquad \bar{\boldsymbol{h}} = \begin{bmatrix} \boldsymbol{h}_1 \\ h \end{bmatrix},$$

where $\bar{\boldsymbol{W}} \in \mathbb{R}^{d\times d}$, $\boldsymbol{w}_1, \boldsymbol{w}_2, \boldsymbol{h}_1 \in \mathbb{R}^d$, and $w, h \in \mathbb{R}$.

Additionally, let $\mathcal{I}$ denote the token indices set corresponding to the labeled demonstrations (cf. (G.1)). Recall the prompt $\boldsymbol{Z}$ from (8.3), and $\boldsymbol{X} = [\boldsymbol{x}_1 \ \cdots \ \boldsymbol{x}_n]^\top \in \mathbb{R}^{n\times d}$ and $\boldsymbol{y} = [y_1 \ \cdots \ y_n]^\top \in \mathbb{R}^n$ from (8.10). Then we get

$$\boldsymbol{Z} = \begin{bmatrix} \boldsymbol{x}_1 & \boldsymbol{x}_2 & \cdots & \boldsymbol{x}_n & \boldsymbol{x} \\ y_1 & y_2 & \cdots & y_n & 0 \end{bmatrix}^\top = \begin{bmatrix} \boldsymbol{X}^\top & \boldsymbol{x} \\ \boldsymbol{y}^\top & 0 \end{bmatrix}^\top \in \mathbb{R}^{(n+1)\times(d+1)}. \tag{G.6}$$

Combining (G.5) and (G.6) together, we can rewrite the one-layer linear prediction as

$$
\begin{aligned}
f_{\text{att-1}}(\boldsymbol{Z}) &= [\boldsymbol{x}^\top \;\; 0]\boldsymbol{W}\boldsymbol{Z}^\top\boldsymbol{M}\boldsymbol{Z}\bar{\boldsymbol{h}} \\
&= [\boldsymbol{x}^\top \;\; 0]\begin{bmatrix}\bar{\boldsymbol{W}} & \boldsymbol{w}_1 \\ \boldsymbol{w}_2^\top & w\end{bmatrix}\begin{bmatrix}\boldsymbol{X}^\top & \boldsymbol{x} \\ \boldsymbol{y}^\top & 0\end{bmatrix}\begin{bmatrix}\boldsymbol{I}_n & 0 \\ 0 & 0\end{bmatrix}\begin{bmatrix}\boldsymbol{X}^\top & \boldsymbol{x} \\ \boldsymbol{y}^\top & 0\end{bmatrix}^\top\begin{bmatrix}\boldsymbol{h}_1 \\ h\end{bmatrix} \\
&= [\boldsymbol{x}^\top\bar{\boldsymbol{W}} \;\; \boldsymbol{x}^\top\boldsymbol{w}_1]\begin{bmatrix}\boldsymbol{X}^\top\boldsymbol{X} & \boldsymbol{X}^\top\boldsymbol{y} \\ \boldsymbol{y}^\top\boldsymbol{X} & \boldsymbol{y}^\top\boldsymbol{y}\end{bmatrix}\begin{bmatrix}\boldsymbol{h}_1 \\ h\end{bmatrix} \\
&= [\boldsymbol{x}^\top\bar{\boldsymbol{W}} \;\; \boldsymbol{x}^\top\boldsymbol{w}_1]\begin{bmatrix}\boldsymbol{X}^\top\boldsymbol{X}\boldsymbol{h}_1 + h\boldsymbol{X}^\top\boldsymbol{y} \\ \boldsymbol{y}^\top\boldsymbol{X}\boldsymbol{h}_1 + h\boldsymbol{y}^\top\boldsymbol{y}\end{bmatrix} \\
&= \boldsymbol{x}^\top\bar{\boldsymbol{W}}(\boldsymbol{X}^\top\boldsymbol{X}\boldsymbol{h}_1 + h\boldsymbol{X}^\top\boldsymbol{y}) + \boldsymbol{x}^\top\boldsymbol{w}_1(\boldsymbol{y}^\top\boldsymbol{X}\boldsymbol{h}_1 + h\boldsymbol{y}^\top\boldsymbol{y}) \\
&= \boldsymbol{x}^\top(h\bar{\boldsymbol{W}} + \boldsymbol{w}_1\boldsymbol{h}_1^\top)\boldsymbol{X}^\top\boldsymbol{y} + \boldsymbol{x}^\top(\bar{\boldsymbol{W}}\boldsymbol{X}^\top\boldsymbol{X}\boldsymbol{h}_1 + h\boldsymbol{y}^\top\boldsymbol{y}\boldsymbol{w}_1) \\
&= \boldsymbol{x}^\top\tilde{\boldsymbol{W}}\boldsymbol{X}^\top\boldsymbol{y} + \boldsymbol{x}^\top(\bar{\boldsymbol{W}}\boldsymbol{X}^\top\boldsymbol{X}\boldsymbol{h}_1 + mh\boldsymbol{w}_1)
\end{aligned}
$$

where $\tilde{\boldsymbol{W}} := h\bar{\boldsymbol{W}} + \boldsymbol{w}_1\boldsymbol{h}_1^\top$ and we define $m := |\mathcal{I}|$.

Next, recall the loss from (8.7) and consider the squared loss function, $\ell(y, \hat{y}) = (y - \hat{y})^2$. We have

$$
\begin{aligned}
\mathcal{L}_{\text{att-1}}(\mathcal{W}^{(1)}, \boldsymbol{h}) &= \mathbb{E}\left[(f_{\text{att-1}}(\boldsymbol{Z}) - y)^2\right] \\
&= \mathbb{E}\left[\left(\boldsymbol{x}^\top\tilde{\boldsymbol{W}}\boldsymbol{X}\boldsymbol{y} + \boldsymbol{x}^\top\left(\bar{\boldsymbol{W}}\boldsymbol{X}^\top\boldsymbol{X}\boldsymbol{h}_1 + mh\boldsymbol{w}_1\right) - y\right)^2\right] \\
&= \mathbb{E}\left[\left(y\boldsymbol{x}^\top\tilde{\boldsymbol{W}}\boldsymbol{X}\boldsymbol{y} + y\boldsymbol{x}^\top\left(\bar{\boldsymbol{W}}\boldsymbol{X}^\top\boldsymbol{X}\boldsymbol{h}_1 + mh\boldsymbol{w}_1\right) - 1\right)^2\right].
\end{aligned}
$$

For simplicity and without loss of generality, we omit y and use $\boldsymbol{x}$ to represent $y\boldsymbol{x}$. Note that the distribution of (updated) $\boldsymbol{x}$ is not conditioned on its class and given mean vector $\boldsymbol{\mu}$, it follows $\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{\mu}, \sigma^2\boldsymbol{I})$. Similarly, let $\boldsymbol{x}_i$ represent $y_i^c\boldsymbol{x}_i$. We can then write

$$
\begin{aligned}
\mathcal{L}_{\text{att-1}}(\mathcal{W}^{(1)}, \boldsymbol{h}) &= \mathbb{E}\left[\left(\boldsymbol{x}^\top\tilde{\boldsymbol{W}}\sum_{i\in\mathcal{I}}\boldsymbol{x}_i + \boldsymbol{x}^\top\left(\bar{\boldsymbol{W}}\boldsymbol{X}^\top\boldsymbol{X}\boldsymbol{h}_1 + mh\boldsymbol{w}_1\right) - 1\right)^2\right] \qquad \text{(G.7)} \\
&= \mathbb{E}\left[\left(\boldsymbol{x}^\top\tilde{\boldsymbol{W}}\sum_{i\in\mathcal{I}}\boldsymbol{x}_i - 1\right)^2\right] + \mathbb{E}\left[\left(\boldsymbol{x}^\top\left(\bar{\boldsymbol{W}}\boldsymbol{X}^\top\boldsymbol{X}\boldsymbol{h}_1 + mh\boldsymbol{w}_1\right)\right)^2\right] \\
&\quad + 2\,\mathbb{E}\left[\left(\boldsymbol{x}^\top\tilde{\boldsymbol{W}}\sum_{i\in\mathcal{I}}\boldsymbol{x}_i - 1\right)\left(\boldsymbol{x}^\top\left(\bar{\boldsymbol{W}}\boldsymbol{X}^\top\boldsymbol{X}\boldsymbol{h}_1 + mh\boldsymbol{w}_1\right)\right)\right].
\end{aligned}
$$

We start with showing that for any given parameters $\boldsymbol{W} \in \mathbb{R}^{(d+1)\times(d+1)}, \boldsymbol{h} \in \mathbb{R}^{d+1}$, the

component $\mathbb{E}[(\boldsymbol{x}^\top \tilde{\boldsymbol{W}} \sum_{i\in\mathcal{I}} \boldsymbol{x}_i - 1)(\boldsymbol{x}^\top(\bar{\boldsymbol{W}}\boldsymbol{X}^\top \boldsymbol{X}\boldsymbol{h}_1 + mh\boldsymbol{w}_1))] = 0$. To prove it, we first expand

$$
\begin{aligned}
&(\boldsymbol{x}^\top \tilde{\boldsymbol{W}} \sum_{i\in\mathcal{I}} \boldsymbol{x}_i - 1)(\boldsymbol{x}^\top(\bar{\boldsymbol{W}}\boldsymbol{X}^\top \boldsymbol{X}\boldsymbol{h}_1 + mh\boldsymbol{w}_1)) \\
&= \underbrace{(\boldsymbol{x}^\top \tilde{\boldsymbol{W}} \sum_{i\in\mathcal{I}} \boldsymbol{x}_i)(\boldsymbol{x}^\top \bar{\boldsymbol{W}}\boldsymbol{X}^\top \boldsymbol{X}\boldsymbol{h}_1)}_{(a)} - \underbrace{\boldsymbol{x}^\top \bar{\boldsymbol{W}}\boldsymbol{X}^\top \boldsymbol{X}\boldsymbol{h}_1}_{(b)} + \underbrace{(\boldsymbol{x}^\top \tilde{\boldsymbol{W}} \sum_{i\in\mathcal{I}} \boldsymbol{x}_i)(mh\boldsymbol{x}^\top \boldsymbol{w}_1)}_{(c)} - \underbrace{mh\boldsymbol{x}^\top \boldsymbol{w}_1}_{(d)}.
\end{aligned}
$$

In the following, we consider the expectations of $(a), (b), (c), (d)$ sequentially, all of which take the value zero. First note that since $\boldsymbol{\mu} \sim \mathrm{Unif}(\mathbb{S}^{d-1})$ and $(\boldsymbol{\xi}_i)_{i=1}^n, \boldsymbol{\xi} \sim \mathcal{N}(0, \sigma^2 \boldsymbol{I})$, the odd moments of $\boldsymbol{\mu}$, $\boldsymbol{\xi}$ and $\boldsymbol{\xi}_i, i \in [n]$ are all zeros.

$$
\begin{aligned}
(a): \quad & \mathbb{E}\left[(\boldsymbol{x}^\top \tilde{\boldsymbol{W}} \sum_{i\in\mathcal{I}} \boldsymbol{x}_i)(\boldsymbol{x}^\top \bar{\boldsymbol{W}}\boldsymbol{X}^\top \boldsymbol{X}\boldsymbol{h}_1)\right] \\
&= \mathbb{E}\left[(\boldsymbol{\mu}+\boldsymbol{\xi})^\top \tilde{\boldsymbol{W}} \sum_{i\in\mathcal{I}} (\boldsymbol{\mu}+\boldsymbol{\xi}_i)(\boldsymbol{\mu}+\boldsymbol{\xi})^\top \bar{\boldsymbol{W}} \sum_{i\in[n]} (\boldsymbol{\mu}+\boldsymbol{\xi}_i)(\boldsymbol{\mu}+\boldsymbol{\xi}_i)^\top \boldsymbol{h}_1\right] \\
&= \sum_{i\in\mathcal{I}} \sum_{j\in[n]} \mathbb{E}\left[(\boldsymbol{\mu}+\boldsymbol{\xi})^\top \tilde{\boldsymbol{W}} (\boldsymbol{\mu}+\boldsymbol{\xi}_i)(\boldsymbol{\mu}+\boldsymbol{\xi})^\top \bar{\boldsymbol{W}} (\boldsymbol{\mu}+\boldsymbol{\xi}_j)(\boldsymbol{\mu}+\boldsymbol{\xi}_j)^\top \boldsymbol{h}_1\right] \\
&= \sum_{i\in\mathcal{I}} \sum_{j\in[n]} \mathbb{E}\left[\boldsymbol{\mu}^\top \tilde{\boldsymbol{W}}\boldsymbol{\mu}\boldsymbol{\mu}^\top \bar{\boldsymbol{W}}(\boldsymbol{\mu}\boldsymbol{\mu}^\top + \boldsymbol{\xi}_j\boldsymbol{\xi}_j^\top)\boldsymbol{h}_1 + \boldsymbol{\xi}^\top \tilde{\boldsymbol{W}}\boldsymbol{\mu}\boldsymbol{\xi}^\top \bar{\boldsymbol{W}}(\boldsymbol{\mu}\boldsymbol{\mu}^\top + \boldsymbol{\xi}_j\boldsymbol{\xi}_j^\top)\boldsymbol{h}_1\right] \\
&= 0,
\end{aligned}
$$

$$
\begin{aligned}
(b): \quad & \mathbb{E}\left[\boldsymbol{x}^\top \bar{\boldsymbol{W}}\boldsymbol{X}^\top \boldsymbol{X}\boldsymbol{h}_1\right] \\
&= \mathbb{E}\left[(\boldsymbol{\mu}+\boldsymbol{\xi})^\top \bar{\boldsymbol{W}} \sum_{i\in[n]} (\boldsymbol{\mu}+\boldsymbol{\xi}_i)(\boldsymbol{\mu}+\boldsymbol{\xi}_i)^\top \boldsymbol{h}_1\right] \\
&= \mathbb{E}\left[\boldsymbol{\mu}^\top \bar{\boldsymbol{W}} \sum_{i\in[n]} (\boldsymbol{\mu}\boldsymbol{\mu}^\top + \boldsymbol{\xi}_i\boldsymbol{\xi}_i^\top)\boldsymbol{h}_1\right] \\
&= 0,
\end{aligned}
$$

$$
\begin{aligned}
(c): \quad & \mathbb{E}\left[(\boldsymbol{x}^\top \tilde{\boldsymbol{W}} \sum_{i\in\mathcal{I}} \boldsymbol{x}_i)(mh\boldsymbol{x}^\top \boldsymbol{w}_1)\right] \\
&= mh\,\mathbb{E}\left[(\boldsymbol{\mu}+\boldsymbol{\xi})^\top \tilde{\boldsymbol{W}} \sum_{i\in\mathcal{I}} (\boldsymbol{\mu}+\boldsymbol{\xi}_i)(\boldsymbol{\mu}+\boldsymbol{\xi})^\top \boldsymbol{w}_1\right] \\
&= mh \sum_{i\in\mathcal{I}} \mathbb{E}\left[(\boldsymbol{\mu}+\boldsymbol{\xi})^\top \tilde{\boldsymbol{W}} \boldsymbol{\mu}(\boldsymbol{\mu}+\boldsymbol{\xi})^\top \boldsymbol{w}_1\right] \\
&= mh \sum_{i\in\mathcal{I}} \mathbb{E}\left[\boldsymbol{\mu}^\top \tilde{\boldsymbol{W}} \boldsymbol{\mu}\boldsymbol{\mu}^\top \boldsymbol{w}_1 + \boldsymbol{\xi}^\top \tilde{\boldsymbol{W}} \boldsymbol{\mu}\boldsymbol{\xi}^\top \boldsymbol{w}_1\right] \\
&= 0,
\end{aligned}
$$

$$
(d): \quad \mathbb{E}\left[mh\boldsymbol{x}^\top \boldsymbol{w}_1\right] = 0.
$$

Therefore, loss in (G.7) returns

$$
\mathcal{L}_{\text{att-1}}(\mathcal{W}^{(1)}, \boldsymbol{h}) = \underbrace{\mathbb{E}\left[\left(\boldsymbol{x}^\top \tilde{\boldsymbol{W}} \sum_{i\in\mathcal{I}} \boldsymbol{x}_i - 1\right)^2\right]}_{\tilde{\mathcal{L}}(\tilde{\boldsymbol{W}})} + \mathbb{E}\left[\left(\boldsymbol{x}^\top (\bar{\boldsymbol{W}} \boldsymbol{X}^\top \boldsymbol{X} \boldsymbol{h}_1 + mh\boldsymbol{w}_1)\right)^2\right].
$$

Here, the first term $\mathbb{E}[(\boldsymbol{x}^\top \tilde{\boldsymbol{W}} \sum_{i\in\mathcal{I}} \boldsymbol{x}_i - 1)^2] = \tilde{\mathcal{L}}(\tilde{\boldsymbol{W}})$ where $\tilde{\mathcal{L}}(\tilde{\boldsymbol{W}})$ is defined in (G.2).

Recall that $\tilde{\boldsymbol{W}} = h\bar{\boldsymbol{W}} + \boldsymbol{w}_1 \boldsymbol{h}_1^\top$. Then for any $\tilde{\boldsymbol{W}} \in \mathbb{R}^{d\times d}$, setting $\boldsymbol{h}_1 = \boldsymbol{w}_1 = \boldsymbol{0}_d$ and $h = 1$ returns $\mathbb{E}\left[\left(\boldsymbol{x}^\top \left(\bar{\boldsymbol{W}} \boldsymbol{X}^\top \boldsymbol{X} \boldsymbol{h}_1 + mh\boldsymbol{w}_1\right)\right)^2\right] = 0$, and then

$$
\mathcal{L}_{\text{att-1}}(\mathcal{W}^{(1)}, \boldsymbol{h}) = \mathbb{E}\left[\left(\boldsymbol{x}^\top \bar{\boldsymbol{W}} \sum_{i\in\mathcal{I}} \boldsymbol{x}_i - 1\right)^2\right]
$$

Therefore, optimizing $\mathcal{L}_{\text{att-1}}(\mathcal{W}^{(1)}, \boldsymbol{h})$ returns the same minima as optimizing $\tilde{\mathcal{L}}(\boldsymbol{W})$, which completes the proof of (G.3). Note that optimal loss $\mathcal{L}^\star_{\text{att-1}}$ depends on the labeled data $i \in \mathcal{I}$ only.

Furthermore, since $\tilde{\mathcal{L}}(\boldsymbol{W})$ is strongly convex (see (G.8)), $\boldsymbol{W}^\star$ exists and is unique. Therefore, (G.3) and uniqueness of $\boldsymbol{W}^\star$ leads to the conclusion (G.4). ■

Lemma G.2 *Consider the objective defined in* (G.2) *with semi-supervised data following*

Section 8.2. Then the optimal solution $\boldsymbol{W}^\star$ satisfies

$$\boldsymbol{W}^\star = c\boldsymbol{I}$$

for some $c > 0$.

Proof. Recap the Objective (G.2) and its optimal solution $\boldsymbol{W}^\star$. Let $\mathcal{I}$ be the index set corresponding the labeled in-context examples, and $|\mathcal{I}| = m$. Note that, m is also a random variable, independent of $\boldsymbol{x}_i, y_i^c, \boldsymbol{x}, y$.

As in the proof of Lemma G.1, we use $\boldsymbol{x}$ to represent $y\boldsymbol{x}$ and $\boldsymbol{x}_i$ to represent $y_i^c\boldsymbol{x}_i$ for simplicity, where (updated) $\boldsymbol{x}_i, \boldsymbol{x} \sim \mathcal{N}(\boldsymbol{\mu}, \sigma^2\boldsymbol{I})$. Letting $\boldsymbol{\xi}', \boldsymbol{\xi}, \boldsymbol{\xi}_i \sim \mathcal{N}(0, \sigma^2\boldsymbol{I})$ be independent, we obtain

$$\begin{aligned}
\tilde{\mathcal{L}}(\boldsymbol{W}) &= \mathbb{E}\left[(\boldsymbol{x}^\top \boldsymbol{W} \sum_{i\in\mathcal{I}} \boldsymbol{x}_i - 1)^2\right] \qquad \text{(G.8)}\\
&= \mathbb{E}\left[((\boldsymbol{\mu}+\boldsymbol{\xi})^\top \boldsymbol{W} \sum_{i\in\mathcal{I}} (\boldsymbol{\mu}+\boldsymbol{\xi}_i) - 1)^2\right]\\
&= \mathbb{E}\left[((\boldsymbol{\mu}+\boldsymbol{\xi})^\top \boldsymbol{W}(m\boldsymbol{\mu} + \sqrt{m}\boldsymbol{\xi}') - 1)^2\right]\\
&= \mathbb{E}\left[m^2(\boldsymbol{\mu}^\top\boldsymbol{W}\boldsymbol{\mu})^2 + m(\boldsymbol{\mu}^\top\boldsymbol{W}\boldsymbol{\xi}')^2 + m^2(\boldsymbol{\xi}^\top\boldsymbol{W}\boldsymbol{\mu})^2 + m(\boldsymbol{\xi}^\top\boldsymbol{W}\boldsymbol{\xi}')^2 + 1\right] - 2\,\mathbb{E}\left[m\boldsymbol{\mu}^\top\boldsymbol{W}\boldsymbol{\mu}\right]\\
&= \frac{\mathbb{E}[m^2]}{d(d+2)}(\mathtt{tr}\left(\boldsymbol{W}\right)^2 + \mathtt{tr}\left(\boldsymbol{W}\boldsymbol{W}^\top\right) + \mathtt{tr}\left(\boldsymbol{W}^2\right)) + \frac{\mathbb{E}[m+m^2]}{d}\sigma^2\mathtt{tr}\left(\boldsymbol{W}\boldsymbol{W}^\top\right)\\
&\quad + \mathbb{E}[m]\sigma^4\mathtt{tr}\left(\boldsymbol{W}\boldsymbol{W}^\top\right) + 1 - \frac{2\,\mathbb{E}[m]}{d}\mathtt{tr}\left(\boldsymbol{W}\right).
\end{aligned}$$

Differentiating it results in

$$\nabla_{\boldsymbol{W}}\tilde{\mathcal{L}}(\boldsymbol{W}) = \frac{2\,\mathbb{E}[m^2]}{d(d+2)}(\mathtt{tr}\left(\boldsymbol{W}\right)\boldsymbol{I} + \boldsymbol{W} + \boldsymbol{W}^\top) + \frac{2\,\mathbb{E}[m+m^2]\sigma^2}{d}\boldsymbol{W} + 2\,\mathbb{E}[m]\sigma^4\boldsymbol{W} - \frac{2\,\mathbb{E}[m]}{d}\boldsymbol{I}.$$

Setting $\nabla_{\boldsymbol{W}}\tilde{\mathcal{L}}(\boldsymbol{W}) = 0$, we obtain the optimal $\boldsymbol{W}^\star$

$$\boldsymbol{W}^\star = \frac{1}{(1+\sigma^2)\,\mathbb{E}[m^2]/\,\mathbb{E}[m] + \sigma^2 + \sigma^4 d}\boldsymbol{I},$$

which leads to the conclusion that $\boldsymbol{W}^\star = c\boldsymbol{I}$, for $c = \frac{1}{(1+\sigma^2)\,\mathbb{E}[m^2]/\,\mathbb{E}[m]+\sigma^2+\sigma^4 d} > 0$. It completes the proof. ∎

G.1.2 Proof of Theorem 8.1

Proof. Note that (8.8) can be easily proven using Lemmas G.1 and G.2. Then, we focus on proving (8.9).

Given that (8.8) holds, we can rewrite its classification error as

$$\mathbb{P}(y^\star_{\text{att-1}}(\boldsymbol{Z}) \neq y) = \mathbb{P}(\operatorname{sgn}(\boldsymbol{x}^\top \hat{\boldsymbol{\mu}}_s) \neq y) = \mathbb{P}(\operatorname{sgn}(y\boldsymbol{x}^\top \hat{\boldsymbol{\mu}}_s) \neq 1) \tag{G.9}$$

where $\hat{\boldsymbol{\mu}}_s = \frac{1}{|\mathcal{I}|}\sum_{i\in\mathcal{I}} y_i \boldsymbol{x}_i$ defined in (SPI) and $\mathcal{I}$ is the index set of labeled samples. Let $m = |\mathcal{I}|$.

Recall from Section 8.2.1 where $\boldsymbol{x} \sim \mathcal{N}(y \cdot \boldsymbol{\mu}, \sigma^2 \boldsymbol{I})$. We can rewrite

$$y\boldsymbol{x} = \boldsymbol{\mu} + \sigma \boldsymbol{g}_1 \quad \text{where} \quad \boldsymbol{g}_1 \sim \mathcal{N}(0, \boldsymbol{I}).$$

Then for any given $\boldsymbol{\mu}, \hat{\boldsymbol{\mu}}_s$, we get

$$\begin{aligned}
\mathbb{P}\left(\operatorname{sgn}(y\boldsymbol{x}^\top \hat{\boldsymbol{\mu}}_s) \neq 1 \;\middle|\; \boldsymbol{\mu}, \hat{\boldsymbol{\mu}}_s\right) &= \mathbb{P}\left((\boldsymbol{\mu} + \sigma\boldsymbol{g}_1)^\top \hat{\boldsymbol{\mu}}_s < 0 \;\middle|\; \boldsymbol{\mu}, \hat{\boldsymbol{\mu}}_s\right) \\
&= \mathbb{P}\left(\boldsymbol{\mu}^\top \hat{\boldsymbol{\mu}}_s < \sigma \boldsymbol{g}_1^\top \hat{\boldsymbol{\mu}}_s \;\middle|\; \boldsymbol{\mu}, \hat{\boldsymbol{\mu}}_s\right) \\
&= Q\left(\frac{\boldsymbol{\mu}^\top \hat{\boldsymbol{\mu}}_s}{\sigma \|\hat{\boldsymbol{\mu}}_s\|}\right).
\end{aligned} \tag{G.10}$$

Here Q-function is the tail distribution function of the standard normal distribution.

Next, similarly, given that $\boldsymbol{x}_i \sim \mathcal{N}(y_i \cdot \boldsymbol{\mu}, \sigma^2 \boldsymbol{I})$ for $i \in \mathcal{I}$, we can rewrite

$$\hat{\boldsymbol{\mu}}_s = \frac{1}{m}\sum_{i\in\mathcal{I}} y_i \boldsymbol{x}_i = \boldsymbol{\mu} + \frac{\sigma}{\sqrt{m}}\boldsymbol{g}_2 \quad \text{where} \quad \boldsymbol{g}_2 \sim \mathcal{N}(0, \boldsymbol{I}).$$

Then combining (G.9) and (G.10), we have

$$\begin{aligned}
\mathbb{P}(y^\star_{\text{att-1}}(\boldsymbol{Z}) \neq y) &= \mathbb{E}_{\boldsymbol{\mu},\boldsymbol{g}_2}\left[Q\left(\frac{\boldsymbol{\mu}^\top \hat{\boldsymbol{\mu}}_s}{\sigma\|\hat{\boldsymbol{\mu}}_s\|}\right)\right] \\
&= \mathbb{E}_{\boldsymbol{\mu},\boldsymbol{g}_2}\left[Q\left(\frac{\boldsymbol{\mu}^\top(\boldsymbol{\mu} + \frac{\sigma}{\sqrt{m}}\boldsymbol{g}_2)}{\sigma\|\boldsymbol{\mu} + \frac{\sigma}{\sqrt{m}}\boldsymbol{g}_2\|}\right)\right] \\
&= \mathbb{E}_{\boldsymbol{\mu},\boldsymbol{g}_2}\left[Q\left(\frac{1 + \frac{\sigma}{\sqrt{m}}\boldsymbol{\mu}^\top \boldsymbol{g}_2}{\sigma\sqrt{1 + 2\frac{\sigma}{\sqrt{m}}\boldsymbol{\mu}^\top \boldsymbol{g}_2 + \frac{\sigma^2}{m}\|\boldsymbol{g}_2\|^2}}\right)\right].
\end{aligned}$$

Note that for any $\boldsymbol{\mu}$ with $\|\boldsymbol{\mu}\| = 1$, we have $\boldsymbol{\mu}^\top \boldsymbol{g}_2 \sim \mathcal{N}(0,1)$. Therefore, we can write

$$\boldsymbol{\mu}^\top \boldsymbol{g}_2 = g \quad \text{where} \quad g \sim \mathcal{N}(0,1),$$

and let $\boldsymbol{U} \in \mathbb{R}^{d\times d}$ be a unitary matrix with first row being $\boldsymbol{\mu}$. We can write

$$\|\boldsymbol{g}_2\|^2 = \|\boldsymbol{U}\boldsymbol{g}_2\|^2 = g^2 + h \quad \text{where} \quad h \sim \mathcal{X}^2_{d-1}.$$

Here, $\mathcal{X}^2_{d-1}$ denotes chi-squared distribution with $(d-1)$ degrees of freedom. Then, we get

$$\begin{aligned}
\mathbb{P}(y^\star_{\text{att-1}}(\boldsymbol{Z}) \neq y) &= \mathbb{E}_{g,h}\left[Q\left(\frac{1+\frac{\sigma}{\sqrt{m}}g}{\sigma\sqrt{1+2\frac{\sigma}{\sqrt{m}}g+\frac{\sigma^2}{m}(g^2+h)}}\right)\right] \\
&= \mathbb{E}_{g,h}\left[Q\left(\frac{1+\frac{\sigma}{\sqrt{m}}g}{\sigma\sqrt{(1+\frac{\sigma}{\sqrt{m}}g)^2+\frac{\sigma^2}{m}h}}\right)\right], \\
&= \mathbb{E}_{g,h}\left[Q\left(\frac{1+\epsilon_\sigma g}{\sigma\sqrt{(1+\epsilon_\sigma g)^2+\epsilon_\sigma^2 h}}\right)\right],
\end{aligned}$$

where $\epsilon_\sigma := \sigma/\sqrt{m}$. It completes the proof of (8.9).

Next, we derive an upper bound for $\mathbb{P}(y^\star_{\text{att-1}}(\boldsymbol{Z}) \neq y)$. Let $c := \epsilon_\sigma^{-1}$. Then we have

$$\begin{aligned}
\mathbb{P}(y^\star_{\text{att-1}}(\boldsymbol{Z}) \neq y) &= \mathbb{E}_{g,h}\left[Q\left(\frac{c+g}{\sigma\sqrt{(c+g)^2+h}}\right)\right] \\
&= \mathbb{E}_{g\geq -\frac{c}{2},h}\left[Q\left(\frac{c+g}{\sigma\sqrt{(c+g)^2+h}}\right)\right] + \mathbb{E}_{g<-\frac{c}{2},h}\left[Q\left(\frac{c+g}{\sigma\sqrt{(c+g)^2+h}}\right)\right] \\
&\leq \mathbb{E}_{g\geq -\frac{c}{2},h}\left[Q\left(\frac{c+g}{\sigma\sqrt{(c+g)^2+h}}\right)\right] + Q(c/2) \\
&= \mathbb{E}_{g\geq -\frac{c}{2},h}\left[Q\left(\frac{1}{\sigma\sqrt{1+h/(c+g)^2}}\right)\right] + Q(c/2), \qquad \text{(G.11)}
\end{aligned}$$

where the inequality comes from the fact that $\mathbb{P}(g \leq -c/2) = Q(c/2)$ and $Q(x) \leq 1$ for any $x \in \mathbb{R}$. Next, we have

$$\frac{1}{\sqrt{1+h/(c+g)^2}} \geq 1 - \frac{1}{2}\frac{h}{(c+g)^2} \geq 1 - \frac{2h}{c^2}.$$

Here the first inequality comes from that $\frac{1}{\sqrt{1+x}} \geq 1 - \frac{1}{2}x$ and the second utilizes that $g \geq -\frac{c}{2}$.

Since $h \sim \mathcal{X}^2_{d-1}$, from the Laurent-Massart inequality [101], we have that

$$\mathbb{P}\left(h \geq d - 1 + 2\sqrt{(d-1)t_1} + 2t_1\right) \leq e^{-t_1}.$$

Therefore, we have that with probability at least $1 - e^{-t_1}$

$$\frac{1}{\sqrt{1 + h/(c+g)^2}} \geq 1 - \frac{2(d - 1 + 2\sqrt{(d-1)t_1} + 2t_1)}{c^2}.$$

Setting $t_1 = d$, we get with probability at least $1 - e^{-d}$

$$\frac{1}{\sqrt{1 + h/(c+g)^2}} \geq 1 - \frac{10d}{c^2}.$$

Combining the result with (G.11), since $Q(x) \leq 1$ for $x \in \mathbb{R}$ and $Q(x) \leq e^{-x^2/2}$ for $x > 1$, we get that

$$\begin{aligned}\mathbb{P}(y^\star_{\text{att-1}}(\boldsymbol{Z}) \neq y) &\leq e^{-d} + Q(c/2) + Q\left(\frac{1}{\sigma}\left(1 - \frac{10d}{c^2}\right)\right) \\ &\leq e^{-d} + e^{-1/8\epsilon_\sigma^2} + Q\left(\frac{1}{\sigma}\left(1 - 10d\epsilon_\sigma^2\right)\right).\end{aligned}$$

It completes the proof.

■

G.2 Analysis of Multi-layer Linear Attention

G.2.1 Proof of Proposition 8.1

Proof. We consider the following model constructions for the attention matrices in the ℓth layer, $\ell \in [L]$ and the final linear prediction head:

$$\begin{aligned}&\ell\text{th layer:} \quad \boldsymbol{W}_{q\ell}\boldsymbol{W}_{k\ell}^\top = \begin{bmatrix}\boldsymbol{I}_d & 0 \\ 0 & 0\end{bmatrix} \quad \text{and} \quad \boldsymbol{W}_{v\ell} = \begin{bmatrix}a_\ell \boldsymbol{I}_d & 0 \\ 0 & b_\ell\end{bmatrix}; \\ &\text{Prediction head:} \quad \boldsymbol{h} = \begin{bmatrix}\boldsymbol{0}_d \\ c\end{bmatrix}.\end{aligned} \tag{G.12}$$

Suppose the input to ℓth layer is

$$\boldsymbol{Z}_\ell = \begin{bmatrix} \boldsymbol{X}_\ell & \boldsymbol{y}_\ell \\ \boldsymbol{x}_\ell^\top & y_\ell \end{bmatrix} \in \mathbb{R}^{(n+1)\times(d+1)} \quad \text{where} \quad \boldsymbol{Z}_1 = \boldsymbol{Z} = \begin{bmatrix} \boldsymbol{X} & \boldsymbol{y} \\ \boldsymbol{x}^\top & 0 \end{bmatrix}.$$

Recapping the model construction from (G.12), the ℓth layer output returns

$$\begin{aligned} \left(\boldsymbol{Z}_\ell \boldsymbol{W}_{q\ell} \boldsymbol{W}_{k\ell}^\top \boldsymbol{Z}_\ell^\top \boldsymbol{M}\right) \boldsymbol{Z}_\ell \boldsymbol{W}_{v\ell} &= \begin{bmatrix} \boldsymbol{X}_\ell & \boldsymbol{y}_\ell \\ \boldsymbol{x}_\ell^\top & y_\ell \end{bmatrix} \begin{bmatrix} \boldsymbol{I}_d & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{X}_\ell^\top & \boldsymbol{x}_\ell \\ \boldsymbol{y}_\ell^\top & y_\ell \end{bmatrix} \boldsymbol{M} \begin{bmatrix} \boldsymbol{X}_\ell & \boldsymbol{y}_\ell \\ \boldsymbol{x}_\ell^\top & y_\ell \end{bmatrix} \begin{bmatrix} a_\ell \boldsymbol{I}_d & 0 \\ 0 & b_\ell \end{bmatrix} \\ &= \begin{bmatrix} \boldsymbol{X}_\ell \boldsymbol{X}_\ell^\top & \boldsymbol{X}_\ell \boldsymbol{x}_\ell \\ \boldsymbol{x}_\ell^\top \boldsymbol{X}_\ell^\top & \boldsymbol{x}_\ell^\top \boldsymbol{x}_\ell \end{bmatrix} \begin{bmatrix} \boldsymbol{I}_n & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} a_\ell \boldsymbol{X}_\ell & b_\ell \boldsymbol{y}_\ell \\ a_\ell \boldsymbol{x}_\ell^\top & b_\ell y_\ell \end{bmatrix} \\ &= \begin{bmatrix} a_\ell \boldsymbol{X}_\ell \boldsymbol{X}_\ell^\top \boldsymbol{X}_\ell & b_\ell \boldsymbol{X}_\ell \boldsymbol{X}_\ell^\top \boldsymbol{y}_\ell \\ a_\ell \boldsymbol{x}_\ell^\top \boldsymbol{X}_\ell^\top \boldsymbol{X}_\ell & b_\ell \boldsymbol{x}_\ell^\top \boldsymbol{X}_\ell^\top \boldsymbol{y}_\ell \end{bmatrix}. \end{aligned} \tag{G.13}$$

Therefore, following (8.5), the input of $(\ell+1)$th layer is

$$\begin{aligned} \boldsymbol{Z}_{\ell+1} &= \boldsymbol{Z}_\ell + \begin{bmatrix} a_\ell \boldsymbol{X}_\ell \boldsymbol{X}_\ell^\top \boldsymbol{X}_\ell & b_\ell \boldsymbol{X}_\ell \boldsymbol{X}_\ell^\top \boldsymbol{y}_\ell \\ a_\ell \boldsymbol{x}_\ell^\top \boldsymbol{X}_\ell^\top \boldsymbol{X}_\ell & b_\ell \boldsymbol{x}_\ell^\top \boldsymbol{X}_\ell^\top \boldsymbol{y}_\ell \end{bmatrix} \\ &= \begin{bmatrix} \boldsymbol{X}_\ell + a_\ell \boldsymbol{X}_\ell \boldsymbol{X}_\ell^\top \boldsymbol{X}_\ell & \boldsymbol{y}_\ell + b_\ell \boldsymbol{X}_\ell \boldsymbol{X}_\ell^\top \boldsymbol{y}_\ell \\ \boldsymbol{x}_\ell^\top + a_\ell \boldsymbol{x}_\ell^\top \boldsymbol{X}_\ell^\top \boldsymbol{X}_\ell & y_\ell + b_\ell \boldsymbol{x}_\ell^\top \boldsymbol{X}_\ell^\top \boldsymbol{y}_\ell \end{bmatrix} \in \mathbb{R}^{(n+1)\times(d+1)}. \end{aligned} \tag{G.14}$$

• **Label propagation:** We first focus on deriving label propagation results. Suppose that we have

$$a_\ell = 0 \quad \text{for} \quad \ell \in [L].$$

Then following (G.13), the output of ℓ'th layer takes the following form:

$$\left(\boldsymbol{Z}_\ell \boldsymbol{W}_{q\ell} \boldsymbol{W}_{k\ell}^\top \boldsymbol{Z}_\ell^\top \boldsymbol{M}\right) \boldsymbol{Z}_\ell \boldsymbol{W}_{v\ell} = \begin{bmatrix} 0 & b_\ell \boldsymbol{X}_\ell \boldsymbol{X}_\ell^\top \boldsymbol{y}_\ell \\ 0 & b_\ell \boldsymbol{x}_\ell^\top \boldsymbol{X}_\ell^\top \boldsymbol{y}_\ell \end{bmatrix}.$$

Here, the first d coordinates of each token's output are zeros, and therefore, the corresponding input coordinates remain unchanged, and we have

$$\boldsymbol{X}_\ell \equiv \boldsymbol{X} \quad \text{and} \quad \boldsymbol{x}_\ell \equiv \boldsymbol{x} \quad \text{for} \quad \ell \in [L].$$

The prediction (based on the last token output and after applying prediction head) is given by

$$f_{\text{all-}L}(\boldsymbol{Z}) = cb_L \boldsymbol{x}^\top \boldsymbol{X}^\top \boldsymbol{y}_L. \tag{G.15}$$

We next focus on obtaining $\boldsymbol{y}_L$. From (G.14), we have

$$\boldsymbol{y}_{\ell+1} = \boldsymbol{y}_\ell + b_\ell \boldsymbol{X}\boldsymbol{X}^\top \boldsymbol{y}_\ell = (\boldsymbol{I} + b_\ell \boldsymbol{X}\boldsymbol{X}^\top)\boldsymbol{y}_\ell.$$

Therefore,

$$\boldsymbol{y}_L = \prod_{\ell=1}^{L-1} (\boldsymbol{I} + b_\ell \boldsymbol{X}\boldsymbol{X}^\top)\boldsymbol{y}.$$

Combining with (G.15) results in

$$f_{\text{all-}L}(\boldsymbol{Z}) = cb_L \boldsymbol{x}^\top \boldsymbol{X}^\top \prod_{\ell=1}^{L-1} (\boldsymbol{I} + b_\ell \boldsymbol{X}\boldsymbol{X}^\top)\boldsymbol{y} = cb_L \boldsymbol{x}^\top \prod_{\ell=1}^{L-1} (\boldsymbol{I} + b_\ell \boldsymbol{X}^\top\boldsymbol{X})\boldsymbol{X}^\top \boldsymbol{y}.$$

It completes the proof.

