Three Papers in the Applied Use of Machine Learning and Artificial Intelligence Models for the Analysis of Political Text Data

by

Mitchell Bosley

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Political Science)
in the University of Michigan
2024

Doctoral Committee:

Professor Walter Mebane, Co-Chair
Professor George Tsebelis Co-Chair
Professor Ken Kollman
Professor Charles Shipan
Professor Yuki Shiraito

Mitchell Bosley
mcbosley@umich.edu
ORCID iD: 0000-0002-9172-966X

© Mitchell Bosley 2024

DEDICATION

To my wife, Veronica, for your unwavering love and support, to my dog, Oscar, for your constant companionship, and to my family, for always believing in me.

ACKNOWLEDGEMENTS

This dissertation would not have been possible without the support, guidance, and encouragement of many people. I am deeply grateful to everyone who has helped me throughout this journey, both professionally and personally.

First and foremost, I would like to thanks my committee chairs, George Tsebelis and Walter Mebane. Their mentorship, guidance, and support have been invaluable to me throughout this process. They have challenged me to think critically, pushed me to refine my ideas, and helped me navigate the complexities of this project. Their feedback, insight, and encouragement were instrumental in shaping this dissertation, and I am profoundly thankful for their mentorship.

I would like to express my deepest gratitude to my advisors Ken Kollman, Yuki Shiraito, and Charles Shipan, whose intellectual guidance and constant support were invaluable. Ken has been a constant source of inspiration and encouragement from the very first day I joined the University of Michigan. His unwavering belief in my abilities, even when I doubted myself, gave me the confidence to pursue ambitious research questions and to push the boundaries of my field. Yuki's feedback, insight, and encouragement were instrumental in shaping this dissertation, and I am profoundly thankful for his mentorship. Charles has been a source of support and guidance whenever I needed it, and I am grateful for the time and energy he has invested in my development as a scholar.

I would also like to thank my co-authors Saki Kuzushima and Ted Enamorado for their collaboration and support. The work that we have done together has been incredibly rewarding, and an integral part of my development as a scholar. I am grateful for their insights, feedback, and friendship.

I am indebted to my colleagues and peers for their support and collaboration. The stimulating discussions, debates, and collaborations with my peers helped me grow as a scholar, and I value the friendships and professional relationships that have been formed during this time.

A special thank you goes to the University of Michigan for providing the resources and environment that allowed me to pursue this research. I am grateful in particular for the funding support from the Ford Fellowship, which made this work possible.

On a more personal note, I would like to extend my heartfelt thanks to my wife Veronica, whose unwavering support has been my foundation throughout this journey. Your belief in me, even during the most challenging moments, gave me the strength to keep pushing forward. I am deeply thankful for your love and encouragement.

Finally, I want to express my gratitude to my friends, both near and far, who have provided laughter, perspective, and support. To everyone who contributed to this project, directly or indirectly, I offer my sincere thanks. This dissertation is as much a reflection of your support as it is my work, and I am incredibly grateful for each of you.

PREFACE

This dissertation began with a central question: under what conditions is legislative obstruction tolerated, and when does it lead to changes in parliamentary rules? My initial goal was to measure obstruction in legislative speech and explore how legislatures respond to tactics such as filibusters, delaying motions, and procedural challenges. While these theoretical questions did not remain the central focus of the final dissertation, they deeply influenced the methods I developed in the first two papers. These methods, initially designed for measuring obstruction, evolved into tools for analyzing broader aspects of legislative discourse.

At the start, my research was driven by the challenge of quantifying obstruction in legislative debates—an inherently complex and contentious behavior that often defies easy categorization. The need to capture both the rhetorical strategies and underlying intent of obstructionist tactics posed a significant challenge, especially at scale. As a result, my early focus was on developing computational methods, specifically through artificial intelligence, that could handle this dual demand for depth and scalability.

A turning point in my research came in late November 2022, with the release of generative AI models like ChatGPT. These models transformed the landscape of artificial intelligence and opened up a new set of possibilities for my work. Their ability to generate human-like text allowed me to refine the prompt-engineering techniques that I had originally designed to analyze obstruction. Generative AI offered both a new approach to scaling qualitative analysis and a deeper level of sophistication in parsing complex forms of political speech. This development significantly broadened the scope of my research, allowing me to apply these methods to a wider range of parliamentary dynamics beyond obstruction alone.

The initial focus on obstruction was critical to shaping the methodological innovations in the first two papers. The prompt-engineering framework that I developed was originally intended to scale the analysis of obstructionist speech, but it evolved into a more generalizable tool for coding various qualitative dimensions of deliberation. Similarly, the graph-based approach in the second paper, which began as a method for mapping obstruction in debates, expanded into a framework for visualizing the full structure of parliamentary discourse. In both cases, the initial challenge of capturing obstruction at scale laid the groundwork for the broader applications of these tools.

As my research progressed, the intellectual challenges of measuring obstruction continued to inform the methods, even as the focus broadened. Obstruction does not occur in isolation; it is intricately linked to the strategies and dynamics within legislative bodies. This realization allowed me to adapt the methods for capturing these broader dynamics, resulting in tools that are flexible enough to be applied to various aspects of legislative speech, including but not limited to obstruction.

Reflecting on the development of this dissertation, I can see how the introduction of generative AI models provided new opportunities for enhancing the analytical power of my methods. These tools enabled me to overcome many of the limitations that initially constrained my research, allowing for more sophisticated and scalable analyses. While the theoretical focus on obstruction became less central as the research evolved, the methodologies retain their origins in the challenge of understanding how speech acts shape the function of legislative bodies.

Looking ahead, I am excited by the possibilities that these methods open up—not only for measuring obstruction but for exploring a wide range of dynamics in parliamentary speech. The tools I have developed provide new avenues for analyzing how legislative bodies function, how debates unfold, and how speech acts influence the rules and practices of governance. The emergence of generative AI during the course of this research was a turning point, and I believe that these innovations will continue to shape the future of computational political science. This dissertation reflects my commitment to using these tools to explore political questions with greater depth and scale, and I hope that this work will enable more sophisticated research in legislative studies.

TABLE OF CONTENTS

DEDICATION ii

ACKNOWLEDGEMENTS iii

PREFACE v

LIST OF FIGURES ix

LIST OF TABLES xiii

LIST OF APPENDICES xiv

ABSTRACT xv

CHAPTER

1 Introduction 1

2 Towards Qualitative Measurement at Scale: A Prompt-Engineering Framework for Large-Scale Analysis of Deliberative Quality in Parliamentary Debates 5

2.1 Introduction 5
2.2 Previous Work 7
2.3 Methodology 11
2.4 Results 21
2.5 Discussion 27
2.6 Conclusion 28

3 LegisGraph: A Graph-Based Method for Analyzing Legislative Debates Using Large Language Models 30

3.1 Introduction 30
3.2 Previous Work 32
3.3 Methods 36
3.4 Applications 54
3.5 Discussion 