• **Feature propagation:** We now focus on the feature propagation setting. In contrast to the label propagation, let us assume that

$$a_\ell \to \infty \quad \text{and} \quad b_\ell \to 0^+ \quad \text{for} \quad \ell \in [L].$$

The prediction (following (G.13), based on the last token output and after applying prediction head) is given by

$$f_{\text{all-}L}(\boldsymbol{Z}) = cb_L \boldsymbol{x}_L^\top \boldsymbol{X}_L^\top \boldsymbol{y}_L. \tag{G.16}$$

We first obtain $\boldsymbol{y}_L$. From (G.14) (since $b_\ell \to 0$), we have

$$\boldsymbol{y}_{\ell+1} = \boldsymbol{y}_\ell + b_\ell \boldsymbol{X}\boldsymbol{X}^\top \boldsymbol{y}_\ell = \boldsymbol{y}_\ell.$$

Therefore,

$$\boldsymbol{y}_\ell \equiv \boldsymbol{y} \quad \text{for} \quad \ell \in [L].$$

Next, we focus on $\boldsymbol{X}_L, \boldsymbol{x}_L$. From (G.14), as $a_\ell \to \infty$, we have

$$\begin{aligned}
\boldsymbol{X}_{\ell+1} &= \boldsymbol{X}_\ell + a_\ell \boldsymbol{X}_\ell \boldsymbol{X}_\ell^\top \boldsymbol{X}_\ell = \boldsymbol{X}_\ell(\boldsymbol{I} + a_\ell \boldsymbol{X}_\ell^\top \boldsymbol{X}_\ell) = a_\ell \boldsymbol{X}_\ell \boldsymbol{X}_\ell^\top \boldsymbol{X}_\ell; \\
\boldsymbol{x}_{\ell+1}^\top &= \boldsymbol{x}_\ell^\top + a_\ell \boldsymbol{x}_\ell^\top \boldsymbol{X}_\ell^\top \boldsymbol{X}_\ell = \boldsymbol{x}_\ell^\top(\boldsymbol{I} + a_\ell \boldsymbol{X}_\ell^\top \boldsymbol{X}_\ell) = a_\ell \boldsymbol{x}_\ell^\top \boldsymbol{X}_\ell^\top \boldsymbol{X}_\ell.
\end{aligned}$$

Therefore,

$$\begin{aligned}
\boldsymbol{X}_L &= a_{L-1} \boldsymbol{X}_{L-1}(\boldsymbol{X}_{L-1}^\top \boldsymbol{X}_{L-1}) \\
&= a_{L-1} a_{L-2}^3 \boldsymbol{X}_{L-2}(\boldsymbol{X}_{L-2}^\top \boldsymbol{X}_{L-2})^{\frac{3^2-1}{2}} \\
&= a_{L-1} a_{L-2}^3 a_{L-3}^{3^2} \boldsymbol{X}_{L-3}(\boldsymbol{X}_{L-3}^\top \boldsymbol{X}_{L-3})^{\frac{3^3-1}{2}} \\
&= \cdots \\
&= a_{L-1} a_{L-2}^3 a_{L-3}^{3^2} \ldots a_1^{3^{L-2}} \boldsymbol{X}(\boldsymbol{X}^\top \boldsymbol{X})^{\frac{3^{L-1}-1}{2}},
\end{aligned}$$

and

$$\begin{aligned}
\boldsymbol{x}_L^\top &= a_{L-1} \boldsymbol{x}_{L-1}^\top(\boldsymbol{X}_{L-1}^\top \boldsymbol{X}_{L-1}) \\
&= a_{L-1} a_{L-2}^3 \boldsymbol{x}_{L-2}^\top(\boldsymbol{X}_{L-2}^\top \boldsymbol{X}_{L-2})^{\frac{3^2-1}{2}} \\
&= a_{L-1} a_{L-2}^3 a_{L-3}^{3^2} \boldsymbol{x}_{L-3}^\top(\boldsymbol{X}_{L-3}^\top \boldsymbol{X}_{L-3})^{\frac{3^3-1}{2}} \\
&= \cdots \\
&= a_{L-1} a_{L-2}^3 a_{L-3}^{3^2} \ldots a_1^{3^{L-2}} \boldsymbol{x}^\top(\boldsymbol{X}^\top \boldsymbol{X})^{\frac{3^{L-1}-1}{2}}.
\end{aligned}$$

Combining all together with (G.16), we have that

$$\begin{aligned}
f_{\text{all-}L}(\boldsymbol{Z}) &= c b_L \boldsymbol{x}_L^\top \boldsymbol{X}_L^\top \boldsymbol{y}_L \\
&= c b_L \left(\prod_{\ell=1}^{L-1} a_\ell^{3^{L-1-\ell}} \right)^2 \boldsymbol{x}^\top (\boldsymbol{X}^\top \boldsymbol{X})^{3^{L-1}-1} \boldsymbol{X}^\top \boldsymbol{y}.
\end{aligned}$$

It completes the proof. ∎

G.2.2 Proof of Proposition 8.2

Proof. The proof follows directly by adopting the same model construction and proof strategy as in Proposition 8.1, under the additional assumption that

$$a_\ell = a \quad \text{and} \quad b_\ell = b \quad \text{for} \quad \ell \in [L].$$

■

G.2.3 Proof of Lemma 8.1

Proof. In the proof of Proposition 8.1, we showed how to derive the label and feature propagation results by restricting the construction to either $a_\ell \equiv 0$ (for label propagation) or $(a_\ell \to \infty, b_\ell \to 0)$ (for feature propagation). Here, we consider a propagation process without imposing restrictions on the choices of (a_ℓ, b_ℓ), and study the form of the final prediction returned by the model.

To avoid the notation conflict, we express the matrix $\boldsymbol{A}$ in (8.12) as

$$\boldsymbol{A} = \sum_{k=0}^{K} e_k (\boldsymbol{X}^\top \boldsymbol{X})^k$$

and let $\boldsymbol{e} = [e_0 \ e_2 \ \cdots \ e_{(3^L-3)/2}]^\top \in \mathbb{R}^{K+1}$.

Recall the same model construction used in the proof of Proposition 8.1, defined in (G.12). From (G.13), we have that

$$f_{\text{att-}L}(\boldsymbol{Z}) = c b_L \boldsymbol{x}_L^\top \boldsymbol{X}_L^\top \boldsymbol{y}_L$$

where following (G.14), we have

$$\begin{aligned}
\boldsymbol{X}_{\ell+1} &= \boldsymbol{X}_\ell (\boldsymbol{I} + a_\ell \boldsymbol{X}_\ell^\top \boldsymbol{X}_\ell), \\
\boldsymbol{x}_{\ell+1}^\top &= \boldsymbol{x}_\ell^\top (\boldsymbol{I} + a_\ell \boldsymbol{X}_\ell^\top \boldsymbol{X}_\ell), \\
\boldsymbol{y}_{\ell+1} &= (\boldsymbol{I} + b_\ell \boldsymbol{X}_\ell \boldsymbol{X}_\ell^\top) \boldsymbol{y}_\ell.
\end{aligned}$$

At each layer, the operations performed are linear combinations and multiplications involving $\boldsymbol{X}_\ell^\top \boldsymbol{X}_\ell$ and identity matrices scaled by the parameters (a_ℓ, b_ℓ). Thus, each coefficient e_k of $(\boldsymbol{X}^\top \boldsymbol{X})^k$ depends smoothly on the scalar parameters (a_ℓ, b_ℓ).

From (G.13) and (G.14), we have that

$$\begin{aligned}
f_{\text{att-}L}(\boldsymbol{Z}) &= c b_L \boldsymbol{x}_L^\top \boldsymbol{X}_L^\top \boldsymbol{y}_L \\
&= c b_L \cdot \boldsymbol{x}_{L-1}^\top (\boldsymbol{I} + a_{L-1} \boldsymbol{X}_{L-1}^\top \boldsymbol{X}_{L-1})^2 (\boldsymbol{I} + b_{L-1} \boldsymbol{X}_{L-1}^\top \boldsymbol{X}_{L-1}) \boldsymbol{X}_{L-1}^\top \boldsymbol{y}_{L-1} \\
&= \cdots
\end{aligned} \tag{G.17}$$

That is, in the final $f_{\text{att-}L}(\boldsymbol{Z})$ expression, the coefficients corresponding to different degrees of $(\boldsymbol{X}^\top \boldsymbol{X})^k$ depend on the model parameters $c b_L$ and $(a_\ell, b_\ell)_{\ell=1}^{L-1}$, which together have at

most $2L-1$ degrees of freedom. Let $\boldsymbol{c} = [cb_L\ a_1\ \cdots\ a_{L-1}\ b_1\ \cdots\ b_{L-1}]^\top$. This means there exists a smooth function $g : \mathbb{R}^{2L-1} \to \mathbb{R}^K$ such that: $\boldsymbol{e} = g(\boldsymbol{c})$.

It remains to show that an L-layer linear attention model can produce terms involving powers of $\boldsymbol{X}^\top\boldsymbol{X}$ up to degree $(3^L-3)/2$.

Let $f(\boldsymbol{Z})$ be a function that contains terms of the form $\boldsymbol{x}^\top(\boldsymbol{X}^\top\boldsymbol{X})^k\boldsymbol{X}^\top\boldsymbol{y}$ for various powers k. Define $\mathcal{P}(f(\boldsymbol{Z}))$ as the projection that extracts the highest degree k present in $f(\boldsymbol{Z})$. For example, $\mathcal{P}\big(\boldsymbol{x}^\top(\boldsymbol{I}+(\boldsymbol{X}^\top\boldsymbol{X})^2)\boldsymbol{X}^\top\boldsymbol{y}\big) = 2$. Then from (G.17), we have

$$\begin{aligned}
\mathcal{P}(f_{\text{att-}L}(\boldsymbol{Z})) &= \mathcal{P}(\boldsymbol{x}_L^\top\boldsymbol{X}_L^\top\boldsymbol{y}_L)\\
&= \mathcal{P}(\boldsymbol{x}_{L-1}^\top(\boldsymbol{X}_{L-1}^\top\boldsymbol{X}_{L-1})^3\boldsymbol{X}_{L-1}^\top\boldsymbol{y}_{L-1})\\
&= \mathcal{P}(\boldsymbol{x}_{L-2}^\top(\boldsymbol{X}_{L-2}^\top\boldsymbol{X}_{L-2})(\boldsymbol{X}_{L-2}^\top\boldsymbol{X}_{L-2})^{3^2}(\boldsymbol{X}_{L-2}^\top\boldsymbol{X}_{L-2})^2\boldsymbol{X}_{L-2}^\top\boldsymbol{y}_{L-2})\\
&= \mathcal{P}(\boldsymbol{x}_{L-2}^\top(\boldsymbol{X}_{L-2}^\top\boldsymbol{X}_{L-2})^{3^2+3}\boldsymbol{X}_{L-2}^\top\boldsymbol{y}_{L-2})\\
&= \mathcal{P}(\boldsymbol{x}_{L-3}^\top(\boldsymbol{X}_{L-3}^\top\boldsymbol{X}_{L-3})(\boldsymbol{X}_{L-3}^\top\boldsymbol{X}_{L-3})^{3^3+3^2}(\boldsymbol{X}_{L-3}^\top\boldsymbol{X}_{L-3})^2\boldsymbol{X}_{L-3}^\top\boldsymbol{y}_{L-3})\\
&= \mathcal{P}(\boldsymbol{x}_{L-3}^\top(\boldsymbol{X}_{L-3}^\top\boldsymbol{X}_{L-3})^{3^3+3^2+3}\boldsymbol{X}_{L-3}^\top\boldsymbol{y}_{L-3})\\
&= \ldots\\
&= \mathcal{P}(\boldsymbol{x}^\top(\boldsymbol{X}^\top\boldsymbol{X})^{3^{L-1}+\cdots+3^2+3}\boldsymbol{X}^\top\boldsymbol{y})\\
&= 3^{L-1}+\cdots+3^2+3 = \frac{3^L-3}{2}.
\end{aligned}$$

It completes the proof.

■

G.2.4 Proof of Theorem 8.2

Proof. Let $\boldsymbol{\xi} \sim \mathcal{N}(0, \boldsymbol{I})$ and rewrite $y\boldsymbol{x} = \boldsymbol{\mu}+\sigma\boldsymbol{\xi}$. For any matrix $\boldsymbol{A} \in \mathbb{R}^{d\times d}$, the prediction error of $\hat{y}_{\boldsymbol{A}} = \operatorname{sgn}(\boldsymbol{x}^\top\boldsymbol{A}\hat{\boldsymbol{\mu}}_s)$ given $\hat{\boldsymbol{\mu}}_s$ returns

$$\begin{aligned}
\mathbb{P}(\hat{\boldsymbol{y}}_{\boldsymbol{A}} \neq y \mid \hat{\boldsymbol{\mu}}_s) &= \mathbb{P}(y\boldsymbol{x}^\top\boldsymbol{A}\hat{\boldsymbol{\mu}}_s < 0 \mid \hat{\boldsymbol{\mu}}_s)\\
&= \mathbb{P}((\boldsymbol{\mu}+\sigma\boldsymbol{\xi})^\top\boldsymbol{A}\hat{\boldsymbol{\mu}}_s < 0 \mid \hat{\boldsymbol{\mu}}_s)\\
&= Q\left(\frac{\boldsymbol{\mu}^\top\boldsymbol{A}\hat{\boldsymbol{\mu}}_s}{\sigma\|\boldsymbol{A}\hat{\boldsymbol{\mu}}_s\|}\right). \qquad \text{(G.18)}
\end{aligned}$$

For any $\boldsymbol{A} \in \mathbb{R}^{d\times d}$, we can decompose it as

$$\boldsymbol{A} = \sum_{i=1}^{d}\lambda_i\boldsymbol{u}_i\boldsymbol{v}_i^\top$$

where $\boldsymbol{u}_1 = \boldsymbol{\mu}$, $\|\boldsymbol{u}_i\| = 1$ and $\boldsymbol{u}_i^\top \boldsymbol{u}_j = 0$ for any $i \neq j$. Let $\lambda_1 > 0$. Then, we get

$$\begin{aligned}
\boldsymbol{\mu}^\top \boldsymbol{A}\hat{\boldsymbol{\mu}}_s &= \boldsymbol{\mu}^\top (\sum_{i=1}^{d} \lambda_i \boldsymbol{u}_i \boldsymbol{v}_i^\top)\hat{\boldsymbol{\mu}}_s \\
&= \sum_{i=1}^{d} \lambda_i \boldsymbol{\mu}^\top \boldsymbol{u}_i \boldsymbol{v}_i^\top \hat{\boldsymbol{\mu}}_s \\
&= \lambda_1 \boldsymbol{\mu}^\top \boldsymbol{u}_1 \boldsymbol{v}_1^\top \hat{\boldsymbol{\mu}}_s \\
&= \lambda_1 \boldsymbol{v}_1^\top \hat{\boldsymbol{\mu}}_s. \qquad \text{(G.19)}
\end{aligned}$$

Now consider $\|\boldsymbol{A}\hat{\boldsymbol{\mu}}_s\|$ where we have

$$\begin{aligned}
\boldsymbol{A}\hat{\boldsymbol{\mu}}_s &= \sum_{i=1}^{d} \lambda_i \boldsymbol{u}_i \boldsymbol{v}_i^\top \hat{\boldsymbol{\mu}}_s \\
&= \lambda_1 \boldsymbol{\mu} \boldsymbol{v}_1^\top \hat{\boldsymbol{\mu}}_s + \sum_{i=2}^{d} \lambda_i \boldsymbol{u}_i \boldsymbol{v}_i^\top \hat{\boldsymbol{\mu}}_s.
\end{aligned}$$

Since $\boldsymbol{u}_i$, $i \neq 1$ is orthogonal to $\boldsymbol{\mu}$, $\lambda_1 \boldsymbol{\mu} \boldsymbol{v}_1^\top \hat{\boldsymbol{\mu}}_s$ is orthogonal to $\sum_{i=2}^{d} \lambda_i \boldsymbol{u}_i \boldsymbol{v}_i^\top \hat{\boldsymbol{\mu}}_s$. Therefore, given $\|\boldsymbol{u}_i\| = 1$ for all $i \in [d]$, it obeys

$$\|\boldsymbol{A}\hat{\boldsymbol{\mu}}_s\|^2 = \|\lambda_1 \boldsymbol{\mu} \boldsymbol{v}_1^\top \hat{\boldsymbol{\mu}}_s\|^2 + \sum_{i=2}^{d} \|\lambda_i \boldsymbol{u}_i \boldsymbol{v}_i^\top \hat{\boldsymbol{\mu}}_s\|^2 = (\lambda_1 \boldsymbol{v}_1^\top \hat{\boldsymbol{\mu}}_s)^2 + \lambda_1^2 \sum_{i=2}^{d} (\lambda_1^{-1} \lambda_i \boldsymbol{v}_i^\top \hat{\boldsymbol{\mu}}_s)^2. \qquad \text{(G.20)}$$

For simplicity, define

$$\Delta(\hat{\boldsymbol{\mu}}_s) = \sum_{i=2}^{d} (\lambda_1^{-1} \lambda_i \boldsymbol{v}_i^\top \hat{\boldsymbol{\mu}}_s)^2$$

where $\Delta(\cdot)$ is a function of λ_1 and $(\lambda_i, \boldsymbol{v}_i)$'s for $i \geq 2$, and we have

$$\Delta(\hat{\boldsymbol{\mu}}_s) \geq 0 \quad \text{and} \quad \Delta(-\hat{\boldsymbol{\mu}}_s) = \Delta(\hat{\boldsymbol{\mu}}_s).$$

Recall that $\hat{\boldsymbol{\mu}}_s$ is the SPI estimator (cf. (SPI)). Let $|\mathcal{I}| = m$. We can write $\hat{\boldsymbol{\mu}}_s = \boldsymbol{\mu} + \boldsymbol{\xi}'/\sqrt{m}$ where $\boldsymbol{\xi}' \sim \mathcal{N}(0, \sigma^2 \boldsymbol{I})$.

Using (G.18), (G.19) and (G.20), the classification error becomes

$$\begin{aligned}
\mathbb{P}(\hat{\boldsymbol{y}}_{\boldsymbol{A}} \neq y) &= \mathbb{E}_{\hat{\boldsymbol{\mu}}_s}\left[Q\left(\frac{\boldsymbol{\mu}^\top \boldsymbol{A}\hat{\boldsymbol{\mu}}_s}{\sigma\|\boldsymbol{A}\hat{\boldsymbol{\mu}}_s\|}\right)\right] \\
&= \mathbb{E}_{\hat{\boldsymbol{\mu}}_s}\left[Q\left(\frac{\boldsymbol{v}_1^\top \hat{\boldsymbol{\mu}}_s}{\sigma\sqrt{(\boldsymbol{v}_1^\top \hat{\boldsymbol{\mu}}_s)^2 + \Delta(\hat{\boldsymbol{\mu}}_s)}}\right)\right] \\
&= \mathbb{E}_{\boldsymbol{v}_1^\top \hat{\boldsymbol{\mu}}_s<0}\left[Q\left(\frac{\boldsymbol{v}_1^\top \hat{\boldsymbol{\mu}}_s}{\sigma\sqrt{(\boldsymbol{v}_1^\top \hat{\boldsymbol{\mu}}_s)^2 + \Delta(\hat{\boldsymbol{\mu}}_s)}}\right)\right] + \mathbb{E}_{\boldsymbol{v}_1^\top \hat{\boldsymbol{\mu}}_s\geq 0}\left[Q\left(\frac{\boldsymbol{v}_1^\top \hat{\boldsymbol{\mu}}_s}{\sigma\sqrt{(\boldsymbol{v}_1^\top \hat{\boldsymbol{\mu}}_s)^2 + \Delta(\hat{\boldsymbol{\mu}}_s)}}\right)\right].
\end{aligned}$$

First, note that for any $x > 0$, $Q(x) < 0.5 < Q(-x)$. Therefore, the optimal choice of $\boldsymbol{v}_1 \in \mathbb{R}^d$ that minimizes $\mathbb{P}(\hat{y}_{\boldsymbol{A}} \neq y)$ is contained within the set of $\boldsymbol{v}_1$ values that maximize $\mathbb{P}(\boldsymbol{v}_1^\top \hat{\boldsymbol{\mu}}_s > 0)$. Let $\boldsymbol{v}_1^\star := \arg\max_{\boldsymbol{v}_1 \in \mathbb{R}^d} \mathbb{P}(\boldsymbol{v}_1^\top \hat{\boldsymbol{\mu}}_s > 0)$. Given that $\hat{\boldsymbol{\mu}}_s \sim \mathcal{N}(\boldsymbol{\mu}, \sigma^2/m\boldsymbol{I})$, we have that $\boldsymbol{v}_1^\star = c\boldsymbol{\mu}$ for $c > 0$. Let $c = 1$ and therefore, $\boldsymbol{v}_1^\star = \boldsymbol{\mu}$ without loss of generality (since λ_1 can be any positive scalar). Then we obtain

$$\min_{\boldsymbol{A}\in\mathbb{R}^{d\times d}} \mathbb{P}(\hat{\boldsymbol{y}}_{\boldsymbol{A}} \neq y) = \min_{\Delta} \mathbb{E}_{\hat{\boldsymbol{\mu}}_s}\left[Q\left(\frac{\boldsymbol{\mu}^\top \hat{\boldsymbol{\mu}}_s}{\sigma\sqrt{(\boldsymbol{\mu}^\top \hat{\boldsymbol{\mu}}_s)^2 + \Delta(\hat{\boldsymbol{\mu}}_s)}}\right)\right].$$

Let $f(\hat{\boldsymbol{\mu}}_s)$ be the probability density function of $\hat{\boldsymbol{\mu}}_s$. Since $\hat{\boldsymbol{\mu}}_s \sim \mathcal{N}(\boldsymbol{\mu}, \sigma^2/m\boldsymbol{I})$, then it satisfies

$$f(\hat{\boldsymbol{\mu}}_s) \geq f(-\hat{\boldsymbol{\mu}}_s) \quad \text{for any } \boldsymbol{\mu}^\top \hat{\boldsymbol{\mu}}_s > 0. \tag{G.21}$$

Therefore, the classification error becomes

$$\begin{aligned}
&\mathbb{P}(\hat{\boldsymbol{y}}_{\boldsymbol{A}} \neq y \mid \boldsymbol{v}_1 = \boldsymbol{\mu}) \\
&= \int_{\hat{\boldsymbol{\mu}}_s} f(\hat{\boldsymbol{\mu}}_s) Q\left(\frac{\boldsymbol{\mu}^\top \hat{\boldsymbol{\mu}}_s}{\sigma\sqrt{(\boldsymbol{\mu}^\top \hat{\boldsymbol{\mu}}_s)^2 + \Delta(\hat{\boldsymbol{\mu}}_s)}}\right) d\hat{\boldsymbol{\mu}}_s \\
&= \int_{\boldsymbol{\mu}^\top \hat{\boldsymbol{\mu}}_s>0} f(\hat{\boldsymbol{\mu}}_s) Q\left(\frac{\boldsymbol{\mu}^\top \hat{\boldsymbol{\mu}}_s}{\sigma\sqrt{(\boldsymbol{\mu}^\top \hat{\boldsymbol{\mu}}_s)^2 + \Delta(\hat{\boldsymbol{\mu}}_s)}}\right) + f(-\hat{\boldsymbol{\mu}}_s) Q\left(\frac{-\boldsymbol{\mu}^\top \hat{\boldsymbol{\mu}}_s}{\sigma\sqrt{(\boldsymbol{\mu}^\top \hat{\boldsymbol{\mu}}_s)^2 + \Delta(\hat{\boldsymbol{\mu}}_s)}}\right) d\hat{\boldsymbol{\mu}}_s \\
&= \int_{\boldsymbol{\mu}^\top \hat{\boldsymbol{\mu}}_s>0} \left(f(\hat{\boldsymbol{\mu}}_s) - f(-\hat{\boldsymbol{\mu}}_s)\right) Q\left(\frac{\boldsymbol{\mu}^\top \hat{\boldsymbol{\mu}}_s}{\sigma\sqrt{(\boldsymbol{\mu}^\top \hat{\boldsymbol{\mu}}_s)^2 + \Delta(\hat{\boldsymbol{\mu}}_s)}}\right) + f(-\hat{\boldsymbol{\mu}}_s) d\hat{\boldsymbol{\mu}}_s.
\end{aligned}$$

Following (G.21), to minimize the error, we need minimize $Q\left(\frac{\boldsymbol{\mu}^\top \hat{\boldsymbol{\mu}}_s}{\sigma\sqrt{(\boldsymbol{\mu}^\top \hat{\boldsymbol{\mu}}_s)^2 + \Delta(\hat{\boldsymbol{\mu}}_s)}}\right)$ for $\boldsymbol{\mu}^\top \hat{\boldsymbol{\mu}}_s > 0$, which can be easily done by choosing $\lambda_i = 0$ for $i \geq 2$. Then we get $\Delta(\hat{\boldsymbol{\mu}}_s) \equiv 0$. Therefore,

the optimal solution set $\mathcal{A}^\star$ defined in Theorem 8.2 satisfies:

$$\mathcal{A}^\star = \left\{ \lambda_1 \boldsymbol{\mu}\boldsymbol{\mu}^\top \;\middle|\; \lambda_1 > 0 \right\}.$$

Combining all together, we obtain

$$\begin{aligned}
\min_{\boldsymbol{A}\in\mathbb{R}^{d\times d}} \mathbb{P}(\hat{\boldsymbol{y}}_{\boldsymbol{A}} \neq y) &= \int_{\boldsymbol{\mu}^\top \hat{\boldsymbol{\mu}}_s > 0} \left(f(\hat{\boldsymbol{\mu}}_s) - f(-\hat{\boldsymbol{\mu}})\right) Q\left(\frac{1}{\sigma}\right) + f(-\hat{\boldsymbol{\mu}}_s) d\hat{\boldsymbol{\mu}}_s \\
&= \int_{\boldsymbol{\mu}^\top \hat{\boldsymbol{\mu}}_s > 0} f(\hat{\boldsymbol{\mu}}_s) d\hat{\boldsymbol{\mu}}_s \cdot Q\left(\frac{1}{\sigma}\right) + \int_{\boldsymbol{\mu}^\top \hat{\boldsymbol{\mu}}_s < 0} f(\hat{\boldsymbol{\mu}}_s) d\hat{\boldsymbol{\mu}}_s \cdot \left(1 - Q\left(\frac{1}{\sigma}\right)\right) \\
&= Q\left(-\frac{\sqrt{m}}{\sigma}\right) Q\left(\frac{1}{\sigma}\right) + Q\left(\frac{\sqrt{m}}{\sigma}\right)\left(1 - Q\left(\frac{1}{\sigma}\right)\right) \\
&= \left(1 - Q\left(\frac{\sqrt{m}}{\sigma}\right)\right) Q\left(\frac{1}{\sigma}\right) + Q\left(\frac{\sqrt{m}}{\sigma}\right)\left(1 - Q\left(\frac{1}{\sigma}\right)\right) \\
&= Q\left(\frac{1}{\sigma}\right) + Q\left(\frac{\sqrt{m}}{\sigma}\right) - 2Q\left(\frac{\sqrt{m}}{\sigma}\right) Q\left(\frac{1}{\sigma}\right).
\end{aligned}$$

It completes the proof. ∎

G.2.5 Proof of Theorem 8.4

Proof. Recap from Proposition 8.1. For any L-layer attention model with $L \geq 2$, it can output

$$f_{\text{att-}L}(\boldsymbol{Z}) = \boldsymbol{x}^\top (\boldsymbol{X}^\top \boldsymbol{X}/n - \sigma^2 \boldsymbol{I}) \hat{\boldsymbol{\mu}}_s. \tag{G.22}$$

Let

$$\hat{y} = \operatorname{sgn}(f_{\text{att-}L}(\boldsymbol{Z}))$$

with $f_{\text{att-}L}(\boldsymbol{Z})$ defined in (G.22). Then we have

$$\mathbb{P}(y^\star_{\text{att-}L}(\boldsymbol{Z}) \neq y) \leq \mathbb{P}(\hat{y} \neq y).$$

Therefore, in the following, we focus on upper-bounding the classification error $\mathbb{P}(\hat{y} \neq y)$ corresponding to (G.22). Given that the optimal prediction under the form $\operatorname{sgn}(\boldsymbol{x}^\top \boldsymbol{A} \hat{\boldsymbol{\mu}}_s)$ is given by $\hat{y}_{\boldsymbol{\mu}\boldsymbol{\mu}^\top} := \operatorname{sgn}(\boldsymbol{x}^\top \boldsymbol{\mu}\boldsymbol{\mu}^\top \hat{\boldsymbol{\mu}}_s)$ (cf. Theorem 8.2), with its corresponding error presented in (8.13). To analyze the performance of $\hat{y}$, we study its difference from the prediction $\hat{y}_{\boldsymbol{\mu}\boldsymbol{\mu}^\top}$.

To begin with, let $\boldsymbol{g}_i = \boldsymbol{\xi}_i/\sigma \sim \mathcal{N}(0, \boldsymbol{I})$ and $\boldsymbol{g} = \sum_{i=1}^n \boldsymbol{\xi}_i/\sigma\sqrt{n} \sim \mathcal{N}(0, \boldsymbol{I})$. For simplicity,

let $\boldsymbol{A} := \boldsymbol{X}^\top \boldsymbol{X}/n - \sigma^2 \boldsymbol{I}$. We get

$$\begin{aligned}\boldsymbol{A} &= \frac{1}{n}\boldsymbol{X}^\top \boldsymbol{X} - \sigma^2 \boldsymbol{I} \\ &= \frac{1}{n}\left(\sum_{i=1}^n \boldsymbol{\mu}\boldsymbol{\mu}^\top + \boldsymbol{\mu}\boldsymbol{\xi}_i^\top + \boldsymbol{\xi}_i\boldsymbol{\mu}^\top + \boldsymbol{\xi}_i\boldsymbol{\xi}_i^\top\right) - \sigma^2 \boldsymbol{I} \\ &= \boldsymbol{\mu}\boldsymbol{\mu}^\top + \frac{\sigma}{\sqrt{n}}(\boldsymbol{\mu}\boldsymbol{g}^\top + \boldsymbol{g}\boldsymbol{\mu}^\top) + \sigma^2\left(\frac{\sum_{i=1}^n \boldsymbol{g}_i\boldsymbol{g}_i^\top}{n} - \boldsymbol{I}\right).\end{aligned}$$

Recall (G.18) from the proof of Theorem 8.2. Our goal is to bound

$$\mathbb{P}(\hat{y} \neq y) = \mathbb{E}_{\hat{\boldsymbol{\mu}}}\left[Q\left(\frac{\boldsymbol{\mu}^\top \boldsymbol{A}\hat{\boldsymbol{\mu}}_s}{\sigma\|\boldsymbol{A}\hat{\boldsymbol{\mu}}_s\|}\right)\right].$$

Define

$$\boldsymbol{\Delta} := \boldsymbol{A} - \boldsymbol{\mu}\boldsymbol{\mu}^\top = \frac{\sigma}{\sqrt{n}}(\boldsymbol{\mu}\boldsymbol{g}^\top + \boldsymbol{g}\boldsymbol{\mu}^\top) + \sigma^2\left(\frac{\sum_{i=1}^n \boldsymbol{g}_i\boldsymbol{g}_i^\top}{n} - \boldsymbol{I}\right). \tag{G.23}$$

From the Laurent-Massart inequality [101], we have that with probability at least $1 - e^{-t_1}$ (assuming $t_1 \geq d$), the first term of (G.23) can be bounded by

$$\frac{1}{\sqrt{n}}\left\|\boldsymbol{\mu}\boldsymbol{g}^\top + \boldsymbol{g}\boldsymbol{\mu}^\top\right\| \leq \frac{2\|\boldsymbol{g}\|}{\sqrt{n}} \leq 6\sqrt{\frac{t_1}{n}}. \tag{G.24}$$

Additionally, from [141], we have that with probability at least $1 - e^{-t_2}$ (assuming $t_2 \geq d$), the second term of $\boldsymbol{\Delta}$ (cf. (G.23)) is bounded by (with a universal constant $C > 0$)

$$\left\|\frac{\sum_{i=1}^n \boldsymbol{g}_i\boldsymbol{g}_i^\top}{n} - \boldsymbol{I}\right\| \leq C \cdot \sqrt{\frac{t_2}{n}}. \tag{G.25}$$

Combining (G.24) and (G.25), we get with probability at least $1 - 2e^{-t}$ (for $t \geq d$)

$$\|\boldsymbol{\Delta}\| \leq C_1\sqrt{\frac{t}{n}} \quad \text{where} \quad C_1 := 6\sigma + C\sigma^2.$$

We also bound $\|\hat{\boldsymbol{\mu}}_s\|$ as follows. Let $\hat{\boldsymbol{\mu}}_s = \boldsymbol{\mu} + \sigma/\sqrt{m}\boldsymbol{g}' \sim \mathcal{N}(\boldsymbol{\mu}, \sigma^2 m\boldsymbol{I})$, similar to (G.24), with probability at least $1 - e^{-t_3}$ (assuming $2d \leq t_3 \leq m/4\sigma^2$), we can bound

$$\|\hat{\boldsymbol{\mu}}_s\| \leq 1 + \frac{\sigma}{\sqrt{m}}\|\boldsymbol{g}'\| \leq 1 + 3\sigma\sqrt{\frac{t_3}{m}} \leq 3.$$

Then consider a significantly large n (to ensure that $\|\boldsymbol{\Delta}\| \leq 1/12$, e.g., $n \geq (12C_1)^2 t$). With

probability at least $1 - 3e^{-\min(t,t_3)}$ and suppose that $\boldsymbol{\mu}^\top \hat{\boldsymbol{\mu}}_s > 0.5$, we can bound

$$\begin{aligned}
\left| \frac{\boldsymbol{\mu}^\top \boldsymbol{A}\hat{\boldsymbol{\mu}}_s}{\|\boldsymbol{A}\hat{\boldsymbol{\mu}}_s\|} - \frac{\boldsymbol{\mu}^\top \boldsymbol{\mu}\boldsymbol{\mu}^\top \hat{\boldsymbol{\mu}}_s}{\|\boldsymbol{\mu}\boldsymbol{\mu}^\top \hat{\boldsymbol{\mu}}_s\|} \right| &= \left| \frac{\boldsymbol{\mu}^\top (\boldsymbol{\Delta} + \boldsymbol{\mu}\boldsymbol{\mu}^\top)\hat{\boldsymbol{\mu}}_s}{\|(\boldsymbol{\Delta} + \boldsymbol{\mu}\boldsymbol{\mu}^\top)\hat{\boldsymbol{\mu}}_s\|} - \frac{\boldsymbol{\mu}^\top \boldsymbol{\mu}\boldsymbol{\mu}^\top \hat{\boldsymbol{\mu}}_s}{\|\boldsymbol{\mu}\boldsymbol{\mu}^\top \hat{\boldsymbol{\mu}}_s\|} \right| \\
&\leq \left| \frac{\boldsymbol{\mu}^\top \boldsymbol{\Delta}\hat{\boldsymbol{\mu}}_s}{\min(\|(\boldsymbol{\Delta} + \boldsymbol{\mu}\boldsymbol{\mu}^\top)\hat{\boldsymbol{\mu}}_s\|, \|\boldsymbol{\mu}\boldsymbol{\mu}^\top \hat{\boldsymbol{\mu}}_s\|)} \right| \\
&\leq \frac{\|\boldsymbol{\Delta}\| \cdot \|\hat{\boldsymbol{\mu}}_s\|}{\boldsymbol{\mu}^\top \hat{\boldsymbol{\mu}}_s - \|\boldsymbol{\Delta}\| \cdot \|\hat{\boldsymbol{\mu}}_s\|} \\
&\leq 4\|\boldsymbol{\Delta}\| \cdot \|\hat{\boldsymbol{\mu}}_s\| \\
&\leq C_2 \sqrt{\frac{t}{n}} \quad \text{where} \quad C_2 := 12C_1.
\end{aligned}$$

Now, we are ready to bound the classification error, where we get

$$\begin{aligned}
\mathbb{P}(\hat{y} \neq y) &= \mathbb{E}_{\hat{\boldsymbol{\mu}}} \left[Q\left(\frac{\boldsymbol{\mu}^\top \boldsymbol{A}\hat{\boldsymbol{\mu}}_s}{\sigma \|\boldsymbol{A}\hat{\boldsymbol{\mu}}_s\|} \right) \right] \\
&= \mathbb{E}_{\hat{\boldsymbol{\mu}}} \left[Q\left(\frac{1}{\sigma} + \frac{1}{\sigma} \left(\frac{\boldsymbol{\mu}^\top \boldsymbol{A}\hat{\boldsymbol{\mu}}_s}{\|\boldsymbol{A}\hat{\boldsymbol{\mu}}_s\|} - \frac{\boldsymbol{\mu}^\top \boldsymbol{\mu}\boldsymbol{\mu}^\top \hat{\boldsymbol{\mu}}_s}{\|\boldsymbol{\mu}\boldsymbol{\mu}^\top \hat{\boldsymbol{\mu}}_s\|} \right) \right) \right] \\
&\leq \mathbb{P}(\boldsymbol{\mu}^\top \hat{\boldsymbol{\mu}}_s > 0.5) \left(Q\left(\frac{1 - C_2\sqrt{t/n}}{\sigma} \right) + 3e^{-\min(t,t_3)} \right) + \mathbb{P}(\boldsymbol{\mu}^\top \hat{\boldsymbol{\mu}}_s < 0.5) \\
&\leq Q\left(\frac{1 - C_2\sqrt{t/n}}{\sigma} \right) + 3e^{-\min(t,t_3)} + Q\left(\frac{\sqrt{m}}{2\sigma} \right).
\end{aligned}$$

Choosing $t = t_3 = 2d$, since $m/4\sigma^2 \geq 2d$, we obtain

$$\begin{aligned}
\mathbb{P}(\hat{y} \neq y) &\leq Q\left(\frac{1 - C_2\sqrt{2d/n}}{\sigma} \right) + 3e^{-2d} + 0.5e^{-d} \\
&\leq Q\left(\frac{1 - C_2\sqrt{2d/n}}{\sigma} \right) + e^{-d}.
\end{aligned}$$

It completes the proof. ■

APPENDIX H

Appendix for Chapter 9

H.1 Stability of Transformer-based ICL

Lemma H.1 *Let $\boldsymbol{x}, \boldsymbol{\epsilon} \in \mathbb{R}^n$ be vectors obeying $\|\boldsymbol{x}\|_{\ell_\infty}, \|\boldsymbol{x}+\boldsymbol{\epsilon}\|_{\ell_\infty} \leq c$. Then, there exists a constant $C = C(c)$, such that*

$$\|\mathbb{S}(\boldsymbol{x})\|_{\ell_\infty} \leq e^{2c}/n \quad \textit{and} \quad \|\mathbb{S}(\boldsymbol{x}) - \mathbb{S}(\boldsymbol{x}+\boldsymbol{\epsilon})\|_{\ell_1} \leq e^{2c}\|\boldsymbol{\epsilon}\|_{\ell_1}/n.$$

Proof. Without losing generality, assume the first coordinate is the largest. Using monotonicity of softmax, we obtain $\|\mathbb{S}(\boldsymbol{x})\|_{\ell_\infty} \leq \frac{e^c}{e^c + \sum_{i=2}^n e^{-c}} \leq \frac{e^{2c}}{n}$. Say $\|\boldsymbol{\epsilon}\| = \epsilon$. As $\boldsymbol{\epsilon} \to 0$, we have

$$\lim_{\epsilon \to 0} [\mathbb{S}(\boldsymbol{x}+\boldsymbol{\epsilon}) - \mathbb{S}(\boldsymbol{x})]/\epsilon = [\mathrm{diag}(\mathbb{S}(\boldsymbol{x})) - \mathbb{S}(\boldsymbol{x})\mathbb{S}(\boldsymbol{x})^\top]\boldsymbol{\epsilon}/\epsilon$$

$\|[\mathrm{diag}(\mathbb{S}(\boldsymbol{x})) - \mathbb{S}(\boldsymbol{x})\mathbb{S}(\boldsymbol{x})^\top]\boldsymbol{\epsilon}\|_{\ell_1} \leq e^{2c}\|\boldsymbol{\epsilon}\|_{\ell_1}/n$. Integrating the derivative along $\boldsymbol{\epsilon}$, we obtain the result. ■

For a matrix $\boldsymbol{A}$, let $\|\boldsymbol{A}\|_{2,p}$ denote the ℓ_p norm of the vector obtained by the ℓ_2 norms of its rows.

Lemma H.2 *Let $\boldsymbol{X} = [\boldsymbol{x}_1 \ \ldots \ \boldsymbol{x}_n]^\top$ and $\boldsymbol{E} = [\boldsymbol{\epsilon}_1 \ \ldots \ \boldsymbol{\epsilon}_n]^\top$ be the input and perturbation matrices respectively. Assume that the tokens $(\boldsymbol{x}_i, \boldsymbol{x}_i + \boldsymbol{\epsilon}_i)$ lie in unit ball i.e. $\|\boldsymbol{X}\|_{2,\infty}, \|\boldsymbol{X}+\boldsymbol{E}\|_{2,\infty} \leq 1$. Let $\boldsymbol{W}_v, \boldsymbol{W} \in \mathbb{R}^{d\times d}$ be the weights of the self-attention layer obeying $\|\boldsymbol{W}_v\| \leq 1$ and $\|\boldsymbol{W}\| \leq \Gamma$. Define the attention outputs $\boldsymbol{A} = \mathbb{S}(\boldsymbol{X}\boldsymbol{W}\boldsymbol{X}^\top)\boldsymbol{X}\boldsymbol{W}_v$ and $\bar{\boldsymbol{A}} = \mathbb{S}(\bar{\boldsymbol{X}}\boldsymbol{W}\bar{\boldsymbol{X}}^\top)\bar{\boldsymbol{X}}\boldsymbol{W}_v$. We have that*

$$\|\boldsymbol{A}\|_{2,\infty}, \|\bar{\boldsymbol{A}}\|_{2,\infty} \leq 1, \quad \|\boldsymbol{A} - \bar{\boldsymbol{A}}\|_{2,1} \leq (2\Gamma+1)e^{2\Gamma}\|\boldsymbol{E}\|_{2,1}.$$

Proof. First observe that $\boldsymbol{W}_v$ preserves norms i.e. $\boldsymbol{X}\boldsymbol{W}_v$ obeys $\|\boldsymbol{X}\boldsymbol{W}_v\|_{2,\infty} \leq \|\boldsymbol{X}\|_{2,\infty} \leq 1$ and $\|\boldsymbol{E}\boldsymbol{W}_v\|_{2,1} \leq \|\boldsymbol{E}\|_{2,1}$.

Next, set $\bar{\boldsymbol{X}} = \boldsymbol{X} + \boldsymbol{E}$ and define attention outputs $\boldsymbol{A} = \mathbb{S}(\boldsymbol{X}\boldsymbol{W}\boldsymbol{X}^\top)\boldsymbol{X}\boldsymbol{W}_v$, $\bar{\boldsymbol{A}} = \mathbb{S}(\bar{\boldsymbol{X}}\boldsymbol{W}\bar{\boldsymbol{X}}^\top)\bar{\boldsymbol{X}}\boldsymbol{W}_v$. Observe that, since softmax applies row-wise to the similarities (e.g. $\boldsymbol{X}\boldsymbol{W}\boldsymbol{X}^\top$), we preserve the feature norms i.e. $\|\boldsymbol{A}\|_{2,\infty}, \|\bar{\boldsymbol{A}}\|_{2,\infty} \leq 1$ as advertised.

Now, consider the attention output difference $\boldsymbol{P} = \bar{\boldsymbol{A}} - \boldsymbol{A}$

$$\boldsymbol{P} = [\mathbb{S}(\bar{\boldsymbol{X}}\boldsymbol{W}\bar{\boldsymbol{X}}^\top) - \mathbb{S}(\boldsymbol{X}\boldsymbol{W}\boldsymbol{X}^\top)]\boldsymbol{X}\boldsymbol{W}_v + \mathbb{S}(\bar{\boldsymbol{X}}\boldsymbol{W}\bar{\boldsymbol{X}}^\top)\boldsymbol{E}\boldsymbol{W}_v.$$

For any pairs of tokens, we have $|\boldsymbol{x}_i^\top \boldsymbol{W}\boldsymbol{x}_j| \leq \Gamma$. Using Lemma H.1

$$\|\boldsymbol{P}_2\|_{2,1} = \|\mathbb{S}(\bar{\boldsymbol{X}}\boldsymbol{W}\bar{\boldsymbol{X}}^\top)\boldsymbol{E}\boldsymbol{W}_v\|_{2,1} \leq n\|\mathbb{S}(\bar{\boldsymbol{X}}\boldsymbol{W}\bar{\boldsymbol{X}}^\top)\|_\infty \|\boldsymbol{E}\|_{2,1} \leq e^{2\Gamma}\|\boldsymbol{E}\|_{2,1}.$$

Secondly, set $\boldsymbol{P}_1 = [\mathbb{S}(\bar{\boldsymbol{X}}\boldsymbol{W}\bar{\boldsymbol{X}}^\top) - \mathbb{S}(\boldsymbol{X}\boldsymbol{W}\boldsymbol{X}^\top)]\boldsymbol{X}\boldsymbol{W}_v$. We have that

$$\begin{aligned}\|\boldsymbol{P}_1\|_{2,1} &\leq \|\mathbb{S}(\bar{\boldsymbol{X}}\boldsymbol{W}\bar{\boldsymbol{X}}^\top) - \mathbb{S}(\boldsymbol{X}\boldsymbol{W}\boldsymbol{X}^\top)\|_{\ell_1}\|\boldsymbol{X}\boldsymbol{W}_v\|_{2,\infty} \\ &\leq \|\mathbb{S}(\bar{\boldsymbol{X}}\boldsymbol{W}\bar{\boldsymbol{X}}^\top) - \mathbb{S}(\boldsymbol{X}\boldsymbol{W}\boldsymbol{X}^\top)\|_{\ell_1}.\end{aligned}$$

To proceed, let us control the derivative by letting the perturbation $\boldsymbol{E} \to 0$ and then integrate the derivative along $\boldsymbol{E}$. Clearly, as $\boldsymbol{E} \to 0$, the quadratic-terms involving $\boldsymbol{E}$ disappear and $\|\mathbb{S}(\bar{\boldsymbol{X}}\boldsymbol{W}\bar{\boldsymbol{X}}^\top) - \mathbb{S}(\boldsymbol{X}\boldsymbol{W}\boldsymbol{X}^\top)\|_{\ell_1}$

$$\leq \|\mathbb{S}(\bar{\boldsymbol{X}}\boldsymbol{W}\boldsymbol{X}^\top) - \mathbb{S}(\boldsymbol{X}\boldsymbol{W}\boldsymbol{X}^\top)\|_{\ell_1} + \|\mathbb{S}(\boldsymbol{X}\boldsymbol{W}\bar{\boldsymbol{X}}^\top) - \mathbb{S}(\boldsymbol{X}\boldsymbol{W}\boldsymbol{X}^\top)\|_{\ell_1}.$$

To bound the latter, consider each row individually, namely pick a row from $\boldsymbol{X}, \boldsymbol{X}+\boldsymbol{E}$ each denoted by the pair $(\boldsymbol{x}, \boldsymbol{x}+\boldsymbol{\epsilon})$. Note that for any cross-product, we are guaranteed to have $|(\boldsymbol{x}+\boldsymbol{\epsilon})^\top \boldsymbol{W}\boldsymbol{x}_i|, |\boldsymbol{x}^\top \boldsymbol{W}\boldsymbol{x}_i| \leq \Gamma, \|\boldsymbol{\epsilon}^\top \boldsymbol{W}\boldsymbol{X}\|_{\ell_1} \leq \Gamma n\|\boldsymbol{\epsilon}\|$, $\|\boldsymbol{x}^\top \boldsymbol{W}\boldsymbol{E}^\top\|_{\ell_1} \leq \Gamma\|\boldsymbol{E}\|_{2,1}$. Applying perturbation bound of Lemma H.1, we get

$$\|\mathbb{S}((\boldsymbol{x}+\boldsymbol{\epsilon})^\top \boldsymbol{W}\boldsymbol{X}^\top) - \mathbb{S}(\boldsymbol{x}^\top \boldsymbol{W}\boldsymbol{X}^\top)\|_{\ell_1} \leq \Gamma e^{2\Gamma}\|\boldsymbol{\epsilon}\|_{\ell_1} \tag{H.1}$$

$$\|\mathbb{S}(\boldsymbol{x}^\top \boldsymbol{W}(\boldsymbol{X}+\boldsymbol{E})^\top) - \mathbb{S}(\boldsymbol{x}^\top \boldsymbol{W}\boldsymbol{X}^\top)\|_{\ell_1} \leq \Gamma e^{2\Gamma}\|\boldsymbol{E}\|_{2,1}/n. \tag{H.2}$$

Adding up all n rows, we obtain (denoting $\epsilon = \|\boldsymbol{E}\|$)

$$\lim_{\epsilon \to 0} \|\mathbb{S}(\bar{\boldsymbol{X}}\boldsymbol{W}\bar{\boldsymbol{X}}^\top) - \mathbb{S}(\boldsymbol{X}\boldsymbol{W}\boldsymbol{X}^\top)\|_{\ell_1}/\epsilon \leq 2\Gamma e^{2\Gamma}\|\boldsymbol{E}\|_{2,1}/\epsilon.$$

Integrating the derivative along $\boldsymbol{E}$, we obtain $\|\boldsymbol{P}_1\|_{2,1} \leq 2\Gamma e^{2\Gamma}\|\boldsymbol{E}\|_{2,1}$. Cumulatively, we obtain the main claim $\|\boldsymbol{P}\|_{2,1} \leq (2\Gamma+1)e^{2\Gamma}\|\boldsymbol{E}\|_{2,1}$. ∎

Lemma H.3 (Single-layer transformer stability) *Consider the setup of Lemma H.2.*

Let ϕ be a 1-Lipschitz activation function with $\phi(0) = 0$ (e.g. ReLU or Identity). Let $(\boldsymbol{M}_i)_{i=1}^n \in \mathbb{R}^{d\times d}$ be weights of the parallel MLPs following self-attention. Suppose $\|\boldsymbol{M}_i\| \le 1$ and denote the MLP outputs associated to $\boldsymbol{A}, \bar{\boldsymbol{A}}$ by $\boldsymbol{B}, \boldsymbol{B}$. We have that

$$\|\boldsymbol{B}\|_{2,\infty}, \|\boldsymbol{B}\|_{2,\infty} \le 1, \quad \|\boldsymbol{B} - \boldsymbol{B}\|_{2,1} \le (2\Gamma + 1)e^{2\Gamma}\|\boldsymbol{E}\|_{2,1}.$$

Proof. First note that each row of $\boldsymbol{B}$ is given by $\boldsymbol{b}_i = \phi(\boldsymbol{M}_i\boldsymbol{a}_i)$ thus $\|\boldsymbol{b}_i\| \le \|\phi(\boldsymbol{M}_i\boldsymbol{a}_i)\| \le \|\boldsymbol{M}_i\boldsymbol{a}_i\| \le \|\boldsymbol{a}_i\| \le 1$. Secondly, we can write $\|\boldsymbol{b}_i - \bar{\boldsymbol{b}}_i\| \le \|\phi(\boldsymbol{M}_i\boldsymbol{a}_i) - \phi(\boldsymbol{M}_i\bar{\boldsymbol{a}}_i)\| \le \|\boldsymbol{M}_i(\boldsymbol{a}_i - \bar{\boldsymbol{a}}_i)\| \le \|\boldsymbol{a}_i - \bar{\boldsymbol{a}}_i\|$. Thus, $\|\boldsymbol{B} - \boldsymbol{B}\|_{2,1} \le \|\boldsymbol{A} - \bar{\boldsymbol{A}}\|_{2,1}$ and we conclude via Lemma H.2. ∎

Theorem H.1 *Consider a transformer composed of L layers with (1) output weights: $\boldsymbol{H} \in \mathbb{R}^{m\times d}$, (2) self-attention weights: combined key-query weights $(\boldsymbol{W}_i)_{i=1}^L \in \mathbb{R}^{d\times d}$ and value weights $(\boldsymbol{W}_{vi})_{i=1}^L \in \mathbb{R}^{d\times d}$, (3) MLP weights: $(\boldsymbol{M}_j^{(i)})_{i=1,j=1}^{L,m} \in \mathbb{R}^{d\times d}$ and activation $\phi \in \{ReLU, Identity\}$. For some $\Gamma > 0$, assume $\|\boldsymbol{W}_{vi}\| \le 1, \|\boldsymbol{M}_j^{(i)}\| \le 1, \|\boldsymbol{W}_i\| \le \Gamma/2$ and $\|\boldsymbol{H}\|_{2,\infty} \le c/m$. Suppose input space is $\mathcal{S} = [\boldsymbol{z}_1\ \boldsymbol{z}_2\ \dots\ \boldsymbol{z}_m]^\top$ with $\|\boldsymbol{z}_i\| \le 1$. The model prediction is given as follows*

- *$\mathcal{S}_{(0)} = \mathcal{S}$. Layer i outputs $\mathcal{S}_{(i)} = \texttt{Parallel\_MLP}_{\boldsymbol{M}^{(i)}}(\texttt{Att}_{\boldsymbol{W}_i,\boldsymbol{W}_{vi}}(\mathcal{S}_{(i-1)}))$. Here the self-attention layer is given by $\texttt{Att}_{\boldsymbol{W}_i,\boldsymbol{W}_{vi}}(\mathcal{S}) = \mathbb{S}(\mathcal{S}\boldsymbol{W}_i\mathcal{S}^\top)\mathcal{S}\boldsymbol{W}_v$ and $\texttt{Parallel\_MLP}$ applies $f(\boldsymbol{x}) = \phi(\boldsymbol{M}_j^{(i)}\boldsymbol{x})$ on j^{th} token of the $\texttt{Att}$ output.*
- *$\texttt{TF}(\mathcal{S}) = \langle \boldsymbol{H}, \mathcal{S}_{(L)} \rangle$.*

First, this model is properly normalized in the sense that it can attain $\texttt{TF}(\mathcal{S}) = \pm c$. Second, this model obeys the stability guarantee

$$|\texttt{TF}(\mathcal{S}) - \texttt{TF}(\mathcal{S}')| \le \frac{c}{m}((1+\Gamma)e^\Gamma)^L\|\mathcal{S} - \mathcal{S}'\|_{2,1}.$$

Proof. To see the first claim, let us set $\boldsymbol{W}_{vi} = \boldsymbol{M}_{i,l} = \boldsymbol{I}$ and set all tokens $\boldsymbol{z}_i$ to be identical i.e. $\mathcal{S} = \boldsymbol{1}_n\boldsymbol{z}^\top$. Additionally choose a $\boldsymbol{z}$ with $\|\boldsymbol{z}\| = 1$ and nonnegative entries. Observe that, thanks to the softmax structure, regardless of $\boldsymbol{W}_i$, we have that $\mathcal{S} = \texttt{Att}_{\boldsymbol{W}_i,\boldsymbol{W}_{vi}}(\mathcal{S}) = \mathbb{S}(\mathcal{S}\boldsymbol{W}_i\mathcal{S}^\top)\mathcal{S}$. After attention, MLPs again preserves the tokens i.e. $\phi(\boldsymbol{M}_{i,l}\boldsymbol{z}_i) = \boldsymbol{z}$ for $\phi \in \{\text{ReLU}, \text{Identity}\}$. Thus, after proceeding L layers of this, we find that $\texttt{TF}(\mathcal{S}) = \langle \boldsymbol{H}, \mathcal{S}\rangle = \sum_{i=1}^n h_i^\top\boldsymbol{z}$. To conclude, by setting all tokens $h_i = c\boldsymbol{z}/m$ or $h_i = -c\boldsymbol{z}/m$, we obtain the claim. Note that, in general $\mathcal{S}$ can be arbitrary (they don't have to be all same tokens): We can let $\boldsymbol{W} \to \infty$ (by allowing a larger Γ). This way the attention matrix $\mathbb{S}(\mathcal{S}\boldsymbol{W}\mathcal{S}^\top) \to \boldsymbol{I}$ so that we end up with the same argument of $\mathcal{S}$ being (almost perfectly) transmitted across the layers and we obtain $\pm c$ output by setting $\boldsymbol{H} = \mathcal{S}/m$.

To show the stability guarantee, we use Lemmas H.2 and H.3. Specifically, Lemma H.3 guarantees that

- After each layer we are guaranteed to have $\|\mathcal{S}_{(i)}\|_{2,\infty}, \|\mathcal{S}'_{(i)}\|_{2,\infty} \leq 1$.
- After each layer we are guaranteed to have $\|\mathcal{S}_{(i)} - \mathcal{S}'_{(i)}\|_{2,1} \leq (1+\Gamma)e^{\Gamma}\|\mathcal{S}_{(i-1)} - \mathcal{S}'_{(i-1)}\|_{2,1}$.

This means that, at the end of the final layer, we have

$$\|\mathcal{S}_{(L)} - \mathcal{S}'_{(L)}\|_{2,1} \leq ((1+\Gamma)e^{\Gamma})^L \|\mathcal{S} - \mathcal{S}'\|_{2,1}.$$

Collapsing these by the linear $\boldsymbol{H}$, we find that $|\mathtt{TF}(\mathcal{S}') - \mathtt{TF}(\mathcal{S})| \leq \left\langle \boldsymbol{H}, \mathcal{S}_{(L)} - \mathcal{S}'_{(L)} \right\rangle \leq \|\boldsymbol{H}\|_{2,\infty} \|\mathcal{S}_{(L)} - \mathcal{S}'_{(L)}\|_{2,1}$ concluding the proof. ∎

H.1.1 Proof of Theorem 9.1

Proof. We need to specialize Theorem H.1 to obtain the result. Observe that when prompts differ only at the inputs $\boldsymbol{z}_j = (\boldsymbol{x}_j, y_j)$ with $\boldsymbol{z}'_j = (\boldsymbol{x}'_j, y'_j)$, we have that $\|\boldsymbol{X}_{\text{prompt}} - \boldsymbol{X}'_{\text{prompt}}\|_{2,1} \leq 2$. This implies that $|\mathtt{TF}(\boldsymbol{X}_{\text{prompt}}) - \mathtt{TF}(\boldsymbol{X}'_{\text{prompt}})| \leq \frac{2}{2m-1}((1+\Gamma)e^{\Gamma})^D$ for a depth D transformer. Finally, since the loss function ℓ is L-Lipschitz, we obtain the result $K = 2L((1+\Gamma)e^{\Gamma})^D$. ∎

H.2 Proofs and Supplementary Results for Sections 9.3,9.4,9.5

H.2.1 Proof of Theorem 9.2

Theorem H.2 (Theorem 9.2 restated) *Suppose Assumption 9.1 holds and assume loss function $\ell(\boldsymbol{y}, \hat{\boldsymbol{y}})$ is L-Lipschitz for all $\boldsymbol{y} \in \mathcal{Y}$ and takes values in $[0, B]$. Let $\widehat{\boldsymbol{\alpha}}$ be the empirical solution of* (ICL-ERM) *and $\mathcal{N}(\mathcal{A}, \rho, u)$ be the covering number of the algorithm space $\mathcal{A}$ following Definition 9.1&9.2. Then with probability at least $1 - 2\delta$, the excess MTL risk in* (MTL-risk) *obeys*

$$R_{MTL}(\widehat{\boldsymbol{\alpha}}) \leq \inf_{\epsilon > 0} \left\{ 4L\epsilon + 2(B + K\log n)\sqrt{\frac{\log(\mathcal{N}(\mathcal{A}, \rho, \epsilon)/\delta)}{cnT}} \right\}.$$

Additionally, set $D := \sup_{\boldsymbol{\alpha},\boldsymbol{\alpha}' \in \mathcal{A}} \rho(\boldsymbol{\alpha}, \boldsymbol{\alpha}')$ *and assume* $D < \infty$. *With probability at least* $1 - 4\delta$,

$$R_{MTL}(\widehat{\boldsymbol{\alpha}}) \leq \inf_{\epsilon > 0} \left\{ 8L\epsilon + 8(2L + K \log n) \int_{\epsilon}^{D/2} \sqrt{\frac{\log\left(\log \frac{D}{\epsilon} \cdot \mathcal{N}(\mathcal{A}, \rho, u)/\delta\right)}{c'nT}} du \right\} + 2(B + K \log n) \sqrt{\frac{\log(1/\delta)}{cnT}}.$$

Proof. Recall the MTL problem setting of independent (input, label) pairs in Section 9.2: There are T tasks each with n in-context training samples denoted by $(\mathcal{S}_t)_{t=1}^T \overset{\text{i.i.d.}}{\sim} (\mathcal{D}_t)_{t=1}^T$ where $\mathcal{S}_t = \{(\boldsymbol{x}_{ti}, \boldsymbol{y}_{ti})\}_{i=1}^n$, and let $\mathcal{S}_{\text{all}} = \bigcup_{t=1}^T \mathcal{S}_t$. We use $\mathcal{A}$ to denote the algorithm set. For a $\boldsymbol{\alpha} \in \mathcal{A}$, we define the training risk $\widehat{\mathcal{L}}_{\mathcal{S}_{\text{all}}}(\boldsymbol{\alpha}) = \frac{1}{nT} \sum_{t=1}^T \sum_{i=1}^n \ell(\boldsymbol{y}_{ti}, f^{\boldsymbol{\alpha}}_{\mathcal{S}_t^{(i-1)}}(\boldsymbol{x}_{ti}))$, and the test risk $\mathcal{L}_{\text{MTL}}(\boldsymbol{\alpha}) = \mathbb{E}[\widehat{\mathcal{L}}_{\mathcal{S}_{\text{all}}}(\boldsymbol{\alpha})]$. Define empirical risk minima $\widehat{\boldsymbol{\alpha}} = \arg\min_{\boldsymbol{\alpha} \in \mathcal{A}} \widehat{\mathcal{L}}_{\mathcal{S}_{\text{all}}}(\boldsymbol{\alpha})$ and population minima $\boldsymbol{\alpha}^\star = \arg\min_{\boldsymbol{\alpha} \in \mathcal{A}} \mathcal{L}_{\text{MTL}}(\boldsymbol{\alpha})$. For cleaner exposition, in the following discussion, we drop the subscripts MTL and $\mathcal{S}_{\text{all}}$. The excess MTL risk is decomposed as follows:

$$\begin{aligned} R_{\text{MTL}}(\widehat{\boldsymbol{\alpha}}) &= \mathcal{L}(\widehat{\boldsymbol{\alpha}}) - \mathcal{L}(\boldsymbol{\alpha}^\star) \\ &= \underbrace{\mathcal{L}(\widehat{\boldsymbol{\alpha}}) - \widehat{\mathcal{L}}(\widehat{\boldsymbol{\alpha}})}_{a} + \underbrace{\widehat{\mathcal{L}}(\widehat{\boldsymbol{\alpha}}) - \widehat{\mathcal{L}}(\boldsymbol{\alpha}^\star)}_{b} + \underbrace{\widehat{\mathcal{L}}(\boldsymbol{\alpha}^\star) - \mathcal{L}(\boldsymbol{\alpha}^\star)}_{c}. \end{aligned}$$

Since $\widehat{\boldsymbol{\alpha}}$ is the minimizer of empirical risk, we have $b \leq 0$. To proceed, we consider the concentration problem of upper bounding $\sup_{\boldsymbol{\alpha} \in \mathcal{A}} |\mathcal{L}(\boldsymbol{\alpha}) - \widehat{\mathcal{L}}(\boldsymbol{\alpha})|$.

Step 1: We start with a concentration bound $|\mathcal{L}(\boldsymbol{\alpha}) - \widehat{\mathcal{L}}(\boldsymbol{\alpha})|$ for a fixed $\boldsymbol{\alpha} \in \mathcal{A}$. Recall that we define the test/train risks of each task as follows:

$$\begin{aligned} \widehat{\mathcal{L}}_t(\boldsymbol{\alpha}) &:= \frac{1}{n} \sum_{i=1}^n \ell(\boldsymbol{y}_{ti}, f^{\boldsymbol{\alpha}}_{\mathcal{S}_t^{(i-1)}}(\boldsymbol{x}_{ti})), \quad \text{and} \\ \mathcal{L}_t(\boldsymbol{\alpha}) &:= \mathbb{E}_{\mathcal{S}_t}\left[\widehat{\mathcal{L}}_t(\boldsymbol{\alpha})\right] = \mathbb{E}_{\mathcal{S}_t}\left[\frac{1}{n} \sum_{i=1}^n \ell(\boldsymbol{y}_{ti}, f^{\boldsymbol{\alpha}}_{\mathcal{S}_t^{(i-1)}}(\boldsymbol{x}_{ti}))\right], \quad \forall t \in [T]. \end{aligned}$$

Define the random variables $X_{t,i} = \mathbb{E}[\widehat{\mathcal{L}}_t(\boldsymbol{\alpha}) | \mathcal{S}_t^{(i)}]$ for $i \in [n]$ and $t \in [T]$, that is, $X_{t,i}$ is the expectation over $\widehat{\mathcal{L}}_t(\boldsymbol{\alpha})$ given training sequence $\mathcal{S}_t^{(i)} = \{(\boldsymbol{x}_{tj}, \boldsymbol{y}_{tj})\}_{j=1}^i$ (which are the filtrations). With this, we have that $X_{t,n} = \mathbb{E}[\widehat{\mathcal{L}}_t(\boldsymbol{\alpha}) | \mathcal{S}_t^{(n)}] = \widehat{\mathcal{L}}_t(\boldsymbol{\alpha})$ and $X_{t,0} = \mathbb{E}[\widehat{\mathcal{L}}_t(\boldsymbol{\alpha})] = \mathcal{L}_t(\boldsymbol{\alpha})$. More generally, $(X_{t,0}, X_{t,1}, \ldots, X_{t,n})$ is a martingale sequence since, for every t in $[T]$, we have that $\mathbb{E}[X_{t,i} | \mathcal{S}_t^{(i-1)}] = X_{t,i-1}$.

For notational simplicitly, in the following discussion, we omit the subscript t from $\boldsymbol{x}, \boldsymbol{y}$

and $\boldsymbol{Z}$ as they will be clear from left hand-side variable $X_{t,i}$. We have that

$$\begin{aligned} X_{t,i} &= \mathbb{E}\left[\frac{1}{n}\sum_{j=1}^{n}\ell(\boldsymbol{y}_j, f^{\boldsymbol{\alpha}}_{\mathcal{S}^{(j-1)}}(\boldsymbol{x}_j))\middle|\mathcal{S}^{(i)}\right] \\ &= \frac{1}{n}\sum_{j=1}^{i}\ell(\boldsymbol{y}_j, f^{\boldsymbol{\alpha}}_{\mathcal{S}^{(j-1)}}(\boldsymbol{x}_j)) + \frac{1}{n}\sum_{j=i+1}^{n}\mathbb{E}\left[\ell(\boldsymbol{y}_j, f^{\boldsymbol{\alpha}}_{\mathcal{S}^{(j-1)}}(\boldsymbol{x}_j))\middle|\mathcal{S}^{(i)}\right] \end{aligned}$$

Next, we wish to upper bound the martingale increments i.e. the difference of neighbors. Let $\mathcal{S}^{(i:i)} = \mathcal{S}^{(i)} - \mathcal{S}^{(i-1)}$ denote the i'th element.

$$\begin{aligned} |X_{t,i} - X_{t,i-1}| &= \left|\mathbb{E}\left[\frac{1}{n}\sum_{j=1}^{n}\ell(\boldsymbol{y}_j, f^{\boldsymbol{\alpha}}_{\mathcal{S}^{(j-1)}}(\boldsymbol{x}_j))\middle|\mathcal{S}^{(i)}\right] - \mathbb{E}\left[\frac{1}{n}\sum_{j=1}^{n}\ell(\boldsymbol{y}_j, f^{\boldsymbol{\alpha}}_{\mathcal{S}^{(j-1)}}(\boldsymbol{x}_j))\middle|\mathcal{S}^{(i-1)}\right]\right| \\ &\leq \frac{1}{n}\sum_{j=i}^{n}\left|\mathbb{E}\left[\ell(\boldsymbol{y}_j, f^{\boldsymbol{\alpha}}_{\mathcal{S}^{(j-1)}}(\boldsymbol{x}_j))\middle|\mathcal{S}^{(i)}\right] - \mathbb{E}\left[\ell(\boldsymbol{y}_j, f^{\boldsymbol{\alpha}}_{\mathcal{S}^{(j-1)}}(\boldsymbol{x}_j))\middle|\mathcal{S}^{(i-1)}\right]\right| \\ &\overset{(a)}{\leq} \frac{B}{n} + \frac{1}{n}\sum_{j=i+1}^{n}\left|\mathbb{E}\left[\ell(\boldsymbol{y}_j, f^{\boldsymbol{\alpha}}_{\mathcal{S}^{(j-1)}}(\boldsymbol{x}_j))\middle|\mathcal{S}^{(i)}\right] - \mathbb{E}\left[\ell(\boldsymbol{y}_j, f^{\boldsymbol{\alpha}}_{\mathcal{S}^{(j-1)}}(\boldsymbol{x}_j))\middle|\mathcal{S}^{(i-1)}\right]\right|. \end{aligned}$$

Here, (a) follows from the fact that loss function $\ell(\cdot,\cdot)$ is bounded by B. To proceed, call the right side terms $D_j := |\,\mathbb{E}[\ell(\boldsymbol{y}_j, f^{\boldsymbol{\alpha}}_{\mathcal{S}^{(j-1)}}(\boldsymbol{x}_j))\big|\mathcal{S}^{(i)}] - \mathbb{E}[\ell(\boldsymbol{y}_j, f^{\boldsymbol{\alpha}}_{\mathcal{S}^{(j-1)}}(\boldsymbol{x}_j))\big|\mathcal{S}^{(i-1)}]|$. Denote $\boldsymbol{z}'_\ell$ to be the realized values of $\boldsymbol{z}_\ell = (\boldsymbol{y}_\ell, \boldsymbol{x}_\ell)$ given $\mathcal{S}^{(i)}$. Let $\mathcal{S} := (\boldsymbol{z}'_1, \ldots, \boldsymbol{z}'_i, \boldsymbol{z}_{i+1}, \ldots, \boldsymbol{z}_j)$ and $\mathcal{S}' := (\boldsymbol{z}'_1, \ldots, \boldsymbol{z}'_{i-1}, \boldsymbol{z}_i, \ldots, \boldsymbol{z}_j)$. Observe that, $\mathcal{S}'$ and $\mathcal{S}$ differs in only at ith index and $i < j$, thus, utilizing Assumption 9.1,

$$D_j := |\,\mathbb{E}[\ell(\boldsymbol{y}_j, f^{\boldsymbol{\alpha}}_{\mathcal{S}}(\boldsymbol{x}_j))] - \mathbb{E}[\ell(\boldsymbol{y}_j, f^{\boldsymbol{\alpha}}_{\mathcal{S}'}(\boldsymbol{x}_j))]| \leq \frac{K}{j}. \tag{H.3}$$

Combining above, for any $n \geq i \geq 1$, we obtain

$$|X_{t,i} - X_{t,i-1}| \leq \frac{B}{n} + \sum_{j=i+1}^{n}\frac{K}{jn} \leq \frac{B + K\log n}{n}.$$

Recall that $|\mathcal{L}_t(\boldsymbol{\alpha}) - \widehat{\mathcal{L}}_t(\boldsymbol{\alpha})| = |X_{t,0} - X_{t,n}|$ and for every $t \in [T]$, we have $\sum_{i=1}^{n}|X_{t,i} - X_{t,i-1}|^2 \leq \frac{(B+K\log n)^2}{n}$. As a result, applying Azuma-Hoeffding's inequality, we obtain

$$\mathbb{P}(|\mathcal{L}_t(\boldsymbol{\alpha}) - \widehat{\mathcal{L}}_t(\boldsymbol{\alpha})| \geq \tau) \leq 2e^{-\frac{n\tau^2}{2(B+K\log n)^2}}, \quad \forall t \in [T]. \tag{H.4}$$

Let us consider $Y_t := \mathcal{L}_t(\boldsymbol{\alpha}) - \widehat{\mathcal{L}}_t(\boldsymbol{\alpha})$ for $t \in [T]$. Then, $(Y_t)_{t=1}^T$ are i.i.d. zero mean sub-Gaussian random variables. There exists an absolute constant $c_1 > 0$ such that, the sub-

gaussian norm, denoted by $\|\cdot\|_{\psi_2}$, obeys $\|Y_t\|_{\psi_2}^2 < \frac{c_1(B+K\log n)^2}{n}$ by [195, Proposition 2.5.2]. Applying Hoeffding's inequality, we derive

$$\mathbb{P}\left(\left|\frac{1}{T}\sum_{t=1}^{T} Y_t\right| \geq \tau\right) \leq 2e^{-\frac{cnT\tau^2}{(B+K\log n)^2}} \Longrightarrow \mathbb{P}(|\widehat{\mathcal{L}}(\boldsymbol{\alpha}) - \mathcal{L}(\boldsymbol{\alpha})| \geq \tau) \leq 2e^{-\frac{cnT\tau^2}{(B+K\log n)^2}} \tag{H.5}$$

where $c > 0$ is an absolute constant. Therefore, we have that for any $\boldsymbol{\alpha} \in \mathcal{A}$, with probability at least $1 - 2\delta$,

$$|\widehat{\mathcal{L}}(\boldsymbol{\alpha}) - \mathcal{L}(\boldsymbol{\alpha})| \leq (B + K\log n)\sqrt{\frac{\log(1/\delta)}{cnT}}. \tag{H.6}$$

Step 2: Next, we turn to bound $\sup_{\boldsymbol{\alpha}\in\mathcal{A}} |\mathcal{L}(\boldsymbol{\alpha}) - \widehat{\mathcal{L}}(\boldsymbol{\alpha})|$ where $\mathcal{A}$ is assumed to be a continuous search space. To start with, set $g(\boldsymbol{\alpha}) := \mathcal{L}(\boldsymbol{\alpha}) - \widehat{\mathcal{L}}(\boldsymbol{\alpha})$ and we aim to bound $\sup_{\boldsymbol{\alpha}\in\mathcal{A}} |g(\boldsymbol{\alpha})|$. Following Definition 9.2, for $\epsilon > 0$, let $\mathcal{A}_\epsilon$ be a minimal ϵ-cover of $\mathcal{A}$ in terms of distance metric ρ. Therefore, $\mathcal{A}_\epsilon$ is a discrete set with cardinality $|\mathcal{A}_\epsilon| := \mathcal{N}(\mathcal{A}, \rho, \epsilon)$. Then, we have

$$\sup_{\boldsymbol{\alpha}\in\mathcal{A}} |\mathcal{L}(\boldsymbol{\alpha}) - \widehat{\mathcal{L}}(\boldsymbol{\alpha})| \leq \sup_{\boldsymbol{\alpha}\in\mathcal{A}} \min_{\boldsymbol{\alpha}'\in\mathcal{A}_\epsilon} |g(\boldsymbol{\alpha}) - g(\boldsymbol{\alpha}')| + \max_{\boldsymbol{\alpha}\in\mathcal{A}_\epsilon} |g(\boldsymbol{\alpha})|\,.$$

• We start by bounding $\sup_{\boldsymbol{\alpha}\in\mathcal{A}} \min_{\boldsymbol{\alpha}'\in\mathcal{A}_\epsilon} |g(\boldsymbol{\alpha}) - g(\boldsymbol{\alpha}')|$. We will utilize that loss function $\ell(\cdot,\cdot)$ is L-Lipschitz. For any $\boldsymbol{\alpha} \in \mathcal{A}$, let $\boldsymbol{\alpha}' \in \mathcal{A}_\epsilon$ be its neighbor following Definition 9.2. We have that

$$\begin{aligned}
\left|\widehat{\mathcal{L}}(\boldsymbol{\alpha}) - \widehat{\mathcal{L}}(\boldsymbol{\alpha}')\right| &= \left|\frac{1}{nT}\sum_{t=1}^{T}\sum_{i=1}^{n}\left(\ell(\boldsymbol{y}_{ti}, f^{\boldsymbol{\alpha}}_{\mathcal{S}_t^{(i-1)}}(\boldsymbol{x}_{ti})) - \ell(\boldsymbol{y}_{ti}, f^{\boldsymbol{\alpha}'}_{\mathcal{S}_t^{(i-1)}}(\boldsymbol{x}_{ti}))\right)\right| \\
&\leq \frac{L}{nT}\sum_{t=1}^{T}\sum_{i=1}^{n}\left\|f^{\boldsymbol{\alpha}}_{\mathcal{S}_t^{(i-1)}}(\boldsymbol{x}_{ti}) - f^{\boldsymbol{\alpha}'}_{\mathcal{S}_t^{(i-1)}}(\boldsymbol{x}_{ti})\right\|_{\ell_2} \\
&\leq L\epsilon.
\end{aligned}$$

Since the same bound applies to all data-sequences, we also obtain that for any $\boldsymbol{\alpha} \in \mathcal{A}$,

$$|\mathcal{L}(\boldsymbol{\alpha}) - \mathcal{L}(\boldsymbol{\alpha}')| \leq L\epsilon.$$

Therefore,

$$\sup_{\boldsymbol{\alpha}\in\mathcal{A}} \min_{\boldsymbol{\alpha}'\in\mathcal{A}_\epsilon} |g(\boldsymbol{\alpha}) - g(\boldsymbol{\alpha}')| \leq \sup_{\boldsymbol{\alpha}\in\mathcal{A}} \min_{\boldsymbol{\alpha}'\in\mathcal{A}_\epsilon} \left|\widehat{\mathcal{L}}(\boldsymbol{\alpha}) - \widehat{\mathcal{L}}(\boldsymbol{\alpha}')\right| + |\mathcal{L}(\boldsymbol{\alpha}) - \mathcal{L}(\boldsymbol{\alpha}')| \leq 2L\epsilon. \tag{H.7}$$

• Next, we turn to bound the second term $\max_{\boldsymbol{\alpha}\in\mathcal{A}_\epsilon}|g(\boldsymbol{\alpha})|$. Applying union bound directly on $\mathcal{A}_\epsilon$ and combining it with (H.6), then we will have that with probability at least $1-2\delta$,

$$\max_{\boldsymbol{\alpha}\in\mathcal{A}_\epsilon}|g(\boldsymbol{\alpha})| \leq (B+K\log n)\sqrt{\frac{\log(\mathcal{N}(\mathcal{A},\rho,\epsilon)/\delta)}{cnT}}. \tag{H.8}$$

Proof of Eq. (9.4)**:** Combining the upper bound above with the perturbation bound (H.7), we obtain that

$$\max_{\boldsymbol{\alpha}\in\mathcal{A}}|g(\boldsymbol{\alpha})| \leq 2L\epsilon + (B+K\log n)\sqrt{\frac{\log(\mathcal{N}(\mathcal{A},\rho,\epsilon)/\delta)}{cnT}}.$$

This in turn concludes the proof of (9.4) since $R_{\mathrm{MTL}}(\widehat{\boldsymbol{\alpha}}) \leq 2\sup_{\boldsymbol{\alpha}\in\mathcal{A}}|\mathcal{L}(\boldsymbol{\alpha})-\widehat{\mathcal{L}}(\boldsymbol{\alpha})|$.

Proof of Eq. (9.5)**:** To conclude, we aim to establish (9.5). To this end, we will bound $\max_{\boldsymbol{\alpha}\in\mathcal{A}_\epsilon}|g(\boldsymbol{\alpha})|$ via successive ϵ-covers which is the chaining argument. Following Definition 9.2, let $D := \sup_{\boldsymbol{\alpha},\boldsymbol{\alpha}'\in\mathcal{A}}\rho(\boldsymbol{\alpha},\boldsymbol{\alpha}')$. Define $M := \min\{m : 2^m\epsilon \geq D\}$, and for any $m \in [M]$, let $\mathcal{U}_m$ denote the minimal $2^m\epsilon$-cover of $\mathcal{U}_{m-1}$, where $\mathcal{U}_0 := \mathcal{A}_\epsilon$. Since $\mathcal{U}_M \subseteq \mathcal{U}_{M-1}\cdots \subseteq \mathcal{U}_0 \subset \mathcal{A}$, we have $|\mathcal{U}_m| = \mathcal{N}(\mathcal{U}_{m-1},\rho,2^m\epsilon) \leq \mathcal{N}(\mathcal{A},\rho,2^m\epsilon)$ and $|\mathcal{U}_M| \leq \mathcal{N}(\mathcal{A},\rho,D) = 1$. Let $\boldsymbol{\alpha}^M \in \mathcal{U}_M$ denote the unique algorithm hypothesis in $\mathcal{U}_M$. We have that

$$\begin{aligned}\max_{\boldsymbol{\alpha}\in\mathcal{A}_\epsilon}|g(\boldsymbol{\alpha})| &\leq \max_{\boldsymbol{\alpha}\in\mathcal{U}_0}\left|g(\boldsymbol{\alpha})-g(\boldsymbol{\alpha}^M)\right| + \left|g(\boldsymbol{\alpha}^M)\right| \\ &\leq \sum_{m=0}^{M-1}\max_{\boldsymbol{\alpha}\in\mathcal{U}_m}\min_{\boldsymbol{\alpha}'\in\mathcal{U}_{m+1}}|g(\boldsymbol{\alpha})-g(\boldsymbol{\alpha}')| + \left|g(\boldsymbol{\alpha}^M)\right|. \end{aligned}\tag{H.9}$$

In what follows, we will prove that for any $\boldsymbol{\alpha},\boldsymbol{\alpha}'$ satisfying $\rho(\boldsymbol{\alpha},\boldsymbol{\alpha}') \leq u$ ($u>0$), with high probability, $|g(\boldsymbol{\alpha})-g(\boldsymbol{\alpha}')|$ is bounded by $\frac{u}{\sqrt{nT}}$ up to logarithmic terms.

Apply similar martingale sequence analysis as in Step 1. This time, we set $X_{t,i} = \mathbb{E}[\widehat{\mathcal{L}}_t(\boldsymbol{\alpha})-\widehat{\mathcal{L}}_t(\boldsymbol{\alpha}')|\mathcal{S}_t^{(i)}]$ where we assume $\rho(\boldsymbol{\alpha},\boldsymbol{\alpha}') \leq u$. Similarly, we have that $X_{t,n} = \widehat{\mathcal{L}}_t(\boldsymbol{\alpha})-\widehat{\mathcal{L}}_t(\boldsymbol{\alpha}')$, and $X_{t,0} = \mathbb{E}[\widehat{\mathcal{L}}_t(\boldsymbol{\alpha})-\widehat{\mathcal{L}}_t(\boldsymbol{\alpha}')] = \mathcal{L}_t(\boldsymbol{\alpha})-\mathcal{L}_t(\boldsymbol{\alpha}')$. Therefore, the sequences $\{X_{t,0},\ldots,X_{t,n}\}, t\in[T]$ are Martingale sequences with respect to $\mathbb{E}[X_{t,i}|\mathcal{S}_t^{(i-1)}] = X_{t,i-1}$. We again omit the subscript t for $\boldsymbol{x},\boldsymbol{y}$ and $\mathcal{S}$ in the following and try to bound the difference of

neighbors.

$$
\begin{aligned}
|X_{t,i} - X_{t,i-1}| &= \left| \mathbb{E}\left[\widehat{\mathcal{L}}_t(\boldsymbol{\alpha}) - \widehat{\mathcal{L}}_t(\boldsymbol{\alpha}') \middle| \mathcal{S}^{(i)} \right] - \mathbb{E}\left[\widehat{\mathcal{L}}_t(\boldsymbol{\alpha}) - \widehat{\mathcal{L}}_t(\boldsymbol{\alpha}') \middle| \mathcal{S}^{(i-1)} \right] \right| \\
&\leq \frac{1}{n} \sum_{j=i}^{n} \left| \mathbb{E}\left[\ell(\boldsymbol{y}_j, f^{\boldsymbol{\alpha}}_{\mathcal{S}^{(j-1)}}(\boldsymbol{x}_j)) - \ell(\boldsymbol{y}_j, f^{\boldsymbol{\alpha}'}_{\mathcal{S}^{(j-1)}}(\boldsymbol{x}_j)) \middle| \mathcal{S}^{(i)} \right] \right. \\
&\qquad \left. - \mathbb{E}\left[\ell(\boldsymbol{y}_j, f^{\boldsymbol{\alpha}}_{\mathcal{S}^{(j-1)}}(\boldsymbol{x}_j)) - \ell(\boldsymbol{y}_j, f^{\boldsymbol{\alpha}'}_{\mathcal{S}^{(j-1)}}(\boldsymbol{x}_j)) \middle| \mathcal{S}^{(i-1)} \right] \right| \\
&\overset{(d)}{\leq} \frac{2Lu}{n} + \frac{1}{n} \sum_{j=i+1}^{n} \left| \mathbb{E}\left[\ell(\boldsymbol{y}_j, f^{\boldsymbol{\alpha}}_{\mathcal{S}^{(j-1)}}(\boldsymbol{x}_j)) - \ell(\boldsymbol{y}_j, f^{\boldsymbol{\alpha}'}_{\mathcal{S}^{(j-1)}}(\boldsymbol{x}_j)) \middle| \mathcal{S}^{(i)} \right] \right. \\
&\qquad \left. - \mathbb{E}\left[\ell(\boldsymbol{y}_j, f^{\boldsymbol{\alpha}}_{\mathcal{S}^{(j-1)}}(\boldsymbol{x}_j)) - \ell(\boldsymbol{y}_j, f^{\boldsymbol{\alpha}'}_{\mathcal{S}^{(j-1)}}(\boldsymbol{x}_j)) \middle| \mathcal{S}^{(i-1)} \right] \right| \\
&\overset{(e)}{\leq} \frac{2Lu}{n} + \frac{1}{n} \sum_{j=i+1}^{n} \frac{Ku}{j} < \frac{2Lu + Ku \log n}{n}.
\end{aligned} \tag{H.10}
$$

for $i < n$. Here, (d) is from the facts that loss function $\ell(\cdot, \cdot)$ is L-Lipschitzness and $\rho(\boldsymbol{\alpha}, \boldsymbol{\alpha}') \leq u$ by following the same analysis in deriving (H.7), and (e) follows the Assumption 9.1. Then we have

$$
|X_{t,n} - X_{t,n-1}| \leq \frac{2Lu}{n} < \frac{2Lu + Ku \log n}{n}.
$$

Note that $|\mathcal{L}_t(\boldsymbol{\alpha}) - \mathcal{L}_t(\boldsymbol{\alpha}') - (\widehat{\mathcal{L}}_t(\boldsymbol{\alpha}) - \widehat{\mathcal{L}}_t(\boldsymbol{\alpha}'))| = |X_{t,0} - X_{t,n}|$ and for every $t \in [T]$, we have $\sum_{i=1}^{n} |X_{t,i} - X_{t,i-1}|^2 \leq \frac{u^2(2L + K \log n)^2}{n}$. As a result of applying Azuma-Hoeffding's inequality, we obtain

$$
\mathbb{P}(|\mathcal{L}_t(\boldsymbol{\alpha}) - \mathcal{L}_t(\boldsymbol{\alpha}') - (\widehat{\mathcal{L}}_t(\boldsymbol{\alpha}) - \widehat{\mathcal{L}}_t(\boldsymbol{\alpha}'))| \geq \tau) \leq 2e^{-\frac{n\tau^2}{2u^2(2L + K \log n)^2}}, \quad \forall t \in [T].
$$

Now let us instead consider $Y_t := g(\boldsymbol{\alpha}) - g(\boldsymbol{\alpha}')$ for $t \in [T]$. Then following proof as in Step 1, we derive

$$
\mathbb{P}\left(\left| \frac{1}{T} \sum_{t=1}^{T} Y_t \right| \geq \tau \right) < 2e^{-\frac{c' n T \tau^2}{u^2(2L + K \log n)^2}} \Longrightarrow \mathbb{P}(|g(\boldsymbol{\alpha}) - g(\boldsymbol{\alpha}')| \geq \tau) \leq 2e^{-\frac{c' n T \tau^2}{u^2(2L + K \log n)^2}}
$$

where $c' > 0$ is an absolute constant. Consider the discrete set $\mathcal{U}_m$ with cardinality $|\mathcal{U}_m| = \mathcal{N}(\mathcal{U}_m, \rho, 2^m \epsilon) \leq \mathcal{N}(\mathcal{A}, \rho, 2^m \epsilon)$ and its $2^{m+1}\epsilon$-cover $\mathcal{U}_{m+1}$. Applying union bound over $\mathcal{U}_m$,

we have that with probability at least $1-2\delta$,

$$\max_{\boldsymbol{\alpha}\in\mathcal{U}_m}\min_{\boldsymbol{\alpha}'\in\mathcal{U}_{m+1}}|g(\boldsymbol{\alpha})-g(\boldsymbol{\alpha}')|\leq 2^{m+1}\epsilon(2L+K\log n)\sqrt{\frac{\log(\mathcal{N}(\mathcal{A},\rho,2^m\epsilon)/\delta)}{c'nT}}.$$

Now by again applying union bound, with probability at least $1-2\delta$, the first term in (H.9) is bounded by

$$\begin{aligned}&\sum_{m=0}^{M-1}\max_{\boldsymbol{\alpha}\in\mathcal{U}_m}\min_{\boldsymbol{\alpha}'\in\mathcal{U}_{m+1}}|g(\boldsymbol{\alpha})-g(\boldsymbol{\alpha}')| &\text{(H.11)}\\ &\leq(2L+K\log n)\sum_{m=0}^{M-1}2^{m+1}\epsilon\sqrt{\frac{\log(M\cdot\mathcal{N}(\mathcal{A},\rho,2^m\epsilon)/\delta)}{c'nT}}\\ &\leq 4(2L+K\log n)\int_{\epsilon/2}^{D/2}\sqrt{\frac{\log(M\cdot\mathcal{N}(\mathcal{A},\rho,u)/\delta)}{c'nT}}du. &\text{(H.12)}\end{aligned}$$

Now combining the results of (H.6), (H.9) and (H.12), and following the evidence that $\boldsymbol{\alpha}^M\in\mathcal{U}_M$ is unique, we bound $\sup_{\boldsymbol{\alpha}\in\mathcal{A}_\epsilon}|g(\boldsymbol{\alpha})|$ as follows, that with probability at least $1-4\delta$

$$\sup_{\boldsymbol{\alpha}\in\mathcal{A}_\epsilon}|g(\boldsymbol{\alpha})|\leq 4(2L+K\log n)\int_{\epsilon/2}^{D/2}\sqrt{\frac{\log(M\cdot\mathcal{N}(\mathcal{A},\rho,u)/\delta)}{c'nT}}du+(B+K\log n)\sqrt{\frac{\log(1/\delta)}{cnT}}. \tag{H.13}$$

Here $D:=\sup_{\boldsymbol{\alpha},\boldsymbol{\alpha}'\in\mathcal{A}}\rho(\boldsymbol{\alpha},\boldsymbol{\alpha}')$ and $M:=\min\{m:2^m\epsilon\geq D\}$.

- Combining (H.7) and (H.13), we obtain that with probability at least $1-4\delta$,

$$\begin{aligned}&\sup_{\boldsymbol{\alpha}\in\mathcal{A}}\left|\mathcal{L}(\boldsymbol{\alpha})-\widehat{\mathcal{L}}(\boldsymbol{\alpha})\right| &\text{(H.14)}\\ &\leq\inf_{\epsilon>0}\left\{4L\epsilon+4(2L+K\log n)\int_{\epsilon}^{D/2}\sqrt{\frac{\log(M\cdot\mathcal{N}(\mathcal{A},\rho,u)/\delta)}{c'nT}}du+(B+K\log n)\sqrt{\frac{\log(1/\delta)}{cnT}}\right\},\end{aligned}$$

where $D:=\sup_{\boldsymbol{\alpha},\boldsymbol{\alpha}'\in\mathcal{A}}\rho(\boldsymbol{\alpha},\boldsymbol{\alpha}')$ and $M:=\min\{m:2^{m+1}\epsilon\geq D\}\leq\log\frac{D}{\epsilon}$.

Applying $R_{\mathrm{MTL}}(\widehat{\boldsymbol{\alpha}})\leq 2\sup_{\boldsymbol{\alpha}\in\mathcal{A}}\left|\mathcal{L}(\boldsymbol{\alpha})-\widehat{\mathcal{L}}(\boldsymbol{\alpha})\right|$ completes the proof. ∎

Till now, we consider the setting where each task is trained with only one trajectory. In the following, we also consider the case where each task contains multiple trajectories. To start with, we define the following objective function as an extension of (ICL-ERM) to the

multi-trajectory setting.

$$\widehat{\boldsymbol{\alpha}} = \arg\min_{\boldsymbol{\alpha}\in\mathcal{A}} \widehat{\mathcal{L}}_{\mathcal{S}_{\text{all}}}(\boldsymbol{\alpha}) := \frac{1}{TM}\sum_{t=1}^{T}\sum_{m=1}^{M} \widehat{\mathcal{L}}_{t,m}(\boldsymbol{\alpha}) \tag{H.15}$$
$$\text{where} \quad \widehat{\mathcal{L}}_{t,m}(\boldsymbol{\alpha}) = \frac{1}{n}\sum_{i=1}^{n} \ell(\boldsymbol{y}_{tmi}, f^{\boldsymbol{\alpha}}_{\mathcal{S}_t^{(i-1)}}(\boldsymbol{x}_{tmi})).$$

Here, we assume each task $t \in [T]$ contains M trajectories, and $\mathcal{S}_{\text{all}} = \{\{\mathcal{S}_{t,m}\}_{m=1}^{M}\}_{t=1}^{T}$ where $\mathcal{S}_{t,m} = \{(\boldsymbol{x}_{tmi}, \boldsymbol{y}_{tmi})\}_{i=1}^{n}$ and $(\boldsymbol{x}_{tmi}, \boldsymbol{y}_{tmi}) \overset{\text{i.i.d.}}{\sim} \mathcal{D}_t$. Then the following theorem states a more general version of Theorem 9.2.

Theorem H.3 *Suppose the same assumptions as in Theorem 9.2 hold and let $\widehat{\boldsymbol{\alpha}}$ be the empirical solution of* (H.15)*. Then, with the same probability, we obtain identical bounds to Theorem 9.2 by updating T with TM in Equations* (9.4) *and* (9.5)*.*

Choosing $M = 1$ results in the exactly same bound as in Theorem 9.2.

Proof. Following the same proof steps, and then we derive similar result as (H.4):

$$\mathbb{P}(|\mathcal{L}_{t,m}(\boldsymbol{\alpha}) - \widehat{\mathcal{L}}_{t,m}(\boldsymbol{\alpha})| \geq \tau) \leq 2e^{-\frac{n\tau^2}{2(B+K\log n)^2}}, \quad \forall t \in [T], m \in [M]. \tag{H.16}$$

Let $Y_{t,m} := \mathcal{L}_{t,m}(\boldsymbol{\alpha}) - \widehat{\mathcal{L}}_{t,m}(\boldsymbol{\alpha})$. Since in-context samples are independent, $((Y_{t,m})_{m=1}^{M})_{t=1}^{T}$ are independent zero mean sub-Gaussian random variables, with norm $\|Y_{t,m}\|_{\psi_2} < \frac{c_1(B+K\log n)^2}{n}$. Applying Hoeffding's inequality, we derive

$$\mathbb{P}(|\widehat{\mathcal{L}}(\boldsymbol{\alpha}) - \mathcal{L}(\boldsymbol{\alpha})| \geq \tau) \leq 2e^{-\frac{cnMT\tau^2}{(B+K\log n)^2}} \tag{H.17}$$

where $c > 0$ is an absolute constant. Therefore, we have that for any $\boldsymbol{\alpha} \in \mathcal{A}$, with probability at least $1 - 2\delta$,

$$|\widehat{\mathcal{L}}(\boldsymbol{\alpha}) - \mathcal{L}(\boldsymbol{\alpha})| \leq (B + K\log n)\sqrt{\frac{\log(1/\delta)}{cnMT}}. \tag{H.18}$$

The result is simply replacing T with MT in (H.6). It is from the fact that trajectories are all independent no matter they are from the same task or not. By applying the similar analysis, the proof is competed. ∎

H.2.2 Transfer Discussion from the Lens of Task Diversity

In Section 9.4, we motivated the fact that transfer risk is controlled in terms of MTL risk and an additive term that captures the distributional distance i.e. $\mathcal{L}_{\mathcal{T}}(\boldsymbol{\alpha}) \leq \mathcal{L}_{\text{MTL}}(\boldsymbol{\alpha}) +$

$\text{dist}(\mathcal{T}, (\mathcal{D}_t)_{t=1}^T)$. The following definition is a generalization of this relation which can be used to formally control the transfer risk in terms of MTL risk.

Definition H.1 (Task diversity) *Following Section 9.2, we say that task $\mathcal{T}$ is (ν, ϵ)-diverse over the T source tasks if for any $\boldsymbol{\alpha}, \boldsymbol{\alpha}' \in \mathcal{A}$,*

$$\mathcal{L}_{\mathcal{T}}(\boldsymbol{\alpha}) - \mathcal{L}_{\mathcal{T}}(\boldsymbol{\alpha}') \leq \left(\frac{1}{T} \sum_{t=1}^{T} (\mathcal{L}_t(\boldsymbol{\alpha}) - \mathcal{L}_t(\boldsymbol{\alpha}')) \right) / \nu + \epsilon.$$

Now let us discuss transferability in light of this assumption and Thm 9.2. Consider the scenario where n is small and $T \to \infty$. The excess MTL risk will be small thanks to infinitely many tasks. The transfer risk would also be small because larger T results in higher diversity covering the task space. However, if the target task uses a different/longer prompt length, transfer may fail since the model never saw prompts longer than n. Conversely, if we let $n \to \infty$ and T to be small, although the MTL risk is again zero, due to lack of diversity, it may not benefit transfer learning strongly. Task diversity assumption leads to the following lemma that bounds transfer learning in terms of MTL risk.

Lemma H.4 *Consider the setting of Theorem 9.2. Let $\widehat{\boldsymbol{\alpha}}$ be the solution of* (ICL-ERM) *and assume that target task $\mathcal{T}$ is (ν, ϵ)-diverse over T source tasks. Then with the same probability as in Theorem 9.2, the excess transfer learning risk $R_{\mathcal{T}}(\widehat{\boldsymbol{\alpha}}) = \mathcal{L}_{\mathcal{T}}(\widehat{\boldsymbol{\alpha}}) - \min_{\boldsymbol{\alpha} \in \mathcal{A}} \mathcal{L}_{\mathcal{T}}(\boldsymbol{\alpha})$ obeys $R_{\mathcal{T}}(\widehat{\boldsymbol{\alpha}}) \leq \frac{R_{MTL}(\widehat{\boldsymbol{\alpha}})}{\nu} + 2\epsilon$.*

Here we emphasize that the statement holds for arbitrary source and target tasks; however the challenge is verifying the assumption which is left as an interesting and challenging future direction. On the bright side, as illustrated in Figures 9.4&9.5, we indeed observe that, transfer learning can work with reasonably small T and it works better if the target task is closer to the source tasks.

Proof. Let $\widehat{\boldsymbol{\alpha}}, \boldsymbol{\alpha}^\star$ be the empirical and population solutions of (ICL-ERM) and let $\boldsymbol{\alpha}^\dagger := \arg\min_{\boldsymbol{\alpha} \in \mathcal{A}} \mathcal{L}_{\mathcal{T}}(\boldsymbol{\alpha})$. Then the transfer learning excess test risk of given algorithm $\widehat{\boldsymbol{\alpha}} \in \mathcal{A}$ is formulated as

$$\begin{aligned} R_{\mathcal{T}}(\widehat{\boldsymbol{\alpha}}) &= \mathcal{L}_{\mathcal{T}}(\widehat{\boldsymbol{\alpha}}) - \mathcal{L}_{\mathcal{T}}(\boldsymbol{\alpha}^\dagger) \\ &= \mathcal{L}_{\mathcal{T}}(\widehat{\boldsymbol{\alpha}}) - \mathcal{L}_{\mathcal{T}}(\boldsymbol{\alpha}^\star) + \mathcal{L}_{\mathcal{T}}(\boldsymbol{\alpha}^\star) - \mathcal{L}_{\mathcal{T}}(\boldsymbol{\alpha}^\dagger). \end{aligned}$$

Since target task $\mathcal{T}$ is (ν, ϵ)-diverse over source tasks, following Definition H.1, we derive

that

$$\mathcal{L}_{\mathcal{T}}(\widehat{\boldsymbol{\alpha}}) - \mathcal{L}_{\mathcal{T}}(\boldsymbol{\alpha}^\star) \leq \frac{\mathcal{L}_{\mathrm{MTL}}(\widehat{\boldsymbol{\alpha}}) - \mathcal{L}_{\mathrm{MTL}}(\boldsymbol{\alpha}^\star)}{\nu} + \epsilon = \frac{R_{\mathrm{MTL}}(\widehat{\boldsymbol{\alpha}})}{\nu} + \epsilon$$
$$\mathcal{L}_{\mathcal{T}}(\boldsymbol{\alpha}^\star) - \mathcal{L}_{\mathcal{T}}(\boldsymbol{\alpha}^\dagger) \leq \frac{\mathcal{L}_{\mathrm{MTL}}(\boldsymbol{\alpha}^\star) - \mathcal{L}_{\mathrm{MTL}}(\boldsymbol{\alpha}^\dagger)}{\nu} + \epsilon \leq \epsilon.$$

Here, since $\boldsymbol{\alpha}^\star$ is the minimizer of $\mathcal{L}_{\mathrm{MTL}}(\boldsymbol{\alpha})$, $\mathcal{L}_{\mathrm{MTL}}(\boldsymbol{\alpha}^\star) - \mathcal{L}_{\mathrm{MTL}}(\boldsymbol{\alpha}^\dagger) \leq 0$. Then, Lemma H.4 is easily proved by combining the above two inequalities. ■

H.2.3 Transfer Learning Bound with i.i.d. Tasks

Following training with (ICL-ERM), suppose source tasks are i.i.d. sampled from a task distribution $\mathcal{D}_{\mathrm{task}}$, and let $\widehat{\boldsymbol{\alpha}}$ be the empirical MTL solution. We consider the following transfer learning problem. Concretely, assume a target task $\mathcal{T}$ with a distribution $\mathcal{T} \sim \mathcal{D}_{\mathrm{task}}$ and training sequence $\mathcal{S}_{\mathcal{T}} = (\boldsymbol{z}_i)_{i=1}^n \sim \mathcal{D}_{\mathcal{T}}$. Define the empirical and population risks on $\mathcal{T}$ as $\widehat{\mathcal{L}}_{\mathcal{T}}(\boldsymbol{\alpha}) = \frac{1}{n}\sum_{i=1}^n \ell(\boldsymbol{y}_i, f^{\boldsymbol{\alpha}}_{\mathcal{S}_{\mathcal{T}}^{(i-1)}}(\boldsymbol{x}_i))$ and $\mathcal{L}_{\mathcal{T}}(\boldsymbol{\alpha}) = \mathbb{E}_{\mathcal{S}_{\mathcal{T}}}[\widehat{\mathcal{L}}_{\mathcal{T}}(\boldsymbol{\alpha})]$. Then the expected excess transfer risk following (ICL-ERM) is defined as

$$\mathbb{E}_{\mathcal{T}}[\mathcal{R}_{\mathcal{T}}(\widehat{\boldsymbol{\alpha}})] = \mathbb{E}_{\mathcal{T}}[\mathcal{L}_{\mathcal{T}}(\widehat{\boldsymbol{\alpha}})] - \arg\min_{\boldsymbol{\alpha}\in\mathcal{A}} \mathbb{E}_{\mathcal{T}}[\mathcal{L}_{\mathcal{T}}(\boldsymbol{\alpha})]. \tag{H.19}$$

Theorem H.4 *Consider the setting of Theorem 9.2 and assume the source tasks are independently drawn from task distribution $\mathcal{D}_{task}$. Let $\widehat{\boldsymbol{\alpha}}$ be the empirical solution of* (ICL-ERM) *and $\mathcal{T} \sim \mathcal{D}_{task}$. Then with probability at least $1 - 2\delta$, the expected excess transfer learning risk* (H.19) *obeys*

$$\mathbb{E}_{\mathcal{T}}[\mathcal{R}_{\mathcal{T}}(\widehat{\boldsymbol{\alpha}})] \leq \min_{\epsilon\geq 0}\left\{4L\epsilon + B\sqrt{\frac{2\log(\mathcal{N}(\mathcal{A},\rho,\epsilon)/\delta)}{T}}\right\}. \tag{H.20}$$

Proof. Recap the problem setting in Section 9.2 and let $\boldsymbol{\alpha}^\dagger = \arg\min_{\boldsymbol{\alpha}\in\mathcal{A}} \mathbb{E}_{\mathcal{T}}[\mathcal{L}_{\mathcal{T}}(\boldsymbol{\alpha})]$. The expected transfer learning excess test risk of given algorithm $\widehat{\boldsymbol{\alpha}} \in \mathcal{A}$ is formulated as

$$\mathbb{E}_{\mathcal{T}}[\mathcal{R}_{\mathcal{T}}(\widehat{\boldsymbol{\alpha}})] \tag{H.21}$$
$$= \mathbb{E}_{\mathcal{T}}[\mathcal{L}_{\mathcal{T}}(\widehat{\boldsymbol{\alpha}})] - \mathbb{E}_{\mathcal{T}}[\mathcal{L}_{\mathcal{T}}(\boldsymbol{\alpha}^\dagger)] \tag{H.22}$$
$$= \underbrace{\mathbb{E}_{\mathcal{T}}[\mathcal{L}_{\mathcal{T}}(\widehat{\boldsymbol{\alpha}})] - \widehat{\mathcal{L}}_{\mathcal{S}_{\mathrm{all}}}(\widehat{\boldsymbol{\alpha}})}_{a} + \underbrace{\widehat{\mathcal{L}}_{\mathcal{S}_{\mathrm{all}}}(\widehat{\boldsymbol{\alpha}}) - \widehat{\mathcal{L}}_{\mathcal{S}_{\mathrm{all}}}(\boldsymbol{\alpha}^\dagger)}_{b} + \underbrace{\widehat{\mathcal{L}}_{\mathcal{S}_{\mathrm{all}}}(\boldsymbol{\alpha}^\dagger) - \mathbb{E}_{\mathcal{T}}[\mathcal{L}_{\mathcal{T}}(\boldsymbol{\alpha}^\dagger)]}_{c}. \tag{H.23}$$

Here since $\widehat{\boldsymbol{\alpha}}$ is the minimizer of training risk, $b < 0$. Then we obtain

$$\mathbb{E}_{\mathcal{T}}[\mathcal{R}_{\mathcal{T}}(\widehat{\boldsymbol{\alpha}})] \leq 2 \sup_{\boldsymbol{\alpha} \in \mathcal{A}} \left| \mathbb{E}_{\mathcal{T}}[\mathcal{L}_{\mathcal{T}}(\boldsymbol{\alpha})] - \frac{1}{T} \sum_{t=1}^{T} \widehat{\mathcal{L}}_t(\boldsymbol{\alpha}) \right|. \tag{H.24}$$

For any $\boldsymbol{\alpha} \in \mathcal{A}$, let $X_t = \widehat{\mathcal{L}}_t(\boldsymbol{\alpha})$ and we observe that

$$\mathbb{E}_{t \sim \mathcal{D}_{\text{task}}}[X_t] = \mathbb{E}_{t \sim \mathcal{D}_{\text{task}}}[\widehat{\mathcal{L}}_t(\boldsymbol{\alpha})] = \mathbb{E}_{t \sim \mathcal{D}_{\text{task}}}[\mathcal{L}_t(\boldsymbol{\alpha})] = \mathbb{E}_{\mathcal{T}}[\mathcal{L}_{\mathcal{T}}(\boldsymbol{\alpha})].$$

Since X_t, $t \in [T]$ are independent, and $0 \leq X_t \leq B$, applying Hoeffding's inequality obeys

$$\mathbb{P}\left(\left| \mathbb{E}_{\mathcal{T}}[\mathcal{L}_{\mathcal{T}}(\boldsymbol{\alpha})] - \frac{1}{T} \sum_{t=1}^{T} \widehat{\mathcal{L}}_t(\boldsymbol{\alpha}) \right| \geq \tau \right) \leq 2e^{-\frac{2T\tau^2}{B^2}}. \tag{H.25}$$

Then with probability at least $1 - 2\delta$, we have that for any $\boldsymbol{\alpha} \in \mathcal{A}$,

$$\left| \mathbb{E}_{\mathcal{T}}[\mathcal{L}_{\mathcal{T}}(\boldsymbol{\alpha})] - \frac{1}{T} \sum_{t=1}^{T} \widehat{\mathcal{L}}_t(\boldsymbol{\alpha}) \right| \leq B \sqrt{\frac{\log(1/\delta)}{2T}}. \tag{H.26}$$

Next, let $\mathcal{A}_\epsilon$ be the minimal ϵ-cover of $\mathcal{A}$ following Definition 9.1, which implies that for any task $\mathcal{T} \sim \mathcal{D}_{\text{task}}$, and any $\boldsymbol{\alpha} \in \mathcal{A}$, there exists $\boldsymbol{\alpha}' \in \mathcal{A}_\epsilon$

$$|\mathcal{L}_{\mathcal{T}}(\boldsymbol{\alpha}) - \mathcal{L}_{\mathcal{T}}(\boldsymbol{\alpha}')|, |\widehat{\mathcal{L}}_{\mathcal{T}}(\boldsymbol{\alpha}) - \widehat{\mathcal{L}}_{\mathcal{T}}(\boldsymbol{\alpha}')| \leq L\epsilon.$$

Since the distance metric following Definition 9.2 is defined by the worst-case datasets, then there exists $\boldsymbol{\alpha}' \in \mathcal{A}_\epsilon$ such that

$$\left| \mathbb{E}_{\mathcal{T}}[\mathcal{L}_{\mathcal{T}}(\boldsymbol{\alpha})] - \frac{1}{T} \sum_{t=1}^{T} \widehat{\mathcal{L}}_t(\boldsymbol{\alpha}) \right| \leq 2L\epsilon. \tag{H.27}$$

Let $\mathcal{N}(\mathcal{A}, \rho, \epsilon) = |\mathcal{A}_\epsilon|$ be the ϵ-covering number. Combining the above inequalities ((H.24), (H.26) and (H.27)), and applying union bound, we have that with probability at least $1 - 2\delta$,

$$\mathbb{E}_{\mathcal{T}}[\mathcal{R}_{\mathcal{T}}(\widehat{\boldsymbol{\alpha}})] \leq \min_{\epsilon \geq 0} \left\{ 4L\epsilon + B \sqrt{\frac{2 \log(\mathcal{N}(\mathcal{A}, \rho, \epsilon)/\delta)}{T}} \right\}.$$

■

H.2.4 Model Selection and Approximation Error Analysis

To proceed with our analysis, we need to make assumptions about what kind of algorithms are realizable by transformers. Given ERM is the work-horse of modern machine learning with general hypothesis classes, we assume that transformers can approximately perform in-context ERM. Hypothesis 9.1 states that the algorithms induced by the transformer can compete with empirical risk minimization over a family of hypothesis classes.

With this hypothesis, instead of searching over the entire hypothesis space $\mathcal{F}_{\text{all}} := \bigcup_{i=1}^{H} \mathcal{F}_i$, given prompt length m we search over the hypothesis space $\mathcal{F}_{h_m}$ only, and $\text{comp}(\mathcal{F}_m) \leq \text{comp}(\mathcal{F}_{\text{all}})$ where $\text{comp}(\cdot)$ captures the complexity of a hypothesis class.

In Hypothesis 9.1, we assume that $\mathbb{F}$ is a family of countable hypothesis classes with $|\boldsymbol{F}| = H$. As stated in Section 9.5, $\mathbb{F}$ is not necessary to be discrete. The following provides some examples of $\mathbb{F}$, where the first three correspond to discrete model selection whereas the left are continuous.

- $\mathbb{F}^{\text{sparse}} = \{\mathcal{F}_s : s\text{-sparse linear model}\}$,
- $\mathbb{F}^{\text{NN}} = \{\mathcal{F}_s : \text{2-layer neural net with width } s\}$,
- $\mathbb{F}^{\text{RF}} = \{\mathcal{F}_s : \text{Random forest with } s \text{ trees}\}$,
- $\mathbb{F}^{\text{ridge}} = \{\mathcal{F}_\lambda : \text{Linear model with parameter bounded by } \|\boldsymbol{\beta}\| \leq \lambda\}$ (akin to ridge regression)
- $\mathbb{F}^{\text{weighted}} = \{\mathcal{F}_{\boldsymbol{\Sigma}} : \text{Linear model with covariance-prior } \boldsymbol{\Sigma},\ \boldsymbol{\beta}^\top \boldsymbol{\Sigma}^{-1} \boldsymbol{\beta} \leq 1\}$ (akin to weighted ridge).

To proceed, let us introduce the following classical result that controls the test risk of an ERM solution in terms of the Rademacher complexity [133, 131].

Theorem H.5 *Let $\mathcal{F} : \mathcal{X} \to \mathcal{Y}$ be a hypothesis set and let $\mathcal{S} = (\boldsymbol{x}_i, \boldsymbol{y}_i)_{i=1}^n \in \mathcal{X} \times \mathcal{Y}$ be a dataset sampled i.i.d. from distribution $\mathcal{D}$. Let $\ell(\boldsymbol{y}, \hat{\boldsymbol{y}})$ be a loss function takes values in $[0, B]$. Here $\ell(\boldsymbol{y}, \cdot)$ is L-Lipschitz in terms of Euclidean norm for all $\boldsymbol{y} \in \mathcal{Y}$. Consider a learning problem that*

$$\hat{f} := \arg\min_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^{n} \ell(\boldsymbol{y}_i, f(\boldsymbol{x}_i)). \tag{H.28}$$

Let $\mathcal{L}^\star = \min_{f \in \mathcal{F}} \mathcal{L}(f)$ where $\mathcal{L}(f) = \mathbb{E}[\ell(\boldsymbol{y}, f(\boldsymbol{x}))]$. Then we have that with probability at

least $1-2\delta$*, the excess test risk obeys*

$$\mathcal{L}(\hat{f}) - \mathcal{L}^\star \le 8L\mathcal{R}_n(\mathcal{F}) + 4B\sqrt{\frac{\log\frac{1}{\delta}}{n}},$$

where $\mathcal{R}_n(\mathcal{F}) = \mathbb{E}_{\mathcal{S}}\mathbb{E}_{\boldsymbol{\sigma}_i}[\sup_{f\in\mathcal{F}}\frac{1}{n}\sum_{i=1}^n \boldsymbol{\sigma}_i^\top f(\boldsymbol{x}_i)]$ *is the Rademacher complexity of* $\mathcal{F}$ *[133] and* $\boldsymbol{\sigma}_i$*'s are vectors with Rademacher random variable in each entry.*

Lemma H.5 (Formal version of Observation 9.1) *Let* $\mathcal{L}_{\mathcal{T}}^\star := \min_{\boldsymbol{\alpha}\in\mathcal{A}}\mathcal{L}_{\mathcal{T}}(\boldsymbol{\alpha})$ *be the optimal target risk as stated in Section 9.2. Assume that Hypothesis 9.1 holds, then the approximation error obeys*

$$\mathcal{L}_{\mathcal{T}}^\star \le \frac{1}{n}\sum_{m=1}^{n-1}\min_{h\in[H]}\left\{\mathcal{L}_h^\star + 8L\mathcal{R}_m(\mathcal{F}_h) + \epsilon_{\mathit{TF}}^{h,m}\right\} + \frac{cB}{\sqrt{n}}, \tag{H.29}$$

where $\mathcal{R}_m(\mathcal{F})$ *is the Rademacher complexity over data distribution* $\mathcal{D}_{\mathcal{T}}$*, and* $\mathcal{L}_h^\star = \min_{f\in\mathcal{F}_h}\mathbb{E}[\ell(\boldsymbol{y}, f(\boldsymbol{x}))]$.

Proof. Let us assume Hypothesis 9.1 holds for algorithm $\widetilde{\texttt{Alg}} \in \mathcal{A}$. Since $\mathcal{L}_{\mathcal{T}}^\star$ is the minimal test loss, we have that

$$\mathcal{L}_{\mathcal{T}}^\star \le \mathcal{L}_{\mathcal{T}}(\widetilde{\texttt{Alg}}) = \mathbb{E}_{\mathcal{S}_{\mathcal{T}}}\left[\frac{1}{n}\sum_{i=1}^n \ell(\boldsymbol{y}_i, f^{\widetilde{\texttt{Alg}}}_{\mathcal{S}^{(i-1)}}(\boldsymbol{x}_i))\right] = \frac{1}{n}\sum_{i=1}^n \mathbb{E}_{(\boldsymbol{x},\boldsymbol{y},\mathcal{S}^{(i-1)})}\left[\ell(\boldsymbol{y}, f^{\widetilde{\texttt{Alg}}}_{\mathcal{S}^{(i-1)}}(\boldsymbol{x}))\right].$$

Then by directly applying Hypothesis 9.1 we have that

$$\mathcal{L}_{\mathcal{T}}^\star \le \frac{1}{n}\mathbb{E}_{(\boldsymbol{x},\boldsymbol{y})}\left[\ell(\boldsymbol{y}, f^{\widetilde{\texttt{Alg}}}_{\mathcal{S}^{(0)}}(\boldsymbol{x}))\right] + \frac{1}{n}\sum_{i=1}^{n-1}\mathbb{E}_{(\boldsymbol{x},\boldsymbol{y},\mathcal{S}^{(i)})}\left[\ell(\boldsymbol{y}, f^{\widetilde{\texttt{Alg}}}_{\mathcal{S}^{(i)}}(\boldsymbol{x}))\right] \tag{H.30}$$

$$\le \frac{B}{n} + \frac{1}{n}\sum_{m=1}^{n-1}\min_{h\in[H]}\left\{\mathrm{risk}(h,m) + \epsilon_{\texttt{TF}}^{h,m}\right\}. \tag{H.31}$$

Here the first term in (H.31) comes from the fact that loss function is bounded by B, and we assume $\mathcal{S}^{(0)} = \emptyset$, and the second term follows the Hypothesis 9.1. Next, we turn to bound $\mathrm{risk}(h,m)$. To proceed, let $X^{h,m} := \mathbb{E}_{(\boldsymbol{x},\boldsymbol{y})}\left[\ell(\boldsymbol{y}, \hat{f}^{(h)}_{\mathcal{S}^{(m)}}(\boldsymbol{x}))\right]$ be the random variables, where we have $|X^{h,m}| \le B$. Following Theorem H.5, we have that for any $m\in[n], h\in[H]$

$$\mathbb{P}\left(X^{h,m} - \mathcal{L}_h^\star - 8L\mathcal{R}_m(\mathcal{F}_h) \ge \tau\right) \le 2e^{-\frac{m\tau^2}{16B^2}}.$$

The upper-tail bound of the last line implies that there exists an absolute constant $c > 0$

such that

$$\operatorname{risk}(h,m) = \mathbb{E}_{\mathcal{S}^{(m)}}\left[X^{h,m}\right] \leq \mathcal{L}_h^\star + 8L\mathcal{R}_m(\mathcal{F}_h) + \frac{cB}{\sqrt{m}}.$$

Combining it with (H.31) and following the evidence $\sum_{m=1}^{n} \frac{1}{\sqrt{m}} \leq 2\sqrt{n}$ complete the proof. ■

H.3 Proof of Theorem 9.3

Lemma H.6 *Let $\mathcal{S}^{(m)} = (\boldsymbol{x}_i)_{i=1}^m$ be any dataset with $m \geq 1$ and $\mathcal{S}_j^{(m)}$ be the dataset such that starting with the j'th sample until the m'th sample from $\mathcal{S}^{(m)}$ is replaced by $(\boldsymbol{x}'_j, \boldsymbol{x}'_{j+1}, \ldots, \boldsymbol{x}'_m)$. Then, for any dataset $\mathcal{S}^{(m)}$, $(\boldsymbol{x}'_i)_{i=j}^m \in \mathcal{X}$, and $j \in [m]$, we have the following:*

$$\left|\mathbb{E}_{\boldsymbol{x}_{m+1}}\left[\ell(\boldsymbol{x}_{m+1}, f^{\boldsymbol{\alpha}}_{\mathcal{S}^{(m)}}(\boldsymbol{x}_m))\right] - \mathbb{E}_{\boldsymbol{x}_{m+1}}\left[\ell(\boldsymbol{x}_{m+1}, f^{\boldsymbol{\alpha}}_{\mathcal{S}_j^{(m)}}(\boldsymbol{x}_m))\right]\right| \leq \frac{K}{m}\sum_{i=j}^{m} \|\boldsymbol{x}_i - \boldsymbol{x}'_i\|_2 . \quad \text{(H.32)}$$

The proof of Lemma H.6 is obtained by applying Assumption 9.3 in a chained way.

Lemma H.7 *Suppose Assumptions 9.2 and 9.3 hold. Assume input and noise spaces $\mathcal{X}, \mathcal{W}$ are bounded by $\bar{x}, \bar{w}$. Let $W = (\boldsymbol{w}_1, \ldots, \boldsymbol{w}_j, \boldsymbol{w}_{j+1}, \ldots, \boldsymbol{w}_m)$ and $W' = (\boldsymbol{w}_1, \ldots, \boldsymbol{w}_{j-1}, \boldsymbol{w}'_j, \boldsymbol{w}_{j+1}, \ldots, \boldsymbol{w}_m)$ be two arbitrary sequences and the only difference between W and W' is the j'th term of the sequence. Then, for any $f \in \mathcal{F}$, $\boldsymbol{\alpha} \in \mathcal{A}$, W, W', m, and $j < m$, we have the following:*

$$\left|\mathbb{E}_{\boldsymbol{x}_{m+1}}\left[\ell(\boldsymbol{x}_{m+1}, f^{\boldsymbol{\alpha}}_{\mathcal{S}^{(m)}}(\boldsymbol{x}_m))\right] - \mathbb{E}_{\boldsymbol{x}_{m+1}}\left[\ell(\boldsymbol{x}_{m+1}, f^{\boldsymbol{\alpha}}_{\mathcal{S}_j^{(m)}}(\boldsymbol{x}'_m))\right]\right| < \frac{K}{m}\frac{2\bar{C}_\rho \bar{w}}{1-\bar{\rho}}.$$

Additionally, focusing on the sequences that differ only at their initial states, for any $\boldsymbol{x}_0, \boldsymbol{x}'_0 \in \mathcal{X}$, we have

$$\left|\mathbb{E}_{\boldsymbol{x}_{m+1}}\left[\ell(\boldsymbol{x}_{m+1}, f^{\boldsymbol{\alpha}}_{\mathcal{S}^{(m)}}(\boldsymbol{x}_m))\right] - \mathbb{E}_{\boldsymbol{x}_{m+1}}\left[\ell(\boldsymbol{x}_{m+1}, f^{\boldsymbol{\alpha}}_{\mathcal{S}_j^{(m)}}(\boldsymbol{x}'_m))\right]\right| < \frac{K}{m}\frac{2\bar{C}_\rho \bar{x}}{1-\bar{\rho}}.$$

Proof. First, let us bound $\|\boldsymbol{x}_i - \boldsymbol{x}'_i\|_2$ for every $i = j, \ldots, n$. For $i = j$, since $\mathcal{W}$ is bounded by $\bar{w}$, we have

$$\left\|\boldsymbol{x}_j - \boldsymbol{x}'_j\right\|_2 = \left\|f(\boldsymbol{x}_{j-1}) + \boldsymbol{w}_j - f(\boldsymbol{x}_{j-1}) - \boldsymbol{w}'_j\right\|_2 \leq 2\bar{w}.$$

For $i > j$, we have the following from Assumption 9.2:

$$\|\boldsymbol{x}_i - \boldsymbol{x}_i'\|_2 \leq \bar{C}_\rho \bar{\rho}^{(i-j)} \left\|\boldsymbol{x}_j - \boldsymbol{x}_j'\right\|_2 \leq 2\bar{C}_\rho \bar{\rho}^{(i-j)} \bar{w}.$$

Finally, using Assumption 9.3, we obtain

$$\begin{aligned}
&\left| \mathbb{E}_{(\boldsymbol{x}_{m+1})} \left[\ell(\boldsymbol{x}_{m+1}, f^{\boldsymbol{\alpha}}_{\mathcal{S}^{(m)}}(\boldsymbol{x}_m)) \right] - \mathbb{E}_{(\boldsymbol{x}_{m+1})} \left[\ell(\boldsymbol{x}_{m+1}, f^{\boldsymbol{\alpha}}_{\mathcal{S}_j^{(m)}}(\boldsymbol{x}'_m)) \right] \right| \\
&\qquad \leq \frac{K}{m} \sum_{i=j}^{m} \|\boldsymbol{x}_i - \boldsymbol{x}_i'\|_2 \\
&\qquad \leq \frac{K}{m} 2\bar{C}_\rho \bar{w} \sum_{i=j}^{m} \bar{\rho}^{i-j} < \frac{K}{m} \frac{2\bar{C}_\rho \bar{w}}{1-\bar{\rho}}.
\end{aligned}$$

To prove the second part of the lemma, similarly we have

$$\|\boldsymbol{x}_0 - \boldsymbol{x}_0'\|_2 \leq 2\bar{x} \quad \text{and then,} \quad \|\boldsymbol{x}_i - \boldsymbol{x}_i'\|_2 \leq 2\bar{C}_\rho \bar{\rho}^i \bar{x}.$$

Again using Assumption 9.3, we obtain

$$\begin{aligned}
&\left| \mathbb{E}_{(\boldsymbol{x}_{m+1})} \left[\ell(\boldsymbol{x}_{m+1}, f^{\boldsymbol{\alpha}}_{\mathcal{S}^{(m)}}(\boldsymbol{x}_m)) \right] - \mathbb{E}_{(\boldsymbol{x}_{m+1})} \left[\ell(\boldsymbol{x}_{m+1}, f^{\boldsymbol{\alpha}}_{\mathcal{S}_j^{(m)}}(\boldsymbol{x}'_m)) \right] \right| \\
&\qquad \leq \frac{K}{m} \sum_{i=0}^{m} \|\boldsymbol{x}_i - \boldsymbol{x}_i'\|_2 \\
&\qquad \leq \frac{K}{m} 2\bar{C}_\rho \bar{x} \sum_{i=0}^{m} \bar{\rho}^i < \frac{K}{m} \frac{2\bar{C}_\rho \bar{x}}{1-\bar{\rho}}.
\end{aligned}$$

■

Theorem H.6 (Theorem 9.3 restated) *Suppose Assumptions 9.2 and 9.3 hold and assume loss function $\ell(\boldsymbol{x}, \hat{\boldsymbol{x}}) : \mathcal{X} \times \mathcal{X} \to [0, B]$ is an L-Lipschitz function for all $\boldsymbol{x} \in \mathcal{X}$. Let $\widehat{\boldsymbol{\alpha}}$ be the empirical solutions of* (ICL-ERM) *under dynamical setting as described in Section 9.2&9.6. Then with probability at least $1 - 2\delta$, the excess MTL test risk* (MTL-risk) *obeys*

$$R_{MTL}(\widehat{\boldsymbol{\alpha}}) \leq \inf_{\epsilon > 0} \left\{ 4L\epsilon + 2(B + \bar{K} \log n) \sqrt{\frac{\log(\mathcal{N}(\mathcal{A}, \rho, \epsilon)/\delta)}{cnT}} \right\}.$$

where $\bar{K} = 2K \frac{\bar{C}_\rho}{1-\bar{\rho}} (\bar{w} + \bar{x}/\sqrt{n})$.

Proof. We follow the similar strategy as in the proof of Theorem 9.2. The main difference

is that we need to consider the dynamical system setting. Therefore, to start with, let us recap the dynamical problem setting in Sections 9.2&9.6. Suppose there are T independent trajectories generated by T dynamical systems, denoted by $\mathcal{S}_t = (\boldsymbol{x}_{t0}, \boldsymbol{x}_{t1}, \cdots, \boldsymbol{x}_{tn})$, $t \in [T]$ where $\boldsymbol{x}_{ti} = f_t(\boldsymbol{x}_{t,i-1}) + \boldsymbol{w}_{ti}$. Here, we consider the prediction function $f^{\boldsymbol{\alpha}}_{\mathcal{S}^{(i)}} : \mathcal{X} \to \mathcal{X}$, and denote the previously observed sequences with $\mathcal{S}^{(i)}_t := (\boldsymbol{x}_{t0}, \cdots, \boldsymbol{x}_{ti})$. The objective function in (ICL-ERM) can be rewritten as follows:

$$\widehat{\boldsymbol{\alpha}} = \arg\min_{\boldsymbol{\alpha}\in\mathcal{A}} \widehat{\mathcal{L}}_{\mathcal{S}_{\text{all}}}(\boldsymbol{\alpha}) := \frac{1}{T}\sum_{t=1}^{T} \widehat{\mathcal{L}}_t(\boldsymbol{\alpha}) \tag{H.33}$$
$$\text{where} \quad \widehat{\mathcal{L}}_t(\boldsymbol{\alpha}) = \frac{1}{n}\sum_{i=1}^{n} \ell(\boldsymbol{x}_{ti}, f^{\boldsymbol{\alpha}}_{\mathcal{S}^{(i-1)}_t}(\boldsymbol{x}_{t,i-1})).$$

Following the same argument as in the proof of Theorem 9.2, excess MTL risk is bounded by:

$$R_{\text{MTL}}(\widehat{\boldsymbol{\alpha}}) \le 2\sup_{\boldsymbol{\alpha}\in\mathcal{A}} |\mathcal{L}(\boldsymbol{\alpha}) - \widehat{\mathcal{L}}(\boldsymbol{\alpha})|.$$

Step 1: We start with the concentration bound $|\mathcal{L}(\boldsymbol{\alpha}) - \widehat{\mathcal{L}}(\boldsymbol{\alpha})|$ for any $\boldsymbol{\alpha} \in \mathcal{A}$. Define the random variables $X_{t,i} = \mathbb{E}[\widehat{\mathcal{L}}_t(\boldsymbol{\alpha})|\boldsymbol{x}_{t0}, (\boldsymbol{w}_{tk})^{i}_{k=1}]$ for $i \in [n]$ and $t \in [T]$, that is, $X_{t,i}$ is the expectation over $\widehat{\mathcal{L}}_t(\boldsymbol{\alpha})$ given the filtration of $\boldsymbol{x}_{t0}$ and $(\boldsymbol{w}_{tk})^{i}_{k=1}$. Then, we have that $X_{t,n} = \mathbb{E}[\widehat{\mathcal{L}}_t(\boldsymbol{\alpha})|\boldsymbol{x}_{t0}, (\boldsymbol{w}_{tk})^{n}_{k=1}] = \widehat{\mathcal{L}}_t(\boldsymbol{\alpha})$. Let $X_{t,0} = \mathbb{E}[\widehat{\mathcal{L}}_t(\boldsymbol{\alpha})]$. Then, for every t in $[T]$, the sequences $\{X_{t,0}, X_{t,1}, \ldots, X_{t,n}\}$ are Martingale sequences. Here we emphasize that $X_{t,0} = \mathbb{E}[X_{t,1}|\boldsymbol{x}_{t0}, \boldsymbol{w}_{t1}]$. For the sake of simplicity, in the following notation, we omit the subscript t for $\boldsymbol{x}, \boldsymbol{w}$ and $\mathcal{S}$, and look at the difference of neighbors for $1 < i < n$. Here, observe that "given $\mathcal{F}_i := \{\boldsymbol{x}_0, (\boldsymbol{w}_k)^{i}_{k=1}\}$" implies $\{\boldsymbol{x}_0, \cdots, \boldsymbol{x}_i\}$ are known with respect to this filtration.

$$\begin{aligned}
&|X_{t,i} - X_{t,i-1}| \qquad\qquad (\text{H.34})\\
&= \left|\mathbb{E}\left[\frac{1}{n}\sum_{j=1}^{n} \ell(\boldsymbol{x}_j, f^{\boldsymbol{\alpha}}_{\mathcal{S}^{(j-1)}}(\boldsymbol{x}_{j-1}))\middle|\boldsymbol{x}_0, (\boldsymbol{w}_k)^{i}_{k=1}\right] - \mathbb{E}\left[\frac{1}{n}\sum_{j=1}^{n} \ell(\boldsymbol{x}_j, f^{\boldsymbol{\alpha}}_{\mathcal{S}^{(j-1)}}(\boldsymbol{x}_{j-1}))\middle|\boldsymbol{x}_0, (\boldsymbol{w}_k)^{i-1}_{k=1}\right]\right|\\
&\le \frac{1}{n}\sum_{j=i}^{n}\left|\mathbb{E}\left[\ell(\boldsymbol{x}_j, f^{\boldsymbol{\alpha}}_{\mathcal{S}^{(j-1)}}(\boldsymbol{x}_{j-1}))\middle|\boldsymbol{x}_0, (\boldsymbol{w}_k)^{i}_{k=1}\right] - \mathbb{E}\left[\ell(\boldsymbol{x}_j, f^{\boldsymbol{\alpha}}_{\mathcal{S}^{(j-1)}}(\boldsymbol{x}_{j-1}))\middle|\boldsymbol{x}_0, (\boldsymbol{w}_k)^{i-1}_{k=1}\right]\right|\\
&\overset{(a)}{\le} \frac{B}{n} + \frac{1}{n}\sum_{j=i+1}^{n}\left|\mathbb{E}\left[\ell(\boldsymbol{x}_j, f^{\boldsymbol{\alpha}}_{\mathcal{S}^{(j-1)}}(\boldsymbol{x}_{j-1}))\middle|\boldsymbol{x}_0, (\boldsymbol{w}_k)^{i}_{k=1}\right] - \mathbb{E}\left[\ell(\boldsymbol{x}_j, f^{\boldsymbol{\alpha}}_{\mathcal{S}^{(j-1)}}(\boldsymbol{x}_{j-1}))\middle|\boldsymbol{x}_0, (\boldsymbol{w}_k)^{i-1}_{k=1}\right]\right|.
\end{aligned}$$

Here, (a) follows from the fact that loss function $\ell(\cdot, \cdot)$ is bounded by B. To proceed, call the right side terms $D_j := \big|\mathbb{E}[\ell(\boldsymbol{x}_j, f^{\boldsymbol{\alpha}}_{\mathcal{S}^{(j-1)}}(\boldsymbol{x}_{j-1}))\big|\boldsymbol{x}_0, (\boldsymbol{w}_k)^{i}_{k=1}] -$

$\mathbb{E}[\ell(\boldsymbol{x}_j, f^{\boldsymbol{\alpha}}_{\mathcal{S}^{(j-1)}}(\boldsymbol{x}_{j-1}))\big|\boldsymbol{x}_0, (\boldsymbol{w}_k)_{k=1}^{i-1}]|$. For an arbitrary sequence of $(\boldsymbol{w}_k^{'})_{k=i+1}^{n}$, we obtain using the first part of Lemma H.7 that

$$\Big|\mathbb{E}[\ell(\boldsymbol{x}_j, f^{\boldsymbol{\alpha}}_{\mathcal{S}^{(j-1)}}(\boldsymbol{x}_{j-1}))\big|\boldsymbol{x}_0, (\boldsymbol{w}_k)_{k=1}^{i}, (\boldsymbol{w}_k)_{k=i+1}^{n} = (\boldsymbol{w}_k^{'})_{k=i+1}^{n}] - \mathbb{E}[\ell(\boldsymbol{x}_j, f^{\boldsymbol{\alpha}}_{\mathcal{S}^{(j-1)}}(\boldsymbol{x}_{j-1}))\big|\boldsymbol{x}_0, (\boldsymbol{w}_k)_{k=1}^{i-1}, (\boldsymbol{w}_k)_{k=i+1}^{n} = (\boldsymbol{w}_k^{'})_{k=i+1}^{n}]\Big| \leq \frac{K}{j-1}\frac{2\bar{C}_\rho \bar{w}}{1-\bar{\rho}}.$$

Since that is valid for every realization of $(\boldsymbol{w}_k)_{k=i+1}^{n}$, we have

$$D_j \leq \frac{K}{j-1}\frac{2\bar{C}_\rho \bar{w}}{1-\bar{\rho}}.$$

Combining above, for any $n > i > 1$, we obtain

$$|X_{t,i} - X_{t,i-1}| \leq \frac{B}{n} + \frac{1}{n}\sum_{j=i+1}^{n}\frac{K}{j-1}\frac{2\bar{C}_\rho \bar{w}}{1-\bar{\rho}} < \frac{B}{n} + \frac{K\log n}{n}\frac{2\bar{C}_\rho \bar{w}}{1-\bar{\rho}}.$$

If we apply the same step above and use the second part of Lemma H.7, we obtain the following bound for $|X_{t,1} - X_{t,0}|$:

$$|X_{t,1} - X_{t,0}| < \frac{B}{n} + \frac{K\log n}{n}\frac{2\bar{C}_\rho (\bar{w} + \bar{x})}{1-\bar{\rho}}.$$

Moreover, as the loss function is bounded by B, we have

$$|X_{t,n} - X_{t,n-1}| \leq \frac{B}{n} < \frac{B}{n} + \frac{K\log n}{n}\frac{2\bar{C}_\rho \bar{w}}{1-\bar{\rho}}.$$

Note that $|\mathcal{L}_t(\boldsymbol{\alpha}) - \widehat{\mathcal{L}}_t(\boldsymbol{\alpha})| = |X_{t,0} - X_{t,n}|$ and for every $t \in [T]$, we obtain

$$\sum_{i=1}^{n}|X_{t,i} - X_{t,i-1}|^2 \leq \frac{(n-1)\left(B + K\log n\frac{2\bar{C}_\rho \bar{w}}{1-\bar{\rho}}\right)^2 + \left(B + K\log n\frac{2\bar{C}_\rho (\bar{w}+\bar{x})}{1-\bar{\rho}}\right)^2}{n^2} \leq \frac{\left(B + K\log n\frac{2\bar{C}_\rho (\bar{w}+\bar{x}/\sqrt{n})}{1-\bar{\rho}}\right)^2}{n}.$$

Then the equivalent result in the Step 1 of the proof of Theorem 9.2 is obtained by updating K to be $\bar{K} = 2K\frac{\bar{C}_\rho}{1-\bar{\rho}}(\bar{w} + \bar{x}/\sqrt{n})$.

Step 2: Next, we turn to bound $\sup_{\boldsymbol{\alpha}\in\mathcal{A}}|\mathcal{L}(\boldsymbol{\alpha}) - \widehat{\mathcal{L}}(\boldsymbol{\alpha})|$ where $\mathcal{A}$ is assumed to be a continuous search space. Following the identical analysis as in Step 2 of proof of

Theorem 9.2, by applying a covering argument, we conclude with the the result. ■

H.4 Additional Experiments for Section 9.4

In Section 9.4, we discussed how transfer risk seems to require $T \propto d^2$ source tasks. In contrast, constructing the empirical covariance $\hat{\mathbf{\Sigma}} = \frac{1}{T}\sum_{i=1}^{T} \boldsymbol{\beta}_i \boldsymbol{\beta}_i^\top$ can make sure that $\hat{\mathbf{\Sigma}}$-weighted LS performs $\mathcal{O}(1)$-close to $\mathbf{\Sigma}$-weighted LS whenever $\|\mathbf{\Sigma} - \hat{\mathbf{\Sigma}}\|/\lambda_{\min}(\mathbf{\Sigma}) \leq \mathcal{O}(1)$. In Figure H.1, *MTL-ridge* curves with circle markers are referring to the $\hat{\mathbf{\Sigma}}$-weighted ridge regression. As suspected, $T = 3d$ is already sufficient to get very close performance to the optimal weighting with true $\mathbf{\Sigma}$ (black curve). We remark that in Figure H.1 $d = 20$, noise variance obeys $\sigma^2 = 0.1$, and linear task vectors $\boldsymbol{\beta}$ are uniformly sampled over the sphere.

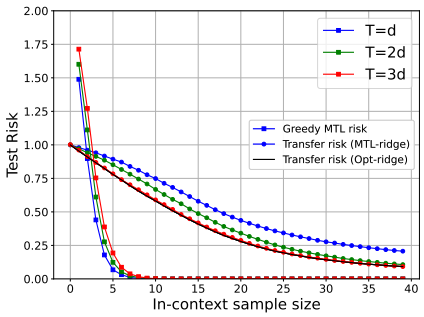


Figure H.1: We display the performance of the *idealized* transfer and MTL algorithms described in Section 9.4. Unlike ICL experiments, these require $T \lesssim d$ tasks.

For MTL, Section 9.4 introduces the following simple greedy algorithm to predict a prompt that belong to one of the T source tasks (aka MTL risk): Evaluate each source task parameter $(\boldsymbol{\beta}_t)_{t=1}^T$ on the prompt and select the parameter with the minimum risk. Since there are T choices, this greedy algorithm will determine the optimal task in $n \lesssim \log(T)$ samples[1]. The experiments of this algorithm is provided under *Greedy MTL* legend (square markers). It can be seen that as T varies $(d, 2d, 3d)$, there is almost no difference in the MTL risk, likely due to the $\log(T)$ dependence. In short, these experiments highlight the contrast between ideal/greedy algorithms and ICL algorithm implemented within the transformer model.

In Section 9.4, we also exclusively focused on the setting $M \to \infty$ i.e. MTL tasks are thoroughly trained. In Figure H.2 we consider the other extreme where each task is trained with a single trajectory $M = 1$, which is closer to the spirit of Theorem 9.2. We set $d = 5$, $n = 10$ and $M = 1$, and vary the number of linear regression tasks T from 2000 to 50000. Not surprisingly, the results show that increasing T helps in reducing the MTL risk. The more interesting observation is that transfer risk and MTL

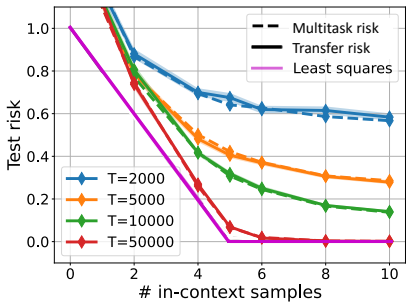


Figure H.2: Comparing MTL and transfer risks when each task has single trajectory ($M = 1$).

[1]Note that, this dependence can be even better if the problem is noiseless, in fact, that is why we added label noise in these experiments.

risk are almost perfectly aligned. We believe that this is due to the small M, n choices which would make it difficult to overfit to the MTL tasks. Thus, when $M = 1$, the gap between transfer and MTL risk seems to vanish and Theorem 9.2 becomes directly informative for the transfer risk. In contrast, as M grows, training process can overfit to the MTL tasks which leads to the split between MTL and transfer risks as in Figure 9.4.

APPENDIX I

Appendix for Chapter 10

I.1 Construction

I.1.1 The Transformer Architecture

For the purpose of this proof, we consider encoder-based transformer architectures and assume that the positional encodings are appended to the input[1]. We also consider that the heads are added and each one has each own key, query and value weight matrices. Formally, we have

$$\text{att}(\boldsymbol{X}) = \boldsymbol{X} + \sum_{h=1}^{H} \boldsymbol{W}_v^h \boldsymbol{X} \mathbb{S}((\boldsymbol{W}_k^h \boldsymbol{X})^\top \boldsymbol{W}_q^h \boldsymbol{X}) \tag{I.1}$$

$$\texttt{TF}(\boldsymbol{X}) = \text{att}(\boldsymbol{X}) + \boldsymbol{W}_2(\boldsymbol{W}_1 \text{att}(\boldsymbol{X}) + \boldsymbol{v}_1 \boldsymbol{1}_n)_+ + \boldsymbol{v}_2 \boldsymbol{1}_n \tag{I.2}$$

where $\boldsymbol{X} \in \mathbb{R}^{d \times n}$, H is the number of heads used and $f(x) = (x)_+$ is the ReLU activation. We also make use of the temperature λ, which is a parameter of the softmax. Specifically, $\mathbb{S}(\boldsymbol{x}) = \{\mathrm{e}^{\lambda x_i} / \sum_j \mathrm{e}^{\lambda x_j}\}_i$. Notice that as $\lambda \to \infty$ we have $\mathbb{S}(\boldsymbol{x}) \to \max_i x_i$. We also assume that the inputs are bounded; we denote with N_{max} the maximum sequence length of the model.

Assumption I.1 *Each entry is bounded by some constant c, which implies that $\|\boldsymbol{X}\| \leq c'$, for some large c' that depends on the maximum sequence length and the width of the transformer.*

[1]We note here that in terms of our construction adding the encodings or appending them to the input can be viewed in a similar manner. Specifically, we can consider that the up-projection step projects the input to zero padded vectors, while the encodings are orthogonal to that projection in the sense that the have zero in the non-zero coordinates of the input. In that case adding the positional encodings corresponds to appending them to the input.

I.1.2 Positional Encodings

In the constructions below we use a combination of the binary representation of each position, as well as some additional information (e.g. 0-1 bits) as described in the following sections. The binary representations and in general encodings we construct, require only logarithmic space with respect to the sequence length. Notice that binary representation of the positions satisfy the following two conditions:

1. Let $\boldsymbol{r}_i$ be the binary representation of the i-th position, then there exists $\varepsilon > 0$ such that $\boldsymbol{r}_i^\top \boldsymbol{r}_i > \boldsymbol{r}_i^\top \boldsymbol{r}_j + \varepsilon$ for all $j \neq i$.

2. There exists a one layer transformer that can implement the addition of two pointers (see Lemma I.2).

I.1.3 Constructing Some Useful "Black-box" Functions

We follow the construction of previous work [62] and use the following individual implementations as they do. We repeat the statements here for convenience. The first lemma is also similar to [4], we however follow a slightly different proof.

Lemma I.1 *A transformer with one layer, two heads, and embedding dimension of $\mathcal{O}(\log n + d)$, where d is the input dimension and n is the sequence length, can copy any block of the input to any other block of the input with error ε arbitrarily small.*

Proof. Consider an input of the following form

$$\boldsymbol{X} = \begin{bmatrix} \boldsymbol{x}_1 & \boldsymbol{x}_2 & \dots & \boldsymbol{x}_n \\ \boldsymbol{y}_1 & \boldsymbol{y}_2 & \dots & \boldsymbol{y}_n \\ \boldsymbol{r}_1 & \boldsymbol{r}_2 & \dots & \boldsymbol{r}_n \\ \boldsymbol{r}_k & \boldsymbol{r}_2 & \dots & \boldsymbol{r}_n \end{bmatrix} \tag{I.3}$$

where $\boldsymbol{x}_i, \boldsymbol{y}_i \in \mathbb{R}^d$ are column vectors for all $i = 1, \dots, n$. We present how we can move a block of data from the $(d+1 : 2d, k)$ position block to the $(1 : d, 1)$ position block, meaning to move the point $\boldsymbol{y}_k$ to the position of the point $\boldsymbol{x}_1$. It is straightforward to generalize the proof that follows to move blocks from $(i : i+k, j)$ to $(i' : i'+k, j')$ for arbitrary i, i', j, j', k.

Let in Eq. I.1 $\boldsymbol{W}_k^1 = \begin{bmatrix} \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{I} & \boldsymbol{0} \end{bmatrix}$ and $\boldsymbol{W}_q^1 = \begin{bmatrix} \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{I} \end{bmatrix}$ and $\boldsymbol{W}_k^2 = \begin{bmatrix} \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{I} \end{bmatrix}$

and $\boldsymbol{W}_q^2 = \begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{0} & \boldsymbol{I} \end{bmatrix}$ then we have

$$(\boldsymbol{W}_k^1 \boldsymbol{X})^\top (\boldsymbol{W}_q^1 \boldsymbol{X}) = \begin{bmatrix} \boldsymbol{r}_1^\top \boldsymbol{r}_k & \boldsymbol{r}_1^\top \boldsymbol{r}_2 & \dots & \boldsymbol{r}_1^\top \boldsymbol{r}_k & \dots & \boldsymbol{r}_1^\top \boldsymbol{r}_n \\ \boldsymbol{r}_2^\top \boldsymbol{r}_k & \boldsymbol{r}_2^\top \boldsymbol{r}_2 & \dots & \boldsymbol{r}_2^\top \boldsymbol{r}_k & \dots & \boldsymbol{r}_2^\top \boldsymbol{r}_n \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ \boldsymbol{r}_k^\top \boldsymbol{r}_k & \boldsymbol{r}_k^\top \boldsymbol{r}_2 & \dots & \boldsymbol{r}_k^\top \boldsymbol{r}_k & \dots & \boldsymbol{r}_k^\top \boldsymbol{r}_n \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ \boldsymbol{r}_n^\top \boldsymbol{r}_k & \boldsymbol{r}_n^\top \boldsymbol{r}_2 & \dots & \boldsymbol{r}_n^\top \boldsymbol{r}_k & \dots & \boldsymbol{r}_n^\top \boldsymbol{r}_n \end{bmatrix} \tag{I.4}$$

As $\lambda \to \infty$ and after the application of the softmax operator the above matrix becomes equal to $\begin{bmatrix} \mathrm{e}_k & \mathrm{e}_2 & \dots & \mathrm{e}_k & \dots \mathrm{e}_n \end{bmatrix} + \epsilon \boldsymbol{M}$, where e_i is the one-hot vector with 1 in the k-th position, $\|M\| \leq 1$ and ϵ is controllable by the temperature parameter and can be arbitrary small. Let finally $\boldsymbol{W}_v^1 = \begin{bmatrix} \mathbf{0} & \boldsymbol{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix}$ we get that

$$\boldsymbol{W}_v^1 \boldsymbol{X} \mathbb{S}((\boldsymbol{W}_k^1 \boldsymbol{X})^\top (\boldsymbol{W}_q^1 \boldsymbol{X})) = \begin{bmatrix} \boldsymbol{y}_k & \boldsymbol{x}_2 & \dots & \boldsymbol{x}_n \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} \end{bmatrix} + \epsilon \boldsymbol{M} \tag{I.5}$$

By repeating the exact same steps for the second head and letting $\boldsymbol{W}_v^2 = \begin{bmatrix} -\boldsymbol{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix}$ and adding back the residual we get the desired result. ■

A slightly different implementation can be found in [62], in which the ReLU layers are also used, together with indicator vectors that define whether a position should be updated or not.

Lemma I.2 *There exists a 1-hidden layer feedforward, ReLU network, with $8d$ activations in the hidden layer and d neurons in the output layer that when given two d-dimensional binary vectors representing two non-negative integers, can output the binary vector representation of their sum, as long as the sum is less than 2^{d+1}.*

Lemma I.3 *Let $\boldsymbol{A} \in \mathbb{R}^{d \times m}$ and $\boldsymbol{B} \in \mathbb{R}^{d \times n}$. Then for any $\epsilon > 0$ there exists a transformer-based function block with 2 layers, 1 head and width $r = O(d)$ that outputs the multiplication $\boldsymbol{A}^\top \boldsymbol{B} + \epsilon \boldsymbol{M}$, for some $\|\boldsymbol{M}\| \leq 1$.*

Remark I.1 *Notice that based on the proof of this lemma, the matrices/scalars/vectors need to be in the same rows,* i.e., *$\boldsymbol{Q} = \begin{bmatrix} \boldsymbol{A} & \boldsymbol{B} \end{bmatrix}$. By also appending the appropriate binary*

encodings we can move the output at any specific place we choose as in Lemma I.1. Also, following the proof of the paper the input matrix is

$$\boldsymbol{X} = \begin{bmatrix} \boldsymbol{Q} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{1}\mathbf{1}^\top & \mathbf{0} \\ \boldsymbol{I} & \mathbf{0} & \mathbf{0} \\ & \boldsymbol{r}^{(1)} & \\ & \boldsymbol{r}^{(2)} & \end{bmatrix}$$

where $\boldsymbol{r}^{(1)}, \boldsymbol{r}^{(2)}$ are chosen as to specify the position of the result.

I.1.4 Results on Filtering

Lemma I.4 *Assume that the input to a transformer layer is of the following form*

$$\boldsymbol{X} = \begin{bmatrix} \boldsymbol{x}_1 & \boldsymbol{x}_2 & \dots & \boldsymbol{x}_{n-1} & \boldsymbol{x}_n \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} \\ b_1 & b_2 & \dots & b_{n-1} & b_n \\ b_1' & b_2' & \dots & b_{n-1}' & b_n \end{bmatrix} \tag{I.6}$$

where $b_i, b_i' \in \{0, 1\}$, with zero indicating that the corresponding point should be ignored. Then there exists a transformer $\mathtt{TF}$ consisting only of a ReLU layer that performs this filtering, i.e.,

$$\mathtt{TF}(\boldsymbol{X}) = \begin{bmatrix} \mathbf{1}\{b_1 \neq 0\}\boldsymbol{x}_1 & \mathbf{1}\{b_2 \neq 0\}\boldsymbol{x}_2 & \dots & \mathbf{1}\{b_{n-1} \neq 0\}\boldsymbol{x}_{n-1} & \mathbf{1}\{b_n \neq 0\}\boldsymbol{x}_n \\ \mathbf{1}\{b_1' \neq 0\}\boldsymbol{x}_1 & \mathbf{1}\{b_2' \neq 0\}\boldsymbol{x}_2 & \dots & \mathbf{1}\{b_{n-1}' \neq 0\}\boldsymbol{x}_{n-1} & \mathbf{1}\{b_n' \neq 0\}\boldsymbol{x}_n \\ b_1 & b_2 & \dots & b_{n-1} & b_n \\ b_1' & b_2' & \dots & b_{n-1}' & b_n \end{bmatrix} \tag{I.7}$$

Proof. The layer is the following:

$$\mathtt{TF}(\boldsymbol{x}_i) = x_i + (-Cb_i - \boldsymbol{x}_i)_+ - (-Cb_i + \boldsymbol{x}_i)_+ \tag{I.8}$$

for some large constant C. Notice that if $b_i = 1$ the output is just $\boldsymbol{x}_i$. But if $b_i = 0$ then the output is zero. For the second set instead of using the bits b_i, we use the b_i'. ∎

Remark I.2 *Notice that if some b_i is instead of 1, $1 \pm \varepsilon$, $\varepsilon < c/C$. Then the output of the*

above layer would be

$$\mathit{TF}(\boldsymbol{x}_i) = \boldsymbol{x}_i + (-C \pm C\varepsilon - \boldsymbol{x}_i) - (-C \pm C\varepsilon + \boldsymbol{x}_i) \tag{I.9}$$

$$= \boldsymbol{x}_i \tag{I.10}$$

while if some $b_i = \pm\varepsilon$ *instead of zero, and assuming that* $\boldsymbol{x}_i > c > 0$ *or* $\boldsymbol{x}_i < -c < 0$ *the output would be*

$$\mathit{TF}(\boldsymbol{x}_i) = \boldsymbol{x}_i + (-C\varepsilon - \boldsymbol{x}_i)_+ - (-Cb_i + \boldsymbol{x}_i)_+ \tag{I.11}$$

$$= \boldsymbol{x}_i \pm C\varepsilon - \boldsymbol{x}_i \tag{I.12}$$

$$\leq c \tag{I.13}$$

If $|\boldsymbol{x}_i| \leq c$*, again the output would be less than or equal to* c*, where* c *can be arbitrarily small.*

Our target is to create these binary tokens, as to perform the filtering. We now describe for clarity how next word prediction is performed. An $d \times N$-dimensional input is given to the transformer architecture, which is up-projected using the embedding layer; the positional encodings are also added/appended to the input. At the last layer, the last token predicted is appended to the initial input to the transformer architecture. The only difference of the new input including the positional encodings (the input for the next iteration) is the $n+1$-th token. This is a property that our construction maintains, *i.e.,* the positional encodings used are oblivious to the prediction step performed and are always the same for each individual token. We consider encoder-based architectures in all of our lemmas below.

In the subsequent lemma we construct an automated process that works along those guidelines. To do so we assume that the input to the transformer contains the following information:

1. An enumeration of the tokens from 1 to N.
2. The ln of the above enumeration.
3. Zeros for the tokens that correspond to the data points, 1 for each token that is the $\boldsymbol{x}_{\text{test}}$ or it is a prediction based on it.
4. An enumeration 1 to L for each one of the data points provided. For example, if we are given three sets of data we would have : $1 \dots L\ 1 \dots L\ 1 \dots L$.
5. Some extra information that is needed to implement a multiplication step as described in Lemma I.3 and to move things to the correct place.

The above information can be viewed as part of the encodings that are appended to the input of the transformer. Formally, we have

Lemma I.5 *Consider that a prompt with n in-context samples is given and the $\ell-1$-th prediction has been made, and the transformer is to predict the ℓ-th one. Assume the input to the transformer is:*

$$\boldsymbol{X} = \left[\begin{array}{cccccccccccc}
\boldsymbol{x}_1 & \dots & \boldsymbol{s}_1^{\ell-1} & \boldsymbol{s}_1^{\ell} & \dots & \boldsymbol{s}_2^{\ell-1} & \boldsymbol{s}_2^{\ell} & \dots & \boldsymbol{s}_n^{L} & \boldsymbol{x}_{test} & \dots & \hat{\boldsymbol{s}}^{\ell-1} \\
\mathbf{0} & \dots & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} \\
\hline
1 & \dots & \ell & \ell+1 & \dots & \ell & \ell+1 & \dots & L+1 & 1 & \dots & \ell \\
1 & \dots & \ell & \ell+1 & \dots & \ell & \ell+1 & \dots & L+1 & 1 & \dots & \ell \\
\hline
1 & \dots & \ell & \ell+1 & \dots & L+\ell+1 & L+\ell+2 & \dots & n(L+1) & n(L+1)+1 & \dots & N \\
\ln(1) & \dots & \ln(\ell) & \ln(\ell+1) & \dots & \ln(L+\ell+1) & \ln(L+\ell+2) & \dots & \ln(n(L+1)) & \ln(n(L+1)+1) & \dots & \ln(N) \\
0 & \dots & 0 & 0 & \dots & 0 & 0 & \dots & 0 & 1 & \dots & 1 \\
1 & \dots & 1 & 1 & \dots & 1 & 1 & \dots & 1 & 1 & \dots & 1 \\
\boldsymbol{m}_1 & \dots & \boldsymbol{m}_\ell & \boldsymbol{m}_{\ell+1} & \dots & \boldsymbol{m}_{L+\ell+1} & \boldsymbol{m}_{L+\ell+2} & \dots & \boldsymbol{m}_{n(L+1)} & \boldsymbol{m}_{n(L+1)+1} & \dots & \boldsymbol{m}_N
\end{array}\right]$$

where $(\hat{\boldsymbol{s}}^i)_{i=1}^{\ell-1}$ denote the first $\ell-1$ recurrent outputs of $\texttt{TF}$ and for simplicity, let $N := n(L+1)+\ell$ denote the total number of tokens. Then there exists a transformer $\texttt{TF}$ consisting of 7 layers that has as output

$$\texttt{TF}(\boldsymbol{X}) = \left[\begin{array}{cccccccccccc}
\mathbf{0} & \dots & \boldsymbol{s}_1^{\ell-1} & \mathbf{0} & \dots & \boldsymbol{s}_2^{\ell-1} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} & \dots & \hat{\boldsymbol{s}}^{\ell-1} \\
\mathbf{0} & \dots & \mathbf{0} & \boldsymbol{s}_1^{\ell} & \dots & \mathbf{0} & \boldsymbol{s}_2^{\ell} & \dots & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} \\
\hline
0 & \dots & 1 & 0 & \dots & 1 & 0 & \dots & 0 & 0 & \dots & 1 \\
0 & \dots & 0 & 1 & \dots & 0 & 1 & \dots & 0 & 0 & \dots & 0 \\
\hline
1 & \dots & \ell & \ell+1 & \dots & L+\ell+1 & L+\ell+2 & \dots & n(L+1) & n(L+1)+1 & \dots & N \\
\ln(1) & \dots & \ln(\ell) & \ln(\ell+1) & \dots & \ln(L+\ell+1) & \ln(L+\ell+2) & \dots & \ln(n(L+1)) & \ln(n(L+1)+1) & \dots & \ln(N) \\
0 & \dots & 0 & 0 & \dots & 0 & 0 & \dots & 0 & 1 & \dots & 1 \\
1 & \dots & 1 & 1 & \dots & 1 & 1 & \dots & 1 & 1 & \dots & 1 \\
\boldsymbol{m}_1 & \dots & \boldsymbol{m}_\ell & \boldsymbol{m}_{\ell+1} & \dots & \boldsymbol{m}_{L+\ell+1} & \boldsymbol{m}_{L+\ell+2} & \dots & \boldsymbol{m}_{n(L+1)} & \boldsymbol{m}_{n(L+1)+1} & \dots & \boldsymbol{m}_N
\end{array}\right]$$

with error up to $\delta\boldsymbol{M}$, where $\|\boldsymbol{M}\| \leq 1$ and $\delta > 0$ is a constant that is controlled and can be

arbitrarily small.

Proof.

Step 1: Extract the sequence length (1 layer). Let

$$\boldsymbol{W}_k = \begin{bmatrix} \mathbf{0} & \mathbf{0} & 0 & 0 & 0 & 1 & 0 & 0 & \mathbf{0} \end{bmatrix} \quad \boldsymbol{W}_q = \begin{bmatrix} \mathbf{0} & \mathbf{0} & 0 & 0 & 0 & 0 & 0 & 0 & 1 & \mathbf{0} \end{bmatrix} \tag{I.14}$$

and thus

$$(\boldsymbol{W}_k \boldsymbol{X})^\top (\boldsymbol{W}_q \boldsymbol{X}) = \begin{bmatrix} \ln(1) & \ln(1) & \dots & \ln(1) \\ \ln(2) & \ln(2) & \dots & \ln(2) \\ \vdots & \vdots & \ddots & \vdots \\ \ln(\mathrm{N}) & \ln(\mathrm{N}) & \dots & \ln(\mathrm{N}) \end{bmatrix}. \tag{I.15}$$

So after the softmax is applied we have

$$\mathbb{S}((\boldsymbol{W}_k \boldsymbol{X})^\top (\boldsymbol{W}_q \boldsymbol{X})) = \begin{bmatrix} \frac{1}{\sum_{i=1}^{\mathrm{N}} i} & \frac{1}{\sum_{i=1}^{\mathrm{N}} i} & \cdots & \frac{1}{\sum_{i=1}^{\mathrm{N}} i} \\ \frac{2}{\sum_{i=1}^{\mathrm{N}} i} & \frac{2}{\sum_{i=1}^{\mathrm{N}} i} & \cdots & \frac{2}{\sum_{i=1}^{\mathrm{N}} i} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\mathrm{N}}{\sum_{i=1}^{\mathrm{N}} i} & \frac{\mathrm{N}}{\sum_{i=1}^{\mathrm{N}} i} & \cdots & \frac{\mathrm{N}}{\sum_{i=1}^{\mathrm{N}} i} \end{bmatrix}. \tag{I.16}$$

We then set the weight value matrix as to zero-out all lines except for one line as follows

$$\boldsymbol{W}_v \boldsymbol{X} = \begin{bmatrix} \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} \\ 1 & 2 & \dots & \mathrm{N} \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} \end{bmatrix}. \tag{I.17}$$

After adding the residual and using an extra head where the softmax returns identity matrix

and the value weight matrix is minus the identity, we get attention output

$$\left[\begin{array}{ccccccccccccc}
\boldsymbol{x}_1 & \dots & \boldsymbol{s}_1^{\ell-1} & \boldsymbol{s}_1^{\ell} & \dots & \boldsymbol{s}_2^{\ell-1} & \boldsymbol{s}_2^{\ell} & \dots & \boldsymbol{s}_n^{L} & \boldsymbol{x}_{\text{test}} & \dots & \hat{\boldsymbol{s}}^{\ell-1} \\
\mathbf{0} & \dots & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} \\
\hline
1 & \dots & \ell & \ell+1 & \dots & \ell & \ell+1 & \dots & L+1 & 1 & \dots & \ell \\
1 & \dots & \ell & \ell+1 & \dots & \ell & \ell+1 & \dots & L+1 & 1 & \dots & \ell \\
\hline
\frac{\sum_{i=1}^{\mathrm{N}} i^2}{\sum_{i=1}^{\mathrm{N}} i} & \dots & \frac{\sum_{i=1}^{\mathrm{N}} i^2}{\sum_{i=1}^{\mathrm{N}} i} & \frac{\sum_{i=1}^{\mathrm{N}} i^2}{\sum_{i=1}^{\mathrm{N}} i} & \dots & \frac{\sum_{i=1}^{\mathrm{N}} i^2}{\sum_{i=1}^{\mathrm{N}} i} & \frac{\sum_{i=1}^{\mathrm{N}} i^2}{\sum_{i=1}^{\mathrm{N}} i} & \dots & \frac{\sum_{i=1}^{\mathrm{N}} i^2}{\sum_{i=1}^{\mathrm{N}} i} & \frac{\sum_{i=1}^{\mathrm{N}} i^2}{\sum_{i=1}^{\mathrm{N}} i} & \dots & \frac{\sum_{i=1}^{\mathrm{N}} i^2}{\sum_{i=1}^{\mathrm{N}} i} \\
\ln(1) & \dots & \ln(\ell) & \ln(\ell+1) & \dots & \ln(L+\ell+1) & \ln(L+\ell+2) & \dots & \ln(n(L+1)) & \ln(n(L+1)+1) & \dots & \ln(\mathrm{N}) \\
0 & \dots & 0 & 0 & \dots & 0 & 0 & \dots & 0 & 1 & \dots & 1 \\
1 & \dots & 1 & 1 & \dots & 1 & 1 & \dots & 1 & 1 & \dots & 1 \\
\boldsymbol{m}_1 & \dots & \boldsymbol{m}_{\ell} & \boldsymbol{m}_{\ell+1} & \dots & \boldsymbol{m}_{L+\ell+1} & \boldsymbol{m}_{L+\ell+2} & \dots & \boldsymbol{m}_{n(L+1)} & \boldsymbol{m}_{n(L+1)+1} & \dots & \boldsymbol{m}_{\mathrm{N}}
\end{array}\right].$$

Notice that $\frac{\sum_{i=1}^{\mathrm{N}} i^2}{\sum_{i=1}^{\mathrm{N}} i} = \frac{2\mathrm{N}(\mathrm{N}+1)(2\mathrm{N}+1)}{6\mathrm{N}(\mathrm{N}+1)} = \frac{2\mathrm{N}+1}{3}$. We then use the ReLU layer as to multiply with 3/2 and subtract 1 from this column. This results to attention output

$$\left[\begin{array}{ccccccccccccc}
\boldsymbol{x}_1 & \dots & \boldsymbol{s}_1^{\ell-1} & \boldsymbol{s}_1^{\ell} & \dots & \boldsymbol{s}_2^{\ell-1} & \boldsymbol{s}_2^{\ell} & \dots & \boldsymbol{s}_n^{L} & \boldsymbol{x}_{\text{test}} & \dots & \hat{\boldsymbol{s}}^{\ell-1} \\
\mathbf{0} & \dots & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} \\
\hline
1 & \dots & \ell & \ell+1 & \dots & \ell & \ell+1 & \dots & L+1 & 1 & \dots & \ell \\
1 & \dots & \ell & \ell+1 & \dots & \ell & \ell+1 & \dots & L+1 & 1 & \dots & \ell \\
\hline
\mathrm{N} & \dots & \mathrm{N} & \mathrm{N} & \dots & \mathrm{N} & \mathrm{N} & \dots & \mathrm{N} & \mathrm{N} & \dots & \mathrm{N} \\
\ln(1) & \dots & \ln(\ell) & \ln(\ell+1) & \dots & \ln(L+\ell+1) & \ln(L+\ell+2) & \dots & \ln(n(L+1)) & \ln(n(L+1)+1) & \dots & \ln(\mathrm{N}) \\
0 & \dots & 0 & 0 & \dots & 0 & 0 & \dots & 0 & 1 & \dots & 1 \\
1 & \dots & 1 & 1 & \dots & 1 & 1 & \dots & 1 & 1 & \dots & 1 \\
\boldsymbol{m}_1 & \dots & \boldsymbol{m}_{\ell} & \boldsymbol{m}_{\ell+1} & \dots & \boldsymbol{m}_{L+\ell+1} & \boldsymbol{m}_{L+\ell+2} & \dots & \boldsymbol{m}_{n(L+1)} & \boldsymbol{m}_{n(L+1)+1} & \dots & \boldsymbol{m}_{\mathrm{N}}
\end{array}\right]$$

Notice that this step does not involve any error.

Step 2: Extract the identifier of the prediction to be made ℓ (3 layers). Now, by setting the key and query weight matrices of the attention as to just keep the all ones row, we get a matrix that attends equally to all tokens. Then the value weight matrix keeps

the row that contains ℓ ones and thus we get the number $\frac{\ell}{\mathrm{N}}$, propagated in all the sequence length. Then the attention output is as follows:

$$\left[\begin{array}{cccccccccccc}
\boldsymbol{x}_1 & \dots & \boldsymbol{s}_1^{\ell-1} & \boldsymbol{s}_1^{\ell} & \dots & \boldsymbol{s}_2^{\ell-1} & \boldsymbol{s}_2^{\ell} & \dots & \boldsymbol{s}_n^{L} & \boldsymbol{x}_{\text{test}} & \dots & \hat{\boldsymbol{s}}^{\ell-1} \\
\mathbf{0} & \dots & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} \\
\hline
1 & \dots & \ell & \ell+1 & \dots & \ell & \ell+1 & \dots & L+1 & 1 & \dots & \ell \\
1 & \dots & \ell & \ell+1 & \dots & \ell & \ell+1 & \dots & L+1 & 1 & \dots & \ell \\
\hline
\mathrm{N} & \dots & \mathrm{N} & \mathrm{N} & \dots & \mathrm{N} & \mathrm{N} & \dots & \mathrm{N} & \mathrm{N} & \dots & \mathrm{N} \\
\frac{\ell}{\mathrm{N}} & \dots & \frac{\ell}{\mathrm{N}} & \frac{\ell}{\mathrm{N}} & \dots & \frac{\ell}{\mathrm{N}} & \frac{\ell}{\mathrm{N}} & \dots & \frac{\ell}{\mathrm{N}} & \frac{\ell}{\mathrm{N}} & \dots & \frac{\ell}{\mathrm{N}} \\
0 & \dots & 0 & 0 & \dots & 0 & 0 & \dots & 0 & 1 & \dots & 1 \\
1 & \dots & 1 & 1 & \dots & 1 & 1 & \dots & 1 & 1 & \dots & 1 \\
\boldsymbol{m}_1 & \dots & \boldsymbol{m}_{\ell} & \boldsymbol{m}_{\ell+1} & \dots & \boldsymbol{m}_{L+\ell+1} & \boldsymbol{m}_{L+\ell+2} & \dots & \boldsymbol{m}_{n(L+1)} & \boldsymbol{m}_{n(L+1)+1} & \dots & \boldsymbol{m}_{\mathrm{N}}
\end{array}\right]. \tag{I.18}$$

So far this step does not involve any error. As described in Lemma I.3 to implement multiplication of two values up to any error ϵ, we need 1) have the two numbers to be multiplied next to each other and an extra structure (some constants, which we consider that are encoded in the last rows of the matrix $\boldsymbol{m}_i$) and 2) the corresponding structure presented in Lemma I.3 . So we need to make the following transformation in the rows containing N and $\frac{\ell}{\mathrm{N}}$

$$\begin{bmatrix} \mathrm{N} & \mathrm{N} & \dots & \mathrm{N} \\ \frac{\ell}{\mathrm{N}} & \frac{\ell}{\mathrm{N}} & \dots & \frac{\ell}{\mathrm{N}} \end{bmatrix} \rightarrow \begin{bmatrix} \mathrm{N} & \frac{\ell}{\mathrm{N}} & \dots & \mathrm{N} \\ \frac{\ell}{\mathrm{N}} & \frac{\ell}{\mathrm{N}} & \dots & \frac{\ell}{\mathrm{N}} \end{bmatrix} \tag{I.19}$$

$$\rightarrow \begin{bmatrix} \ell & * & \dots & * \\ * & * & \dots & * \end{bmatrix} \tag{I.20}$$

$$\rightarrow \begin{bmatrix} \ell & \ell & \dots & \ell \\ * & * & \dots & * \end{bmatrix} \tag{I.21}$$

where $*$ denotes inconsequential values. For the first step we assume that we have the

necessary binary representations [2] and then we use Lemma I.1 which shows how we can perform the operation of copy/paste. This step involves an error and the current output would be $\tilde{\boldsymbol{X}}_1 = \boldsymbol{X}^* + \varepsilon_1 \boldsymbol{M}_1$, where ε_1 is controlled and we will determine it in the sequence, $\|\boldsymbol{M}_1\| \leq 1$ and $\boldsymbol{X}_1^*$ is the desired output. For the second step now we use Lemma I.3 to perform the multiplication, this will affect some of the other values. To analyze the error of this step, we know that given $\boldsymbol{X}_1^*$ we will get the desired output $\boldsymbol{X}_2^* + \varepsilon_2 \boldsymbol{M}_2$, where $\|\boldsymbol{M}_2\| \leq 1$ and ε_2 is to be determined. However, are given $\tilde{X}_1$ in the multiplication procedure[3] which results in the following output $\tilde{\boldsymbol{X}}_2 = (\tilde{\boldsymbol{X}}_1)^\top \tilde{\boldsymbol{X}}_1 + \varepsilon_2 \boldsymbol{M}_2 = \boldsymbol{X}_1^{*\top} \boldsymbol{X}_1^* + \varepsilon_1 \boldsymbol{M}_1^\top \boldsymbol{X}_1^* + \varepsilon_1 \boldsymbol{M}_1^\top \boldsymbol{M}_1 + \varepsilon_2 \boldsymbol{M}_2$. Thus by choosing $\varepsilon_1, \varepsilon_2$ to be of order $\mathcal{O}(\delta)$we get that $\tilde{\boldsymbol{X}}_2 = \boldsymbol{X}_2^* + \frac{\delta}{C}\|M\|$ for some C chosen to be large enough. Then for the last step consider the follow (sub-)rows of the matrix $\boldsymbol{X}$

$$\begin{bmatrix} \ell & * & * & \dots & * \\ \boldsymbol{r}_1 & \boldsymbol{r}_1 & \boldsymbol{r}_1 & \dots & \boldsymbol{r}_1 \\ \boldsymbol{r}_1 & \boldsymbol{r}_2 & \boldsymbol{r}_3 & \dots & \boldsymbol{r}_{\mathrm{N}} \end{bmatrix} \tag{I.22}$$

where $\boldsymbol{r}_i$ is the binary representation of position i. By choosing $\boldsymbol{W}_k, \boldsymbol{W}_q$ as to

$$\boldsymbol{W}_k \boldsymbol{X} = \begin{bmatrix} \boldsymbol{r}_1 & \boldsymbol{r}_2 & \dots & \boldsymbol{r}_{\mathrm{N}} \end{bmatrix}, \quad \boldsymbol{W}_q \boldsymbol{X} = \begin{bmatrix} \boldsymbol{r}_1 & \boldsymbol{r}_1 & \dots & \boldsymbol{r}_1 \end{bmatrix} \tag{I.23}$$

[2] the size that we need will be $2 \log \mathrm{N}_{\max} + 1$, where $\mathrm{N}_{\max}$ is the maximum sequence length

[3] Note that the true error is even smaller since the operation performed only affects one row.

and consider $\boldsymbol{W}_v\boldsymbol{X} = \begin{bmatrix} \ell & * & \dots & * \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} \end{bmatrix}$ we have that

$$\begin{aligned}
\text{att}(\boldsymbol{X}) &= \boldsymbol{X} + \boldsymbol{W}_v\boldsymbol{X}\mathbb{S}((\boldsymbol{W}_k\boldsymbol{X})^\top \boldsymbol{W}_q\boldsymbol{X}) \\
&= \boldsymbol{X} + \begin{bmatrix} \ell & * & \dots & * \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} \end{bmatrix} \mathbb{S}\left(\begin{bmatrix} \boldsymbol{r}_1^\top \boldsymbol{r}_1 & \boldsymbol{r}_1^\top \boldsymbol{r}_1 & \dots & \boldsymbol{r}_1^\top \boldsymbol{r}_1 \\ \boldsymbol{r}_1^\top \boldsymbol{r}_2 & \boldsymbol{r}_1^\top \boldsymbol{r}_2 & \dots & \boldsymbol{r}_1^\top \boldsymbol{r}_2 \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{r}_1^\top \boldsymbol{r}_{\mathrm{N}} & \boldsymbol{r}_1^\top \boldsymbol{r}_{\mathrm{N}} & \dots & \boldsymbol{r}_1^\top \boldsymbol{r}_{\mathrm{N}} \end{bmatrix} \right) \\
&= \boldsymbol{X} + \begin{bmatrix} \ell & * & \dots & * \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} \end{bmatrix} \begin{bmatrix} 1 & 1 & \dots & 1 \\ 0 & 0 & \dots & 0 \\ \vdots & \vdots & \dots & \vdots \\ 0 & 0 & \dots & 0 \end{bmatrix} + \varepsilon_3 \boldsymbol{M}_3
\end{aligned}$$

By subtracting one identity head for the first that we focus on as described in Lemma I.1 we have that $\text{att}(\boldsymbol{X})$ results in the desired matrix. We output this result and overwrite ℓ/N, thus we have

$$\left[\begin{array}{ccccccccccccc}
\boldsymbol{x}_1 & \dots & \boldsymbol{s}_1^{\ell-1} & \boldsymbol{s}_1^{\ell} & \dots & \boldsymbol{s}_2^{\ell-1} & \boldsymbol{s}_2^{\ell} & \dots & \boldsymbol{s}_n^{L} & \boldsymbol{x}_{\text{test}} & \dots & \hat{\boldsymbol{s}}^{\ell-1} \\
\mathbf{0} & \dots & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} \\
\hline
1 & \dots & \ell & \ell+1 & \dots & \ell & \ell+1 & \dots & L+1 & 1 & \dots & \ell \\
1 & \dots & \ell & \ell+1 & \dots & \ell & \ell+1 & \dots & L+1 & 1 & \dots & \ell \\
\hline
\mathrm{N} & \dots & \mathrm{N} & \mathrm{N} & \dots & \mathrm{N} & \mathrm{N} & \dots & \mathrm{N} & \mathrm{N} & \dots & \mathrm{N} \\
\ell & \dots & \ell & \ell & \dots & \ell & \ell & \dots & \ell & \ell & \dots & \ell \\
0 & \dots & 0 & 0 & \dots & 0 & 0 & \dots & 0 & 1 & \dots & 1 \\
1 & \dots & 1 & 1 & \dots & 1 & 1 & \dots & 1 & 1 & \dots & 1 \\
\multicolumn{13}{c}{\boldsymbol{m}_1 \dots \boldsymbol{m}_\ell \ \boldsymbol{m}_{\ell+1} \dots \boldsymbol{m}_{L+\ell+1} \ \boldsymbol{m}_{L+\ell+2} \dots \boldsymbol{m}_{n(L+1)} \ \boldsymbol{m}_{n(L+1)+1} \dots \boldsymbol{m}_{\mathrm{N}}}
\end{array}\right] \tag{I.24}$$

The new error introduced by this operation is again controlled as to be $\varepsilon_3 \boldsymbol{M}_3 = \dfrac{\delta}{C}$, with $\|\boldsymbol{M}_3\| \leq 1$.

Step 3: Create $\ell+1$ (0 layer). We first copy the row with ℓ to the row with N, this can trivially be done with a ReLU layer that outputs zero everywhere else except for the row of Ns that output $(\ell)_+ - (N)_+$ to account also for the residual. We now use one of the bias terms (notice that ℓ is always positive) and set to one in one of the two rows that contain the ℓ, again we account for the residual as before; everything else remains unchanged. Thus, we have

$$\left[\begin{array}{cccccccccccc}
\boldsymbol{x}_1 & \dots & \boldsymbol{s}_1^{\ell-1} & \boldsymbol{s}_1^{\ell} & \dots & \boldsymbol{s}_2^{\ell-1} & \boldsymbol{s}_2^{\ell} & \dots & \boldsymbol{s}_n^{L} & \boldsymbol{x}_{\text{test}} & \dots & \hat{\boldsymbol{s}}^{\ell-1} \\
\mathbf{0} & \dots & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} \\
\hline
1 & \dots & \ell & \ell+1 & \dots & \ell & \ell+1 & \dots & L+1 & 1 & \dots & \ell \\
1 & \dots & \ell & \ell+1 & \dots & \ell & \ell+1 & \dots & L+1 & 1 & \dots & \ell \\
\hline
\ell+1 & \dots & \ell+1 & \ell+1 & \dots & \ell+1 & \ell+1 & \dots & \ell+1 & \ell+1 & \dots & \ell+1 \\
\ell & \dots & \ell & \ell & \dots & \ell & \ell & \dots & \ell & \ell & \dots & \ell \\
0 & \dots & 0 & 0 & \dots & 0 & 0 & \dots & 0 & 1 & \dots & 1 \\
1 & \dots & 1 & 1 & \dots & 1 & 1 & \dots & 1 & 1 & \dots & 1 \\
\boldsymbol{m}_1 & \dots & \boldsymbol{m}_\ell & \boldsymbol{m}_{\ell+1} & \dots & \boldsymbol{m}_{L+\ell+1} & \boldsymbol{m}_{L+\ell+2} & \dots & \boldsymbol{m}_{n(L+1)} & \boldsymbol{m}_{n(L+1)+1} & \dots & \boldsymbol{m}_{\text{N}}
\end{array}\right] \tag{I.25}$$

This operation can collectively be implemented (add the bias + copy the row) in the ReLU layer of the previous transformer layer that was not used in the previous step.

Step 4: Create the binary bits (2 layers). We will now use the information extracted in the previous steps, to create the binary indicators/bit to identify which tokens we want to filter. This can be easily implemented with one layer of transformer and especially the ReLU part of it. Notice that if we subtract the row that contains the tokens that have already been predicted, *i.e.,* $[\ell\ \ell \dots\ \ell]$ from the row that contains $[1\ 2 \dots L\ 1\ 2 \dots L \dots L]$ we will get zero only in the positions that we want to filter out and some non-zero quantity in the rest. This is trivially implemented with one ReLU layer. So, we need to implement an if..then type of operation. Basically, if the quantity at hand is zero we want to set the bit to one, while if it non-zero to set it to be zero. This can be implemented with the following ReLU part of a

transformer layer

$$\texttt{TF}(x_i) = 1 - (x_i)_+ - (-x_i)_+ + (x_i - 1)_+ + (-x_i - 1)_+ - ((x_i)_+ - (-x_i)_+) \tag{I.26}$$

the last two terms are to account for the residual. Again the rest of the rows do not change and are zeroed-out.

Step 5: Implement the filtering (1 layer). We now apply Lemma I.4 and our proof is completed. In this step as in Lemma I.4, the error remains of the order of the error of the previous step. Thus, by fine-tuning the constants appropriately, depending on 1) the bound on the input as of Assumption I.1, the targeted error δ and the constant used in Lemma I.4 we achieve that the error has the target upped bound. ∎

Theorem I.1 (Theorem 10.1 restated) *Consider a prompt $\boldsymbol{p}_n(f)$ generated from an L-layer MLP $f(\cdot)$ as described in Definition 10.1, and assume given test example $(\boldsymbol{x}_{test}, \boldsymbol{s}^1_{test}, \dots \boldsymbol{s}^L_{test})$. For any resolution $\epsilon > 0$, there exists $\delta = \delta(\epsilon)$, iteration choice $T = \mathcal{O}(\kappa^2_{\max} \log(1/\epsilon))$, and a backend transformer construction $\texttt{TF}_{\texttt{BE}}$ such that the concatenated transformer $\texttt{TF} = \texttt{TF}_{\texttt{LR}} \circ \texttt{TF}_{\texttt{BE}}$ implements the following: Let $(\hat{\boldsymbol{s}}^i)_{i=1}^{\ell-1}$ denote the first $\ell-1$ CoT-I/O outputs of $\texttt{TF}$ and set $\boldsymbol{p}[\ell] = (\boldsymbol{p}_n(f), \boldsymbol{x}_{test}, \hat{\boldsymbol{s}}^1 \dots \hat{\boldsymbol{s}}^{\ell-1})$. At step ℓ, $\texttt{TF}$ implements*

1. ***Filtering.*** *Define the filtered prompt with input/output features of layer ℓ,*

$$\boldsymbol{p}_n^{filter} = \begin{pmatrix} \dots \mathbf{0}, \boldsymbol{s}_1^{\ell-1}, \mathbf{0} \ \dots \mathbf{0}, \boldsymbol{s}_n^{\ell-1}, \mathbf{0} \ \dots \mathbf{0}, \hat{\boldsymbol{s}}^{\ell-1} \\ \dots \mathbf{0}, \ \mathbf{0}, \ \boldsymbol{s}_1^{\ell} \dots \mathbf{0}, \ \mathbf{0}, \ \boldsymbol{s}_n^{\ell} \dots \mathbf{0}, \ \mathbf{0} \end{pmatrix}.$$

   There exists a fixed projection matrix $\boldsymbol{\Pi}$ that applies individually on tokens such that the backend output obeys $\|\boldsymbol{\Pi}(\texttt{TF}_{\texttt{BE}}(\boldsymbol{p}[\ell])) - \boldsymbol{p}_n^{filter}\| \le \delta$.

2. ***Gradient descent.*** *The combined model obeys $\|\texttt{TF}(\boldsymbol{p}[\ell]) - \boldsymbol{s}^{\ell}_{test}\| \le \ell \cdot \epsilon / L$.*

$\texttt{TF}_{\texttt{BE}}$ has constant number of layers independent of T and n. Consequently, after L rounds of CoT-I/O, $\texttt{TF}$outputs $f(\boldsymbol{x}_{test})$ up to ϵ accuracy.

Proof. We apply Lemma I.5 from which it is clear that there exists a projection such that the result stated in (1. Filtering) holds, and it is independent to T, n and ℓ. Next we turn to prove (2. Gradient descent). In Definition 10.1, we assume that the network's activation function is leaky-ReLU, *i.e.,*

$$\phi(x) = \begin{cases} x, & \text{if } x \ge 0 \\ \alpha x, & \text{otherwise.} \end{cases} \tag{I.27}$$

Thus, as a first step we construct the inverse of leaky-ReLU and apply it in the second row of $\boldsymbol{p}_n^{\text{filter}}$ where the inverse of leaky-ReLU is

$$\phi^{-1}(y) = \begin{cases} y, & \text{if } y \geq 0 \\ y/\alpha, & \text{otherwise.} \end{cases} \tag{I.28}$$

This can be implemented with the following activation function (denoted by $\sigma(\cdot)$) using ReLUs:

$$\sigma(x) = (x)_+ - 1/\alpha(-x)_+. \tag{I.29}$$

After it, it remains $\texttt{TF}_{\texttt{LR}}$ to solve linear regression problems. Taking ℓth layer, first neuron prediction as an example, and letting $\boldsymbol{x}_i' := \boldsymbol{s}_i^{\ell-1}$, $y_i' := \phi^{-1}(\boldsymbol{s}_i^{\ell}[0])$ and $\boldsymbol{w} = \boldsymbol{W}_\ell[0]$, linear regression has form of $y_i' = \boldsymbol{w}^\top \boldsymbol{x}_i'$ for $i \in [n]$. Notice that the extra zeros do not contribute in the update performed by gradient descent, and after gradient descent has been performed, we apply back the leaky ReLU. Then following Assumption 10.1, since we assume $\texttt{TF}_{\texttt{LR}}$ performs the same as gradient descent optimizer, given matrix condition as described in Definition 10.1, after running T iterations of gradient descend on the linear regression problem and considering each layer prediction with resolution ϵ/L, we can get that $\|\texttt{TF}_{\texttt{LR}}(\boldsymbol{p}[\ell]) - \bar{\boldsymbol{s}}^\ell\| \leq \epsilon/L$, where $\bar{\boldsymbol{s}}^\ell = \phi(\boldsymbol{W}_\ell \hat{\boldsymbol{s}}^{\ell-1})$ is the correct prediction if taking $\hat{\boldsymbol{s}}^{\ell-1}$ as input. Then we have

$$\|\texttt{TF}_{\texttt{LR}}(\boldsymbol{p}[\ell] - \boldsymbol{s}_{\text{test}}^{\ell})\| \leq \|\texttt{TF}_{\texttt{LR}}(\boldsymbol{p}[\ell]) - \bar{\boldsymbol{s}}^\ell\| + \|\boldsymbol{W}_\ell(\hat{\boldsymbol{s}}^{\ell-1} - \boldsymbol{s}_{\text{test}}^{\ell-1})\| \lesssim \epsilon/L + \|\texttt{TF}_{\texttt{LR}}(\boldsymbol{p}[\ell-1] - \boldsymbol{s}_{\text{test}}^{\ell-1})\|.$$

Let $\texttt{TF}_{\texttt{LR}}(\boldsymbol{p}[0])$ returns $\boldsymbol{x}_{\text{test}}$ and therefore $\|\texttt{TF}_{\texttt{LR}}(\boldsymbol{p}[0]) - \boldsymbol{x}_{\text{test}}\| = 0$. Combing results in that $\|\texttt{TF}_{\texttt{LR}}(\boldsymbol{p}[\ell] - \boldsymbol{s}_{\text{test}}^{\ell})\| \lesssim \ell \cdot \epsilon/L$. Since from Lemma I.5 we have that we can choose δ to be arbitrary. Let $\delta = \epsilon/L$, where L is the total predictions we will make. Then it will result in $\|\texttt{TF}(\boldsymbol{p}[\ell] - \boldsymbol{s}_{\text{test}}^{\ell})\| \lesssim \ell \cdot \epsilon/L$, which completes the proof.

■

I.2 Experimental Details

In this section, we provide the implementation details of our experiments.

I.2.1 Model Evaluation

Recap the same setting as in Section 10.2.3 and assume we have pretrained models with parameters $\hat{\boldsymbol{\beta}}^{\text{CoT-I}}$ and $\hat{\boldsymbol{\beta}}^{\text{CoT-I/O}}$. Next we make predictions following Section 10.2.2. Letting

$\ell(\cdot,\cdot) : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$ be loss function, we can define test risks as follows.

$$\mathcal{L}^{\text{CoT-I}}(n) = \mathbb{E}_{(\boldsymbol{x}_i)_{i=1}^n,(f_\ell)_{\ell=1}^L}\left[\ell(\hat{\boldsymbol{y}}_n, f(\boldsymbol{x}_n))\right] \quad \text{where} \quad \hat{\boldsymbol{y}}_n = \texttt{TF}(\boldsymbol{p}_n(f), \boldsymbol{x}_n; \hat{\boldsymbol{\beta}}^{\text{CoT-I}})$$

and

$$\mathcal{L}^{\text{CoT-I/O}}(n) = \mathbb{E}_{(\boldsymbol{x}_i)_{i=1}^n,(f_\ell)_{\ell=1}^L}\left[\ell(\hat{\boldsymbol{y}}_n, f(\boldsymbol{x}_n))\right] \quad \text{where} \quad \hat{\boldsymbol{y}}_n = \texttt{TF}(\boldsymbol{p}_n(f), \boldsymbol{x}_n, \hat{\boldsymbol{s}}^1 \cdots, \hat{\boldsymbol{s}}^{L-1}; \hat{\boldsymbol{\beta}}^{\text{CoT-I/O}}).$$

Here, we use $\mathcal{L}(n)$ to define the test risk when given prompt with n in-context samples. Then, results shown in Figures 10.3&10.4&10.5&10.6&10.7a are test risks $\mathcal{L}(n)$ given $n \in [N]$. Following model training and evaluation, we can see that once loss function is the same for both training and predicting, CoT-I (as well as ICL) accepts training risk $\mathcal{L}_{\text{train}}^{\text{CoT-I}} = \frac{1}{N}\sum_{n=1}^N \mathcal{L}^{\text{CoT-I}}(n)$.

I.2.2 Implementation

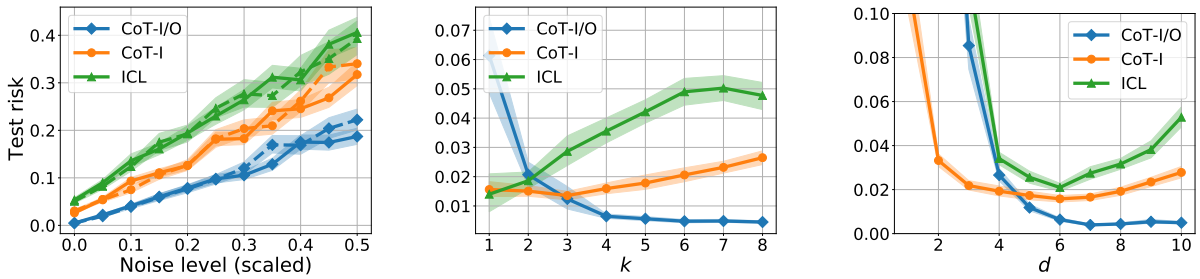
All the transformer experiments use the GPT-2 model [164] and our codebase is based on prior works [59, 206]. Specifically, the model is trained using the `Adam` optimizer with learning rate 0.0001 and batch size 64, and we train 500k iterations in total for all ICL, CoT-I and CoT-I/O methods. Each iteration randomly samples inputs $\boldsymbol{x}$s and functions fs. We also apply curriculum learning over the prompt n as [59] did for 2-layer random MLPs (Sec. 10.4.2). For both training and testing, we use the squared error as the loss function, *i.e.*, $\ell(\hat{\boldsymbol{y}}, \boldsymbol{y}) = \|\hat{\boldsymbol{y}} - \boldsymbol{y}\|^2$ (or $\ell(\hat{y}, y) = (\hat{y} - y)^2$ for scalar).

I.3 Additional Experimental Results

I.3.1 Out-of-distribution Experiments

While our primary focus lies in in-distribution scenarios, wherein the test examples are presumed to follow the same distribution as the training data, this subsection extends our analysis to out-of-distribution (OOD) settings.

To provide a clearer understanding of our model's behavior under distribution shifts and to quantitatively assess the impact on test risk, we conduct experiments with varying levels of distribution shift. Specifically, we evaluate the test risk when the prompt is supplemented with 100 in-context examples. The results of these experiments are depicted in Figure I.1, which serves to illustrate how our model's performance is affected under these OOD conditions. In Fig. I.1a, we analyze noisy in-context samples during testing. The solid and dashed



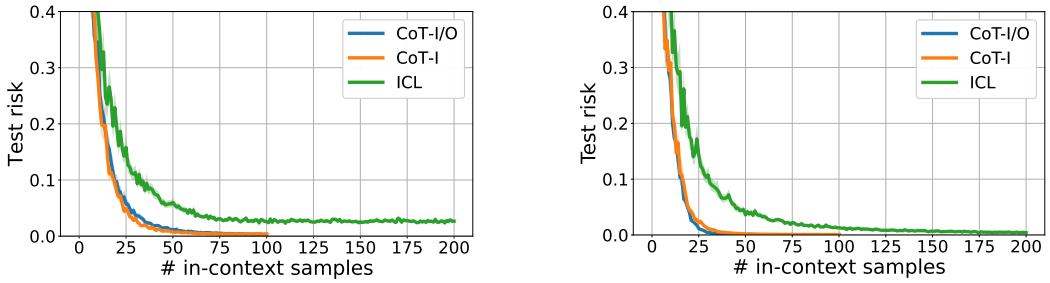
(a) Test with noisy data (b) Test with different k (c) Test with different d

Figure I.1: We implement robustness experiments under different distribution shift levels. In Fig. I.1a we add noise to the label y (solid) or input features $\boldsymbol{x}$ (dashed). In Fig. I.1b, we in-context learn an MLP with $k \in [8]$ hidden nodes whereas transformer is trained for MLPs with 8 hidden nodes. In Fig. I.1b, we consider misspecification of input dimension: TF is trained with $d = 10$ but we feed a neural net with $d \leq 10$.

curves represent the test risks, corresponding to the noisy in-context samples whose (input, output) takes the form of either $(\boldsymbol{x}, y + \text{noise})$ or $(\boldsymbol{x} + \text{noise}, y)$, respectively. The results indicate that CoT exhibits greater robustness compared to ICL, and the test risks increase linearly with the noise level, with attributed to the randomized MLPs setting. Additionally, in Figs. I.1b&I.1c, we instead explore out-of-distribution test tasks where test MLPs differ in (d, k) from the training phase. For both subfigures, we firstly train small GPT-2 using 2-layer MLPs with $d = 10, k = 8$. In Fig. I.1b, we fix $d = 10$ and vary k from 1 to 8, whereas in Fig. I.1c, we fix $k = 8$ and vary d from 1 to 10. In both instances, the findings reveal that CoT's performance remains almost consistent when $k \geq 4$ or $d \geq 6$, and ICL is unable to surpass it. The improved performance of ICL with smaller values of d or k again reinforces our central assertion that ICL requires $O(dk)$ samples for in-context learning of the 2-layer random MLP, and reducing either d or k helps in improving the performance. Given that we employ the ReLU activation function, smaller values of d or k can lead to significant bias in the intermediate feature. Consequently, CoT cannot derive substantial benefits from this scenario, resulting in a decline in performance.

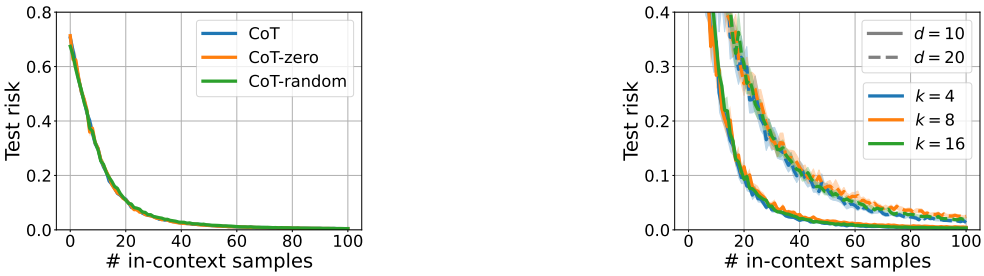
I.3.2 Further Evidence of Model Expressivity

There are two determinants of model learnability in in-context tasks: sample complexity and model expressivity. Sample complexity pertains to the number of samples needed to precisely solve a problem. However, when the transformer model is small, even with a sufficiently large number of samples, due to its lack of expressivity, ICL cannot achieve zero test risk. This contrasts with CoT, which decomposes complex tasks into simpler sub-tasks,



(a) Small GPT-2, $d = 10$, $k = 4$ (b) Standard GPT-2, $d = 10$, $k = 4$

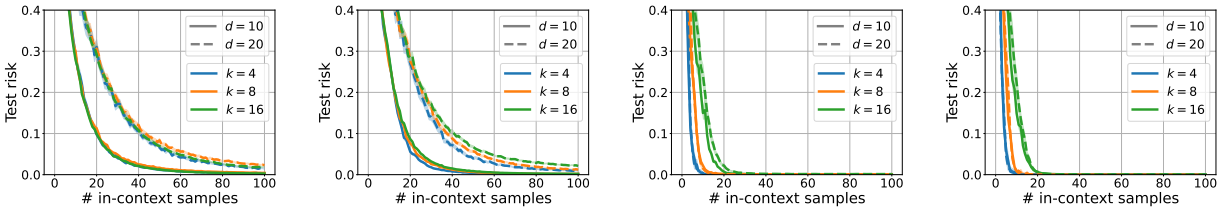
Figure I.2: We train ICL with more in-context examples. The main conclusion is that: For small GPT, ICL can indeed not approximate the neural net even with many examples (unlike CoT) whereas, for large GPT, ICL can do so (although much less efficient). This is in line with our theoretical intuitions on the expressivity benefits of CoT.



(a) Test filtering on first layer prediction (b) CoT-I/O: composed risk (same as Fig. 10.3a)

Figure I.3: Fig. I.3a presents a filtering evidence of 2-layer MLPs. Given a 2-layer MLP in-context example $(\boldsymbol{x}, \boldsymbol{s}, y)$, CoT admits $(\boldsymbol{x}, \boldsymbol{s}, y)$ as test sample; while test samples of CoT-zero and CoT-random are formed by $(\boldsymbol{x}, \boldsymbol{s}, 0)$ and $(\boldsymbol{x}, \boldsymbol{s}, z)$ where $z \sim \mathcal{N}(0, d)$. Fig. I.3b is directly cloned form Fig. 10.3a with error bar for better comparison with results in Fig. I.4.

thereby requiring smaller models for expression. Figure 10.5 has illustrated the expressivity of different GPT-2 architectures, showing that the tiny GPT-2 model is too small to express even a single layer of 2-layer MLPs. Additionally, we have run more experiments, and the results are shown in Figure I.2. Both Figures I.2a and I.2b detail training models with MLP tasks of dimensions $d = 10$ and $k = 4$. In Fig. I.2a, we use a small GPT-2 model, and the results show that the test risk stops decreasing even with more in-context examples. In Fig. I.2b, we train a larger model, and the results demonstrate that the standard GPT-2 is sufficient to express a 2-layer MLP with $d = 10$ and $k = 4$.



(a) CoT-I/O: 1st layer (b) ICL with $(\boldsymbol{x}, \boldsymbol{s})$ pairs (c) CoT-I/O: 2nd layer (d) ICL with $(\boldsymbol{s}, y)$ pairs

Figure I.4: We compare the performance of filtered CoT and ICL. In Fig. I.4a&I.4c, we decouple the composed risk of predicting 2-layer MLPs into risks of individual layers (following Section 10.4.1), which shows the filtered CoT results. In Fig. I.4b&I.4d, we train two additional models using ICL method taking $(\boldsymbol{x}, \boldsymbol{s})$ and $(\boldsymbol{s}, y)$ as inputs.

I.3.3 Filtering Evidence in 2-layer MLPs

Section 10.5 has demonstrated the occurrence of filtering in the linear deep MLPs setting (black dotted curves in Fig. 10.8b). In this section, we present further empirical evidence based on the 2-layer MLPs setting discussed in Section 10.4.2.

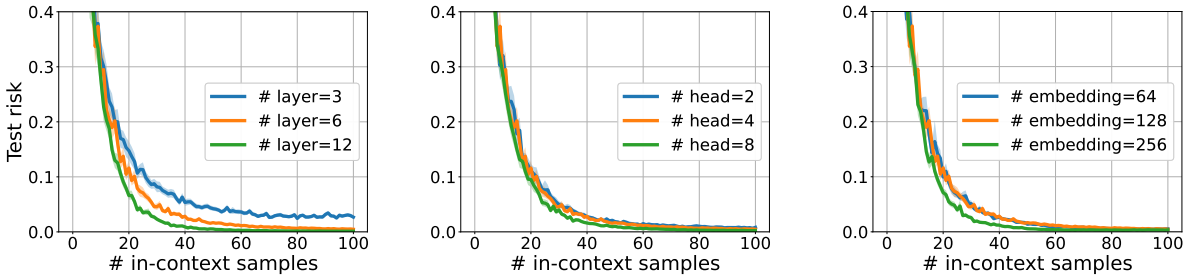
Follow the same setting as Figure 10.3a and choose $d = 10$ and $k = 8$. Assume we have a model pretrained using CoT-I/O method. As described in Section 10.4.2, during training, the prompt consists of in-context samples in the form of $(\boldsymbol{x}, \boldsymbol{s}, y)$ where $\boldsymbol{s} = (\boldsymbol{W}\boldsymbol{x})_+$ and $y = \boldsymbol{v}^\top \boldsymbol{s}$. To investigate filtering, we make three different predictions to evaluate the intermediate output, whose test prompts have in-context examples with the following forms:

$$\text{CoT: } (\boldsymbol{x}, \boldsymbol{s}, y), \quad \text{CoT-zero: } (\boldsymbol{x}, \boldsymbol{s}, 0), \quad \text{CoT-random: } (\boldsymbol{x}, \boldsymbol{s}, z),$$

where $z \sim \mathcal{N}(0, d)$. The results are displayed in Figure I.3a where blue, orange and green curves represent first layer prediction results using CoT, CoT-zero and CoT-random prompts, respectively. From this figure, we observe that the three curves are well aligned, indicating that when making a prediction for input $\boldsymbol{x}$, `TF` will attend only to $(\boldsymbol{x}, \boldsymbol{s})$ and ignore y. Therefore filling the positions of y with any random values (or zero) will not change the performance of first layer prediction.

I.3.4 Comparison of Filtered CoT with ICL

Until now, many experimental results have shown that CoT-I/O provides benefits in terms of sample complexity and model expressivity compared to ICL. As an interpretation, we state that CoT can be decoupled into two phases: *Filtering* and *ICL*, and theoretical results have been provided to prove this statement. As for the empirical evidence, Sections 10.5



(a) Varying GPT-2 layers (b) Varying GPT-2 heads (c) Varying embedding dimensions

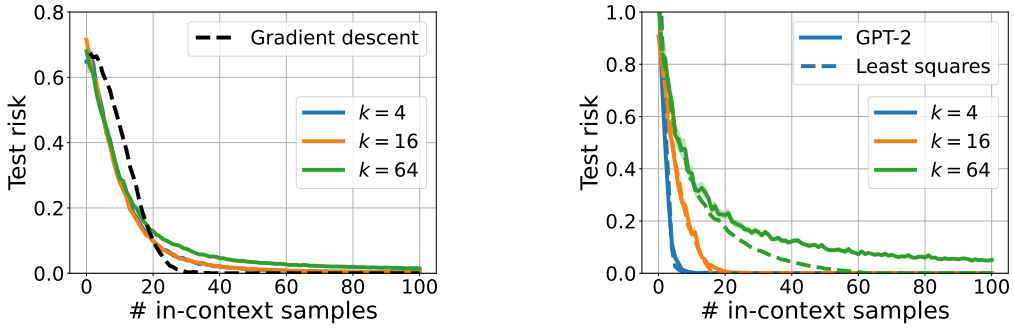
Figure I.5: To further investigate how model architectures impact the prediction performance, we fix the number of heads and embedding dimension in Fig. I.5a and change the layer number in $\{3, 6, 12\}$. Similarly for Fig. I.5b&I.5c but instead, change number of heads (in $\{2, 4, 8\}$) and embedding dimensions (in $\{64, 128, 256\}$).

and I.3.3 precisely show that filtering does occur in practice. In this section, we provide additional experiments to demonstrate that, after filtering, CoT performs similarly to ICL.

For convenience and easier comparison, we repeat the same results as Fig. 10.3a in Fig. I.3b, where $d \in \{10, 20\}$, $k \in \{4, 8, 16\}$, and train with a small GPT-2. We again recap the data setting for the 2-layer MLP, where the in-context examples of CoT prompt are in the form of $(\boldsymbol{x}, \boldsymbol{s}, y)$. Given that filtering happens, we make first and second layer predictions following Section 10.4.1 and results are presented in Fig. I.4a and Fig. I.4c, respectively. These results show the performances of the filtered CoT prompts. Next, we need to compare the performance with separate ICL training. To achieve this goal, we train a small GPT-2 model using ICL method with prompt containing $(\boldsymbol{x}, \boldsymbol{s})$ pairs (first layer). The test results are shown in Fig. I.4b. Additionally, in Fig. I.4d, we train another small GPT-2 model using ICL but with prompts containing $(\boldsymbol{s}, y)$ pairs (second layer). By comparing Fig. I.4a and I.4b, as well as Fig. I.4c and I.4d, we observe that after filtering, CoT-I/O achieves similar performance as individually training a single-step problem through the ICL phase.

I.3.5 CoT across Different Sizes of GPT-2

In Figure 10.5, we have demonstrated that larger models help in improving performance due to their ability of solving more complex function sets. However, since tiny, small and standard GPT-2 models scale the layer number, head number and embedding dimension simultaneously, it is difficult to determine which component has the greatest impact on performance. Therefore in this section, we investigate how different components of transformer model affect the resulting performance by run CoT-I/O on various GPT-2 architectures.



(a) 1st layer: compare with GD solving ReLU (b) 2nd layer: compare with least squares

Figure I.6: Fig. I.6a: compare transformer results (solid) with gradient descent optimizer (dashed) when solving first layer of 2-layer MLPs; Fig. I.6b: compare transformer result (solid) with least squares optimizer (dashed) when solving the second layer of 2-layer MLPs.

We maintain the same setting as in Section 10.4.2, fix $d = 10$ and $k = 8$, and consider a base GPT-2 model (small GPT-2) with 6 attention layers, 4 heads in each layer and 128-dimensional embeddings. In Fig. I.5a, we fix the number of heads at 4 and the embedding dimension at 128, while varying the number of layers in $\{3, 6, 12\}$. Similarly, we explore different models with different numbers of heads and embedding dimensions, and the results are respectively presented in Fig. I.5b and I.5c. Comparing them, we can observe the following: 1) once the problem is sufficiently solved, increasing the model size does not significantly improve the prediction performance (see Fig. I.5b&I.5c); 2) the number of layers influences model expressivity, particularly for small GPT-2 architecture (see Fig. I.5a).

I.3.6 Comparison of Transformer Prediction and Linear Regression

We also provide experimental findings to verify Assumption 10.1 in this section. Previous work [62, 4] has theoretically proven that `TF` can perform similar to gradient descent, and empirical evidence from [39, 59, 113] suggests that `TF` can even be competitive with Bayes optimizer in certain scenario. To this end, we first repeat the same first/layer predictions from Figure 10.4 in Figure I.6, where $d = 10$ and blue, orange and green solid curves represent the performances of $k = 4, 16, 64$ using pretrained small GPT-2 models. We also display the evaluations of gradient descent/least square solutions in dashed curves. Specifically, in Fig. I.6a , we solve problem

$$\hat{\boldsymbol{w}}_n = \arg\min_{\boldsymbol{w}} \frac{1}{n} \sum_{i=1}^{n} \|(\boldsymbol{w}^\top \boldsymbol{x}_i)_+ - y_i\|^2 \quad \text{where} \quad \boldsymbol{x}_i \sim \mathcal{N}(0, \boldsymbol{I}_d), \; y_i = (\boldsymbol{w}^{\star\top} \boldsymbol{x}_i)_+$$

for some $\boldsymbol{w}^\star \sim \mathcal{N}(0, 2\boldsymbol{I}_d)$ and n is the training sample size. Then the normalized test risks are computed by $\mathcal{L}(n) = \mathbb{E}_{\boldsymbol{w}^\star, \boldsymbol{x}}[\|(\hat{\boldsymbol{w}}_n^\top \boldsymbol{x})_+ - y\|^2]/d$, and we show point-to-point results for $n \in [N]$ in black dashed curve in Fig. I.6a[4]. As for the second layer, we solve least squares problems as follows

$$\hat{\boldsymbol{v}}_n = \boldsymbol{S}^\dagger \boldsymbol{y} \quad \text{where} \quad \boldsymbol{S} \in \mathbb{R}^{n \times k},\ \boldsymbol{S}[i] = (\boldsymbol{W}^\star \boldsymbol{x}_i)_+,\ \boldsymbol{y}[i] = \boldsymbol{v}^{\star\top} \boldsymbol{S}[i],\ \boldsymbol{x}_i \in \mathcal{N}(0, \boldsymbol{I}_d)$$

for some $\boldsymbol{W}^\star \in \mathbb{R}^{k \times d} \sim \mathcal{N}(0, 2/k)$ and $\boldsymbol{v}^\star \sim \mathcal{N}(0, \boldsymbol{I}_k)$. Here, † represents the pseudo-inverse operator. Then we calculate the normalized test risk of least square solution (given n training samples) as $\mathcal{L}(n) = \mathbb{E}_{\boldsymbol{W}^\star, \boldsymbol{v}^\star, \boldsymbol{x}}[\|\hat{\boldsymbol{v}}_n^\top \boldsymbol{s} - y\|^2]/d$ where $\boldsymbol{s} = (\boldsymbol{W}^\star \boldsymbol{x})_+$ and $y = \boldsymbol{v}^{\star\top} \boldsymbol{s}$. The results are presented in Fig. I.6b where blue, orange and green dashed curves correspond to solving the problem using different values of $k \in \{4, 16, 64\}$. In this figure, the curves for $k = 4, 16$ are aligned with GPT-2 risk curves, which indicates that `TF` can efficiently solve linear regression as a least squares optimizer. However, the curve for $k = 64$ does not align, which can be attributed to the increased complexity of the function set with higher dimensionality ($k = 64$). Learning such complex functions requires a larger `TF` model.

[4]To mitigate the bias introduced by ReLU activation, we subtract the mean value during prediction, *i.e.*, $\mathcal{L}(n) = \mathbb{E}_{\boldsymbol{w}^\star, \boldsymbol{x}}[\|(\hat{\boldsymbol{w}}_n^\top \boldsymbol{x})_+ - y - (\mathbb{E}_{\boldsymbol{x}}[(\hat{\boldsymbol{w}}_n^\top \boldsymbol{x})_+ - y])\|^2]/d$.

BIBLIOGRAPHY

[1] Huggingface pretrained models, 2023.

[2] Rishabh Agarwal, Avi Singh, Lei M Zhang, Bernd Bohnet, Stephanie Chan, Ankesh Anand, Zaheer Abbas, Azade Nova, John D Co-Reyes, Eric Chu, et al. Many-shot in-context learning. *arXiv preprint arXiv:2404.11018*, 2024.

[3] Kwangjun Ahn, Xiang Cheng, Hadi Daneshmand, and Suvrit Sra. Transformers learn to implement preconditioned gradient descent for in-context learning. *Advances in Neural Information Processing Systems*, 36, 2023.

[4] Ekin Akyürek, Dale Schuurmans, Jacob Andreas, Tengyu Ma, and Denny Zhou. What learning algorithm is in-context learning? investigations with linear models. In *The Eleventh International Conference on Learning Representations*, 2023.

[5] Zeyuan Allen-Zhu, Yuanzhi Li, and Zhao Song. A convergence theory for deep learning via over-parameterization. In *International Conference on Machine Learning*, pages 242–252. PMLR, 2019.

[6] Sanjeev Arora, Simon Du, Wei Hu, Zhiyuan Li, and Ruosong Wang. Fine-grained analysis of optimization and generalization for overparameterized two-layer neural networks. In *International Conference on Machine Learning*, pages 322–332. PMLR, 2019.

[7] Akari Asai, Mohammadreza Salehi, Matthew E Peters, and Hannaneh Hajishirzi. Attempt: Parameter-efficient multi-task tuning via attentional mixtures of soft prompts. *arXiv preprint arXiv:2205.11961*, 2022.

[8] Navid Azizan and Babak Hassibi. Stochastic gradient/mirror descent: Minimax optimality and implicit regularization. In *International Conference on Learning Representations*, 2018.

[9] Olivier Bachem, Mario Lucic, and Andreas Krause. Practical coreset constructions for machine learning. *arXiv preprint arXiv:1703.06476*, 2017.

[10] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.

[11] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *The International Conference on Learning Representations*, 2015.

[12] Yu Bai, Fan Chen, Huan Wang, Caiming Xiong, and Song Mei. Transformers as statisticians: Provable in-context learning with in-context algorithm selection. *Advances in neural information processing systems*, 36, 2024.

[13] Peter L Bartlett, Michael I Jordan, and Jon D McAuliffe. Convexity, classification, and risk bounds. *Journal of the American Statistical Association*, 101(473):138–156, 2006.

[14] Peter L Bartlett, Philip M Long, Gábor Lugosi, and Alexander Tsigler. Benign overfitting in linear regression. *Proceedings of the National Academy of Sciences*, 117(48):30063–30070, 2020.

[15] Jonathan Baxter. A model of inductive bias learning. *Journal of artificial intelligence research*, 12:149–198, 2000.

[16] Maximilian Beck, Korbinian Pöppel, Markus Spanring, Andreas Auer, Oleksandra Prudnikova, Michael Kopp, Günter Klambauer, Johannes Brandstetter, and Sepp Hochreiter. xLSTM: Extended long short-term memory. *arXiv preprint arXiv:2405.04517*, 2024.

[17] Mikhail Belkin, Daniel Hsu, Siyuan Ma, and Soumik Mandal. Reconciling modern machine-learning practice and the classical bias–variance trade-off. *Proc. of the National Academy of Sciences*, 116(32):15849–15854, 2019.

[18] Mikhail Belkin, Daniel J Hsu, and Partha Mitra. Overfitting or perfect fitting? risk bounds for classification and regression rules that interpolate. *Advances in neural information processing systems*, 31, 2018.

[19] Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman Vaughan. A theory of learning from different domains. *Machine learning*, 79(1):151–175, 2010.

[20] Olivier Bousquet and André Elisseeff. Algorithmic stability and generalization performance. *Advances in Neural Information Processing Systems*, 13, 2000.

[21] Olivier Bousquet and André Elisseeff. Stability and generalization. *The Journal of Machine Learning Research*, 2:499–526, 2002.

[22] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

[23] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

[24] Abdulkadir Canatar, Blake Bordelon, and Cengiz Pehlevan. Spectral bias and task-model alignment explain generalization in kernel regression and infinitely wide neural networks. *Nature communications*, 12(1):2914, 2021.

[25] Yuan Cao, Zixiang Chen, Mikhail Belkin, and Quanquan Gu. Benign overfitting in two-layer convolutional neural networks. *arXiv preprint arXiv:2202.06526*, 2022.

[26] Yuan Cao, Zhiying Fang, Yue Wu, Ding-Xuan Zhou, and Quanquan Gu. Towards understanding the spectral bias of deep learning. *arXiv preprint arXiv:1912.01198*, 2019.

[27] Yuan Cao, Quanquan Gu, and Mikhail Belkin. Risk bounds for over-parameterized maximum margin classification on sub-gaussian mixtures. *Advances in Neural Information Processing Systems*, 34:8407–8418, 2021.

[28] Marcus Carlsson. von neumann's trace inequality for hilbert–schmidt operators. *Expositiones Mathematicae*, 39(1):149–157, 2021.

[29] Rich Caruana. Multitask learning. *Machine learning*, 28(1):41–75, 1997.

[30] Xiangyu Chang, Yingcong Li, Muti Kara, Samet Oymak, and Amit K Roy-Chowdhury. Provable benefits of task-specific prompts for in-context learning. *arXiv preprint arXiv:2503.02102*, 2025.

[31] Xiangyu Chang, Yingcong Li, Samet Oymak, and Christos Thrampoulidis. Provable benefits of overparameterization in model compression: From double descent to pruning neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 6974–6983, 2021.

[32] Niladri S Chatterji and Philip M Long. Finite-sample analysis of interpolating linear classifiers in the overparameterized regime. *Journal of Machine Learning Research*, 22(129):1–30, 2021.

[33] Mia Xu Chen, Orhan Firat, Ankur Bapna, Melvin Johnson, Wolfgang Macherey, George Foster, Llion Jones, Mike Schuster, Noam Shazeer, Niki Parmar, Ashish Vaswani, Jakob Uszkoreit, Lukasz Kaiser, Zhifeng Chen, Yonghui Wu, and Macduff Hughes. The best of both worlds: Combining recent advances in neural machine translation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 76–86, Melbourne, Australia, July 2018. Association for Computational Linguistics.

[34] Qiwei Chen, Huan Zhao, Wei Li, Pipei Huang, and Wenwu Ou. Behavior sequence transformer for e-commerce recommendation in alibaba. In *Proceedings of the 1st International Workshop on Deep Learning Practice for High-Dimensional Sparse Data*, pages 1–4, 2019.

[35] Zihan Chen, Song Wang, Zhen Tan, Jundong Li, and Cong Shen. Maple: Many-shot adaptive pseudo-labeling for in-context learning. *arXiv preprint arXiv:2505.16225*, 2025.

[36] Jianpeng Cheng, Li Dong, and Mirella Lapata. Long short-term memory-networks for machine reading. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 551–561, Austin, Texas, November 2016. Association for Computational Linguistics.

[37] Krzysztof Marcin Choromanski, Valerii Likhosherstov, David Dohan, Xingyou Song, Andreea Gane, Tamas Sarlos, Peter Hawkins, Jared Quincy Davis, Afroz Mohiuddin, Lukasz Kaiser, David Benjamin Belanger, Lucy J Colwell, and Adrian Weller. Rethinking attention with performers. In *International Conference on Learning Representations*, 2021.

[38] Liam Collins, Advait Parulekar, Aryan Mokhtari, Sujay Sanghavi, and Sanjay Shakkottai. In-context learning with transformers: Softmax attention adapts to function lipschitzness. *arXiv preprint arXiv:2402.11639*, 2024.

[39] Damai Dai, Yutao Sun, Li Dong, Yaru Hao, Shuming Ma, Zhifang Sui, and Furu Wei. Why can GPT learn in-context? language models secretly perform gradient descent as meta-optimizers. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki, editors, *Findings of the Association for Computational Linguistics: ACL 2023*, pages 4005–4019, Toronto, Canada, July 2023. Association for Computational Linguistics.

[40] Tri Dao and Albert Gu. Transformers are ssms: Generalized models and efficient algorithms through structured state space duality. *arXiv preprint arXiv:2405.21060*, 2024.

[41] Yann N Dauphin, Angela Fan, Michael Auli, and David Grangier. Language modeling with gated convolutional networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 933–941. JMLR. org, 2017.

[42] Mostafa Dehghani, Stephan Gouws, Oriol Vinyals, Jakob Uszkoreit, and Lukasz Kaiser. Universal transformers. In *International Conference on Learning Representations*, 2018.

[43] Zeyu Deng, Abla Kammoun, and Christos Thrampoulidis. A model of double descent for high-dimensional binary linear classification. *Information and Inference: A Journal of the IMA*, 11(2):435–495, 2022.

[44] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[45] Luc Devroye, László Györfi, and Gábor Lugosi. *A probabilistic theory of pattern recognition*, volume 31. Springer Science & Business Media, 2013.

[46] Nan Ding, Tomer Levinboim, Jialin Wu, Sebastian Goodman, and Radu Soricut. Causallm is not optimal for in-context learning. *arXiv preprint arXiv:2308.06912*, 2023.

[47] Yihe Dong, Jean-Baptiste Cordonnier, and Andreas Loukas. Attention is not all you need: Pure attention loses rank doubly exponentially with depth. In *International Conference on Machine Learning*, pages 2793–2803. PMLR, 2021.

[48] Simon Du, Jason Lee, Haochuan Li, Liwei Wang, and Xiyu Zhai. Gradient descent finds global minima of deep neural networks. In *International Conference on Machine Learning*, pages 1675–1685. PMLR, 2019.

[49] Simon S Du, Xiyu Zhai, Barnabas Poczos, and Aarti Singh. Gradient descent provably optimizes over-parameterized neural networks. *arXiv preprint arXiv:1810.02054*, 2018.

[50] Zhe Du, Haldun Balim, Samet Oymak, and Necmiye Ozay. Can transformers learn optimal filtering for unknown systems? *IEEE Control Systems Letters*, 7:3525–3530, 2023.

[51] Chen Dun, Mirian Hipolito Garcia, Guoqing Zheng, Ahmed Hassan Awadallah, Anastasios Kyrillidis, and Robert Sim. Sweeping heterogeneity with smart mops: Mixture of prompts for llm task adaptation. *arXiv preprint arXiv:2310.02842*, 2023.

[52] Karthik Duraisamy. Finite sample analysis and bounds of generalization error of gradient descent in in-context linear regression. *arXiv preprint arXiv:2405.02462*, 2024.

[53] Nouha Dziri, Ximing Lu, Melanie Sclar, Xiang Lorraine Li, Liwei Jian, Bill Yuchen Lin, Peter West, Chandra Bhagavatula, Ronan Le Bras, Jena D Hwang, et al. Faith and fate: Limits of transformers on compositionality. *arXiv preprint arXiv:2305.18654*, 2023.

[54] Benjamin L Edelman, Surbhi Goel, Sham Kakade, and Cyril Zhang. Inductive biases and variable creation in self-attention mechanisms. In *International Conference on Machine Learning*, pages 5793–5831. PMLR, 2022.

[55] Maryam Fazel. *Matrix rank minimization with applications*. PhD thesis, PhD thesis, Stanford University, 2002.

[56] Maryam Fazel, Ting Kei Pong, Defeng Sun, and Paul Tseng. Hankel matrix rank minimization with applications to system identification and realization. *SIAM Journal on Matrix Analysis and Applications*, 34(3):946–977, 2013.

[57] Daniel Y Fu, Tri Dao, Khaled Kamal Saab, Armin W Thomas, Atri Rudra, and Christopher Re. Hungry hungry hippos: Towards language modeling with state space models. In *The Eleventh International Conference on Learning Representations*, 2023.

[58] Hengyu Fu, Tianyu Guo, Yu Bai, and Song Mei. What can a single attention layer learn? a study through the random features lens. *arXiv preprint arXiv:2307.11353*, 2023.

[59] Shivam Garg, Dimitris Tsipras, Percy S Liang, and Gregory Valiant. What can transformers learn in-context? a case study of simple function classes. *Advances in Neural Information Processing Systems*, 35:30583–30598, 2022.

[60] Khashayar Gatmiry, Nikunj Saunshi, Sashank J Reddi, Stefanie Jegelka, and Sanjiv Kumar. Can looped transformers learn to implement multi-step gradient descent for in-context learning? In *Forty-first International Conference on Machine Learning*.

[61] GeminiTeam, Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.

[62] Angeliki Giannou, Shashank Rajput, Jy-yong Sohn, Kangwook Lee, Jason D Lee, and Dimitris Papailiopoulos. Looped transformers as programmable computers. *arXiv:2301.13196*, 2023.

[63] Surbhi Goel, Varun Kanade, Adam Klivans, and Justin Thaler. Reliably learning the relu in polynomial time. In *Conference on Learning Theory*, pages 1004–1042. PMLR, 2017.

[64] Surbhi Goel, Adam Klivans, Pasin Manurangsi, and Daniel Reichman. Tight hardness results for training depth-2 relu networks. *arXiv preprint arXiv:2011.13550*, 2020.

[65] Halil Alperen Gozeten, M Emrullah Ildiz, Xuechen Zhang, Hrayr Harutyunyan, Ankit Singh Rawat, and Samet Oymak. Continuous chain of thought enables parallel exploration and reasoning. *arXiv preprint arXiv:2505.23648*, 2025.

[66] Halil Alperen Gozeten, M Emrullah Ildiz, Xuechen Zhang, Mahdi Soltanolkotabi, Marco Mondelli, and Samet Oymak. Test-time training provably improves transformers as in-context learners. *arXiv preprint arXiv:2503.11842*, 2025.

[67] Alex Graves, Abdel-rahman Mohamed, and Geoffrey Hinton. Speech recognition with deep recurrent neural networks. In *Acoustics, speech and signal processing (icassp), 2013 ieee international conference on*, pages 6645–6649. IEEE, 2013.

[68] Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces. *arXiv preprint arXiv:2312.00752*, 2023.

[69] Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

[70] Michael Hahn and Navin Goyal. A theory of emergent in-context learning as implicit structure induction. *arXiv preprint arXiv:2303.07971*, 2023.

[71] Chi Han, Ziqi Wang, Han Zhao, and Heng Ji. In-context learning of large language models explained as kernel regression. *arXiv preprint arXiv:2305.12766*, 2023.

[72] Steve Hanneke and Samory Kpotufe. On the value of target data in transfer learning. *Advances in Neural Information Processing Systems*, 32, 2019.

[73] Sariel Har-Peled and Soham Mazumdar. On coresets for k-means and k-median clustering. In *Proceedings of the thirty-sixth annual ACM symposium on Theory of computing*, pages 291–300, 2004.

[74] Trevor Hastie, Robert Tibshirani, Jerome H Friedman, and Jerome H Friedman. *The elements of statistical learning: data mining, inference, and prediction*, volume 2. Springer, 2009.

[75] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[76] Geoffrey Hinton, Li Deng, Dong Yu, George E Dahl, Abdel-rahman Mohamed, Navdeep Jaitly, Andrew Senior, Vincent Vanhoucke, Patrick Nguyen, Tara N Sainath, et al. Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *IEEE Signal Processing Magazine*, 29(6):82–97, 2012.

[77] Noah Hollmann, Samuel Müller, Katharina Eggensperger, and Frank Hutter. Tabpfn: A transformer that solves small tabular classification problems in a second. *arXiv preprint arXiv:2207.01848*, 2022.

[78] Noah Hollmann, Samuel Müller, Lennart Purucker, Arjun Krishnakumar, Max Körfer, Shi Bin Hoo, Robin Tibor Schirrmeister, and Frank Hutter. Accurate predictions on small data with a tabular foundation model. *Nature*, 637(8045):319–326, 2025.

[79] Jiri Hron, Yasaman Bahri, Jascha Sohl-Dickstein, and Roman Novak. Infinite attention: Nngp and ntk for deep attention networks. In *International Conference on Machine Learning*, pages 4376–4386. PMLR, 2020.

[80] Daniel Hsu, Vidya Muthukumar, and Ji Xu. On the proliferation of support vectors in high dimensions. In *International Conference on Artificial Intelligence and Statistics*, pages 91–99. PMLR, 2021.

[81] Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022.

[82] Shima Imani, Liang Du, and Harsh Shrivastava. Mathprompter: Mathematical reasoning using large language models. *arXiv preprint arXiv:2303.05398*, 2023.

[83] Arthur Jacot, Franck Gabriel, and Clément Hongler. Neural tangent kernel: Convergence and generalization in neural networks. *arXiv preprint arXiv:1806.07572*, 2018.

[84] Aaron Jaech, Adam Kalai, Adam Lerer, Adam Richardson, Ahmed El-Kishky, Aiden Low, Alec Helyar, Aleksander Madry, Alex Beutel, Alex Carney, et al. Openai o1 system card. *arXiv preprint arXiv:2412.16720*, 2024.

[85] Michael Janner, Qiyang Li, and Sergey Levine. Reinforcement learning as one big sequence modeling problem. In *ICML 2021 Workshop on Unsupervised Reinforcement Learning*, 2021.

[86] Samy Jelassi, Michael Eli Sander, and Yuanzhi Li. Vision transformers provably learn spatial structure. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho, editors, *Advances in Neural Information Processing Systems*, 2022.

[87] Ziwei Ji, Miroslav Dudík, Robert E Schapire, and Matus Telgarsky. Gradient descent follows the regularization path for general losses. In *Conference on Learning Theory*, pages 2109–2136. PMLR, 2020.

[88] Ziwei Ji and Matus Telgarsky. Risk and parameter convergence of logistic regression. *arXiv preprint arXiv:1803.07300*, 2018.

[89] Ziwei Ji and Matus Telgarsky. The implicit bias of gradient descent on nonseparable data. In *Conference on Learning Theory*, pages 1772–1798. PMLR, 2019.

[90] Ziwei Ji and Matus Telgarsky. Risk and parameter convergence of logistic regression. 2019.

[91] Ziwei Ji and Matus Telgarsky. Directional convergence and alignment in deep learning. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 17176–17186. Curran Associates, Inc., 2020.

[92] Angelos Katharopoulos, Apoorv Vyas, Nikolaos Pappas, and François Fleuret. Transformers are rnns: Fast autoregressive transformers with linear attention. In *International Conference on Machine Learning*, pages 5156–5165. PMLR, 2020.

[93] Tobias Katsch. Gateloop: Fully data-controlled linear recurrence for sequence modeling. *arXiv preprint arXiv:2311.01927*, 2023.

[94] Hyunjik Kim, George Papamakarios, and Andriy Mnih. The lipschitz constant of self-attention. In *International Conference on Machine Learning*, pages 5562–5571. PMLR, 2021.

[95] Suhas Kowshik, Dheeraj Nagaraj, Prateek Jain, and Praneeth Netrapalli. Near-optimal offline and streaming algorithms for learning non-linear dynamical systems. *Advances in Neural Information Processing Systems*, 34:8518–8531, 2021.

[96] Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton. The cifar-10 dataset. *online: http://www. cs. toronto. edu/kriz/cifar. html*, 55(5), 2014.

[97] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[98] Andrew K Lampinen, Ishita Dasgupta, Stephanie CY Chan, Kory Matthewson, Michael Henry Tessler, Antonia Creswell, James L McClelland, Jane X Wang, and Felix Hill. Can language models learn from explanations in context? *arXiv preprint arXiv:2204.02329*, 2022.

[99] Jack Lanchantin, Shubham Toshniwal, Jason Weston, Arthur Szlam, and Sainbayar Sukhbaatar. Learning to reason and memorize with self-notes. *arXiv preprint arXiv:2305.00833*, 2023.

[100] Michael Laskin, Luyu Wang, Junhyuk Oh, Emilio Parisotto, Stephen Spencer, Richie Steigerwald, DJ Strouse, Steven Hansen, Angelos Filos, Ethan Brooks, et al. In-context reinforcement learning with algorithm distillation. *arXiv preprint arXiv:2210.14215*, 2022.

[101] Beatrice Laurent and Pascal Massart. Adaptive estimation of a quadratic functional by model selection. *Annals of statistics*, pages 1302–1338, 2000.

[102] Nayoung Lee, Kartik Sreenivasan, Jason D Lee, Kangwook Lee, and Dimitris Papailiopoulos. Teaching arithmetic to small transformers. *arXiv preprint arXiv:2307.03381*, 2023.

[103] Marc Lelarge and Léo Miolane. Asymptotic bayes risk for gaussian mixture in a semi-supervised setting. In *2019 IEEE 8th International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP)*, pages 639–643. IEEE, 2019.

[104] Brian Lester, Rami Al-Rfou, and Noah Constant. The power of scale for parameter-efficient prompt tuning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3045–3059, 2021.

[105] Yoav Levine, Noam Wies, Or Sharir, Hofit Bata, and Amnon Shashua. Limits to depth efficiencies of self-attention. In *Advances in Neural Information Processing Systems*, volume 33, pages 22640–22651, 2020.

[106] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474, 2020.

[107] Hongkang Li, Meng Wang, Sijia Liu, and Pin-Yu Chen. A theoretical understanding of shallow vision transformers: Learning, generalization, and sample complexity. *arXiv preprint arXiv:2302.06015*, 2023.

[108] Liunian Harold Li, Jack Hessel, Youngjae Yu, Xiang Ren, Kai-Wei Chang, and Yejin Choi. Symbolic chain-of-thought distillation: Small models can also" think" step-by-step. *arXiv preprint arXiv:2306.14050*, 2023.

[109] Mingchen Li, Mahdi Soltanolkotabi, and Samet Oymak. Gradient descent with early stopping is provably robust to label noise for overparameterized neural networks. In *International conference on artificial intelligence and statistics*, pages 4313–4324. PMLR, 2020.

[110] Xiang Lisa Li and Percy Liang. Prefix-tuning: Optimizing continuous prompts for generation. *arXiv preprint arXiv:2101.00190*, 2021.

[111] Yingcong Li, Xiangyu Chang, Muti Kara, Xiaofeng Liu, Amit Roy-Chowdhury, and Samet Oymak. When and how unlabeled data provably improve in-context learning. *arXiv preprint arXiv:2506.15329*, 2025.

[112] Yingcong Li, Yixiao Huang, Muhammed E Ildiz, Ankit Singh Rawat, and Samet Oymak. Mechanics of next token prediction with self-attention. In *International Conference on Artificial Intelligence and Statistics*, pages 685–693. PMLR, 2024.

[113] Yingcong Li, M Emrullah Ildiz, Dimitris Papailiopoulos, and Samet Oymak. Transformers as algorithms: Generalization and stability in in-context learning. In *International Conference on Machine Learning*, 2023.

[114] Yingcong Li, Mingchen Li, M Salman Asif, and Samet Oymak. Provable and efficient continual representation learning. *arXiv preprint arXiv:2203.02026*, 2022.

[115] Yingcong Li and Samet Oymak. Provable pathways: Learning multiple tasks over multiple paths. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 8701–8710, 2023.

[116] Yingcong Li, Ankit S Rawat, and Samet Oymak. Fine-grained analysis of in-context linear estimation: Data, architecture, and beyond. *Advances in Neural Information Processing Systems*, 37:138324–138364, 2024.

[117] Yingcong Li, Kartik Sreenivasan, Angeliki Giannou, Dimitris Papailiopoulos, and Samet Oymak. Dissecting chain-of-thought: Compositionality through in-context filtering and learning. *Advances in Neural Information Processing Systems*, 36, 2024.

[118] Yingcong Li, Davoud Ataee Tarzanagh, Ankit Singh Rawat, Maryam Fazel, and Samet Oymak. Gating is weighting: Understanding gated linear attention through in-context learning. *arXiv preprint arXiv:2504.04308*, 2025.

[119] Yuanzhi Li and Yingyu Liang. Learning overparameterized neural networks via stochastic gradient descent on structured data. *arXiv preprint arXiv:1808.01204*, 2018.

[120] Sen Lin, Peizhong Ju, Yingbin Liang, and Ness Shroff. Theory on forgetting and generalization of continual learning. In *International Conference on Machine Learning*, pages 21078–21100. PMLR, 2023.

[121] Zhouhan Lin, Minwei Feng, Cicero Nogueira dos Santos, Mo Yu, Bing Xiang, Bowen Zhou, and Yoshua Bengio. A structured self-attentive sentence embedding. In *International Conference on Learning Representations*, 2017.

[122] Ziqian Lin and Kangwook Lee. Dual operating modes of in-context learning. *arXiv preprint arXiv:2402.18819*, 2024.

[123] Dennis V Lindley and Adrian FM Smith. Bayes estimates for the linear model. *Journal of the Royal Statistical Society: Series B (Methodological)*, 34(1):1–18, 1972.

[124] Adam Liška, Germán Kruszewski, and Marco Baroni. Memorize or generalize? searching for a compositional rnn in a haystack. *arXiv preprint arXiv:1802.06467*, 2018.

[125] Bingbin Liu, Jordan T. Ash, Surbhi Goel, Akshay Krishnamurthy, and Cyril Zhang. Transformers learn shortcuts to automata. In *The Eleventh International Conference on Learning Representations*, 2023.

[126] Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Computing Surveys*, 55(9):1–35, 2023.

[127] Lennart Ljung. System identification. In *Signal analysis and prediction*, pages 163–173. Springer, 1998.

[128] Kaifeng Lyu and Jian Li. Gradient descent maximizes the margin of homogeneous neural networks. *arXiv preprint arXiv:1906.05890*, 2019.

[129] Arvind V. Mahankali, Tatsunori Hashimoto, and Tengyu Ma. One step of gradient descent is provably the optimal in-context learner with one layer of linear self-attention. In *The Twelfth International Conference on Learning Representations*, 2024.

[130] Sadegh Mahdavi, Renjie Liao, and Christos Thrampoulidis. Revisiting the equivalence of in-context learning and gradient descent: The impact of data distribution. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7410–7414. IEEE, 2024.

[131] Andreas Maurer. A vector-contraction inequality for rademacher complexities. In *International Conference on Algorithmic Learning Theory*, pages 3–17. Springer, 2016.

[132] Sewon Min, Xinxi Lyu, Ari Holtzman, Mikel Artetxe, Mike Lewis, Hannaneh Hajishirzi, and Luke Zettlemoyer. Rethinking the role of demonstrations: What makes in-context learning work? *arXiv preprint arXiv:2202.12837*, 2022.

[133] Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar. *Foundations of machine learning*. MIT press, 2018.

[134] Andrea Montanari, Feng Ruan, Youngtak Sohn, and Jun Yan. The generalization error of max-margin linear classifiers: High-dimensional asymptotics in the overparametrized regime. *arXiv preprint arXiv:1911.01544*, 2019.

[135] Siyuan Mu and Sen Lin. A comprehensive survey of mixture-of-experts: Algorithms, theory, and applications. *arXiv preprint arXiv:2503.07137*, 2025.

[136] Niklas Muennighoff, Zitong Yang, Weijia Shi, Xiang Lisa Li, Li Fei-Fei, Hannaneh Hajishirzi, Luke Zettlemoyer, Percy Liang, Emmanuel Candès, and Tatsunori Hashimoto. s1: Simple test-time scaling. *arXiv preprint arXiv:2501.19393*, 2025.

[137] Samuel Müller, Noah Hollmann, Sebastian Pineda Arango, Josif Grabocka, and Frank Hutter. Transformers can do bayesian inference. *arXiv preprint arXiv:2112.10510*, 2021.

[138] Vidya Muthukumar, Adhyyan Narang, Vignesh Subramanian, Mikhail Belkin, Daniel Hsu, and Anant Sahai. Classification vs regression in overparameterized regimes: Does the loss function matter? *The Journal of Machine Learning Research*, 22(1):10104–10172, 2021.

[139] Preetum Nakkiran, Gal Kaplun, Yamini Bansal, Tristan Yang, Boaz Barak, and Ilya Sutskever. Deep double descent: Where bigger models and more data hurt. *Journal of Statistical Mechanics: Theory and Experiment*, 2021(12):124003, 2021.

[140] Sharan Narang, Colin Raffel, Katherine Lee, Adam Roberts, Noah Fiedel, and Karishma Malkan. Wt5?! training text-to-text models to explain their predictions. *arXiv preprint arXiv:2004.14546*, 2020.

[141] Ojash Neopane. Lecture notes on high-dimensional statistics, 2018.

[142] Behnam Neyshabur, Ryota Tomioka, Ruslan Salakhutdinov, and Nathan Srebro. Geometry of optimization and implicit regularization in deep learning. *arXiv preprint arXiv:1705.03071*, 2017.

[143] Tan Minh Nguyen, Tam Minh Nguyen, Nhat Ho, Andrea L Bertozzi, Richard Baraniuk, and Stanley Osher. A primal-dual framework for transformers and neural networks. In *The Eleventh International Conference on Learning Representations*, 2023.

[144] Lorenzo Noci, Chuning Li, Mufan Bill Li, Bobby He, Thomas Hofmann, Chris Maddison, and Daniel M Roy. The shaped transformer: Attention models in the infinite depth-and-width limit. *arXiv preprint arXiv:2306.17759*, 2023.

[145] Maxwell Nye, Anders Johan Andreassen, Guy Gur-Ari, Henryk Michalewski, Jacob Austin, David Bieber, David Dohan, Aitor Lewkowycz, Maarten Bosma, David Luan, et al. Show your work: Scratchpads for intermediate computation with language models. *arXiv preprint arXiv:2112.00114*, 2021.

[146] OpenAI. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

[147] Samet Oymak and Babak Hassibi. New null space results and recovery thresholds for matrix rank minimization. *arXiv preprint arXiv:1011.6326*, 2010.

[148] Samet Oymak, Karthik Mohan, Maryam Fazel, and Babak Hassibi. A simplified approach to recovery conditions for low rank matrices. In *2011 IEEE International Symposium on Information Theory Proceedings*, pages 2318–2322. IEEE, 2011.

[149] Samet Oymak, Ankit Singh Rawat, Mahdi Soltanolkotabi, and Christos Thrampoulidis. On the role of attention in prompt-tuning. In *International Conference on Machine Learning*, 2023.

[150] Samet Oymak and Mahdi Soltanolkotabi. Overparameterized nonlinear learning: Gradient descent takes the shortest path? In *International Conference on Machine Learning*, pages 4951–4960. PMLR, 2019.

[151] Samet Oymak and Mahdi Soltanolkotabi. Toward moderate overparameterization: Global convergence guarantees for training shallow neural networks. *IEEE Journal on Selected Areas in Information Theory*, 1(1):84–105, 2020.

[152] Ankur Parikh, Oscar Täckström, Dipanjan Das, and Jakob Uszkoreit. A decomposable attention model for natural language inference. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2249–2255, Austin, Texas, November 2016. Association for Computational Linguistics.

[153] Romain Paulus, Caiming Xiong, and Richard Socher. A deep reinforced model for abstractive summarization. In *International Conference on Learning Representations*, 2018.

[154] Bo Peng, Eric Alcaide, Quentin Anthony, Alon Albalak, Samuel Arcadinho, Stella Biderman, Huanqi Cao, Xin Cheng, Michael Chung, Matteo Grella, et al. Rwkv: Reinventing rnns for the transformer era. *arXiv preprint arXiv:2305.13048*, 2023.

[155] Bo Peng, Daniel Goldstein, Quentin Anthony, Alon Albalak, Eric Alcaide, Stella Biderman, Eugene Cheah, Teddy Ferdinan, Haowen Hou, Przemysław Kazienko, et al. Eagle and finch: RWKV with matrix-valued states and dynamic recurrence. *arXiv preprint arXiv:2404.05892*, 2024.

[156] Hao Peng, Nikolaos Pappas, Dani Yogatama, Roy Schwartz, Noah A Smith, and Lingpeng Kong. Random feature attention. *arXiv preprint arXiv:2103.02143*, 2021.

[157] Jorge Pérez, Pablo Barceló, and Javier Marinkovic. Attention is turing complete. *The Journal of Machine Learning Research*, 22(1):3463–3497, 2021.

[158] Luis Perez, Lizi Ottens, and Sudharshan Viswanathan. Automatic code generation using pre-trained language models. *arXiv preprint arXiv:2102.10535*, 2021.

[159] Gianluigi Pillonetto, Tianshi Chen, Alessandro Chiuso, Giuseppe De Nicolao, and Lennart Ljung. Regularized linear system identification using atomic, nuclear and kernel-based norms: The role of the stability constraint. *Automatica*, 69:137–149, 2016.

[160] Ofir Press and Lior Wolf. Using the output embedding to improve language models. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*. Association for Computational Linguistics, 2017.

[161] Jorge Pérez, Javier Marinković, and Pablo Barceló. On the Turing completeness of modern neural network architectures. In *International Conference on Learning Representations*, 2019.

[162] Zhen Qin, Songlin Yang, Weixuan Sun, Xuyang Shen, Dong Li, Weigao Sun, and Yiran Zhong. Hgrn2: Gated linear rnns with state expansion. *arXiv preprint arXiv:2404.07904*, 2024.

[163] Jingang Qu, David Holzmüller, Gaël Varoquaux, and Marine Le Morvan. Tabicl: A tabular foundation model for in-context learning on large data. *arXiv preprint arXiv:2502.05564*, 2025.

[164] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

[165] Jack W Rae, Sebastian Borgeaud, Trevor Cai, Katie Millican, Jordan Hoffmann, Francis Song, John Aslanides, Sarah Henderson, Roman Ring, Susannah Young, et al. Scaling language models: Methods, analysis & insights from training gopher. *arXiv preprint arXiv:2112.11446*, 2021.

[166] Benjamin Recht, Maryam Fazel, and Pablo A Parrilo. Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization. *SIAM review*, 52(3):471–501, 2010.

[167] Benjamin Recht, Weiyu Xu, and Babak Hassibi. Null space conditions and thresholds for rank minimization. *Mathematical programming*, 127:175–202, 2011.

[168] Dominic Richards, Jaouad Mourtada, and Lorenzo Rosasco. Asymptotics of ridge (less) regression under general source condition. In *International Conference on Artificial Intelligence and Statistics*, pages 3889–3897. PMLR, 2021.

[169] Saharon Rosset, Ji Zhu, and Trevor Hastie. Margin maximizing loss functions. *Advances in neural information processing systems*, 16, 2003.

[170] Himanshu Sahni, Saurabh Kumar, Farhan Tejani, and Charles Isbell. Learning to compose skills. *arXiv preprint arXiv:1711.11289*, 2017.

[171] Yahya Sattar and Samet Oymak. Non-asymptotic and accurate learning of nonlinear dynamical systems. *Journal of Machine Learning Research*, 23(140):1–49, 2022.

[172] Imanol Schlag, Kazuki Irie, and Jürgen Schmidhuber. Linear transformers are secretly fast weight programmers. In *International Conference on Machine Learning*, pages 9355–9366. PMLR, 2021.

[173] Wei Shen, Ruida Zhou, Jing Yang, and Cong Shen. On the training convergence of transformers for in-context classification. *arXiv preprint arXiv:2410.11778*, 2024.

[174] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[175] Charlie Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. Scaling llm test-time compute optimally can be more effective than scaling model parameters. *arXiv preprint arXiv:2408.03314*, 2024.

[176] Charlie Snell, Ruiqi Zhong, Dan Klein, and Jacob Steinhardt. Approximating how single head attention learns. *arXiv preprint arXiv:2103.07601*, 2021.

[177] Daniel Soudry, Elad Hoffer, Mor Shpigel Nacson, Suriya Gunasekar, and Nathan Srebro. The implicit bias of gradient descent on separable data. *The Journal of Machine Learning Research*, 19(1):2822–2878, 2018.

[178] Nathan Srebro, Jason Rennie, and Tommi Jaakkola. Maximum-margin matrix factorization. *Advances in neural information processing systems*, 17, 2004.

[179] Arun Suggala, Adarsh Prasad, and Pradeep K Ravikumar. Connecting optimization and regularization paths. *Advances in Neural Information Processing Systems*, 31, 2018.

[180] Fei Sun, Jun Liu, Jian Wu, Changhua Pei, Xiao Lin, Wenwu Ou, and Peng Jiang. Bert4rec: Sequential recommendation with bidirectional encoder representations from transformer. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, pages 1441–1450, 2019.

[181] Yutao Sun, Li Dong, Shaohan Huang, Shuming Ma, Yuqing Xia, Jilong Xue, Jianyong Wang, and Furu Wei. Retentive network: A successor to transformer for large language models. *arXiv preprint arXiv:2307.08621*, 2023.

[182] Yutao Sun, Li Dong, Yi Zhu, Shaohan Huang, Wenhui Wang, Shuming Ma, Quanlu Zhang, Jianyong Wang, and Furu Wei. You only cache once: Decoder-decoder architectures for language models. *arXiv preprint arXiv:2405.05254*, 2024.

[183] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. *Advances in neural information processing systems*, 27, 2014.

[184] Robert Tarjan. Depth-first search and linear graph algorithms. *SIAM Journal on Computing*, 1(2):146–160, 1972.

[185] Davoud Ataee Tarzanagh, Yingcong Li, Christos Thrampoulidis, and Samet Oymak. Transformers as support vector machines. *arXiv preprint arXiv:2308.16898*, 2023.

[186] Davoud Ataee Tarzanagh, Yingcong Li, Xuechen Zhang, and Samet Oymak. Margin maximization in attention mechanism. *arXiv preprint arXiv:2306.13596*, 2023.

[187] Davoud Ataee Tarzanagh, Yingcong Li, Xuechen Zhang, and Samet Oymak. Max-margin token selection in attention mechanism. *Advances in Neural Information Processing Systems*, 36, 2024.

[188] Christos Thrampoulidis, Samet Oymak, and Mahdi Soltanolkotabi. Theoretical insights into multiclass classification: A high-dimensional asymptotic view. *Advances in Neural Information Processing Systems*, 33:8907–8920, 2020.

[189] Yuandong Tian, Yiping Wang, Beidi Chen, and Simon Du. Scan and snap: Understanding training dynamics and token composition in 1-layer transformer. *arXiv:2305.16380*, 2023.

[190] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

[191] Nilesh Tripuraneni, Michael Jordan, and Chi Jin. On the theory of transfer learning: The importance of task diversity. *Advances in Neural Information Processing Systems*, 33:7852–7862, 2020.

[192] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[193] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, volume 30, 2017.

[194] Petar Veličković and Charles Blundell. Neural algorithmic reasoning. *Patterns*, 2(7):100273, 2021.

[195] Roman Vershynin. *High-dimensional probability: An introduction with applications in data science*, volume 47. Cambridge university press, 2018.

[196] Johannes Von Oswald, Eyvind Niklasson, Ettore Randazzo, João Sacramento, Alexander Mordvintsev, Andrey Zhmoginov, and Max Vladymyrov. Transformers learn in-context by gradient descent. In *International Conference on Machine Learning*, pages 35151–35174. PMLR, 2023.

[197] Johannes von Oswald, Eyvind Niklasson, Maximilian Schlegel, Seijin Kobayashi, Nicolas Zucchet, Nino Scherrer, Nolan Miller, Mark Sandler, Max Vladymyrov, Razvan Pascanu, et al. Uncovering mesa-optimization algorithms in transformers. *arXiv preprint arXiv:2309.05858*, 2023.

[198] Martin J Wainwright. *High-dimensional statistics: A non-asymptotic viewpoint*, volume 48. Cambridge University Press, 2019.

[199] Ke Wang and Christos Thrampoulidis. Binary classification of gaussian mixtures: Abundance of support vectors, benign overfitting, and regularization. *SIAM Journal on Mathematics of Data Science*, 4(1):260–284, 2022.

[200] Ruochen Wang, Sohyun An, Minhao Cheng, Tianyi Zhou, Sung Ju Hwang, and Cho-Jui Hsieh. One prompt is not enough: Automated construction of a mixture-of-expert prompts. *arXiv preprint arXiv:2407.00256*, 2024.

[201] Sinong Wang, Belinda Z Li, Madian Khabsa, Han Fang, and Hao Ma. Linformer: Self-attention with linear complexity. *arXiv preprint arXiv:2006.04768*, 2020.

[202] Colin Wei, Yining Chen, and Tengyu Ma. Statistically meaningful approximation: a case study on approximating turing machines with transformers. *Advances in Neural Information Processing Systems*, 35:12071–12083, 2022.

[203] Jason Wei, Maarten Bosma, Vincent Y Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. Finetuned language models are zero-shot learners. *arXiv preprint arXiv:2109.01652*, 2021.

[204] Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, et al. Emergent abilities of large language models. *arXiv preprint arXiv:2206.07682*, 2022.

[205] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Ed Chi, Quoc Le, and Denny Zhou. Chain of thought prompting elicits reasoning in large language models. *arXiv preprint arXiv:2201.11903*, 2022.

[206] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*, 2019.

[207] Jingfeng Wu, Difan Zou, Zixiang Chen, Vladimir Braverman, Quanquan Gu, and Peter L Bartlett. How many pretraining tasks are needed for in-context learning of linear regression? *arXiv preprint arXiv:2310.08391*, 2023.

[208] Sang Michael Xie, Aditi Raghunathan, Percy Liang, and Tengyu Ma. An explanation of in-context learning as implicit bayesian inference. In *International Conference on Learning Representations*, 2022.

[209] Greg Yang. Tensor programs ii: Neural tangent kernel for any architecture. *arXiv preprint arXiv:2006.14548*, 2020.

[210] Songlin Yang, Bailin Wang, Yikang Shen, Rameswar Panda, and Yoon Kim. Gated linear attention transformers with hardware-efficient training. *arXiv preprint arXiv:2312.06635*, 2023.

[211] Han-Jia Ye, Si-Yang Liu, and Wei-Lun Chao. A closer look at tabpfn v2: Strength, limitation, and extension. *arXiv preprint arXiv:2502.17361*, 2025.

[212] Zheng Yuan, Hongyi Yuan, Chuanqi Tan, Wei Wang, and Songfang Huang. How well do large language models perform in arithmetic tasks? *arXiv preprint arXiv:2304.02015*, 2023.

[213] Chulhee Yun, Srinadh Bhojanapalli, Ankit Singh Rawat, Sashank Reddi, and Sanjiv Kumar. Are transformers universal approximators of sequence-to-sequence functions? In *International Conference on Learning Representations*, 2019.

[214] Ruiqi Zhang, Spencer Frei, and Peter L Bartlett. Trained transformers learn linear models in-context. *Journal of Machine Learning Research*, 25(49):1–55, 2024.

[215] Tong Zhang and Bin Yu. Boosting with early stopping: Convergence and consistency. *Annals of Statistics*, page 1538, 2005.

[216] Xiao Zhang, Yaodong Yu, Lingxiao Wang, and Quanquan Gu. Learning one-hidden-layer relu networks via gradient descent. In *The 22nd international conference on artificial intelligence and statistics*, pages 1524–1534. PMLR, 2019.

[217] Xuechen Zhang, Zijian Huang, Yingcong Li, Chenshun Ni, Jiasi Chen, and Samet Oymak. Bread: Branched rollouts from expert anchors bridge sft & rl for reasoning. *arXiv preprint arXiv:2506.17211*, 2025.

[218] Xuechen Zhang, Zijian Huang, Chenshun Ni, Ziyang Xiong, Jiasi Chen, and Samet Oymak. Making small language models efficient reasoners: Intervention, supervision, reinforcement. *arXiv preprint arXiv:2505.07961*, 2025.

[219] Yu Zhang and Qiang Yang. A survey on multi-task learning. *IEEE Transactions on Knowledge and Data Engineering*, 2021.

[220] Qinqing Zheng, Amy Zhang, and Aditya Grover. Online decision transformer. In *Proceedings of the 39th International Conference on Machine Learning*, 2022.

[221] Guorui Zhou, Xiaoqiang Zhu, Chenru Song, Ying Fan, Han Zhu, Xiao Ma, Yanghui Yan, Junqi Jin, Han Li, and Kun Gai. Deep interest network for click-through rate prediction. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 1059–1068, 2018.

[222] Hattie Zhou, Azade Nova, Hugo Larochelle, Aaron Courville, Behnam Neyshabur, and Hanie Sedghi. Teaching algorithmic reasoning via in-context learning. *arXiv preprint arXiv:2211.09066*, 2022.

[223] Yucheng Zhou, Xiang Li, Qianning Wang, and Jianbing Shen. Visual in-context learning for large vision-language models. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 15890–15902, 2024.

[224] Itamar Zimerman, Ameen Ali, and Lior Wolf. A unified implicit attention formulation for gated-linear recurrent sequence models. *arXiv preprint arXiv:2405.16504*, 2024.

[225] Chang Zong, Jian Shao, Weiming Lu, and Yueting Zhuang. Stock movement prediction with multimodal stable fusion via gated cross-attention mechanism. *arXiv preprint arXiv:2406.06594*, 2024.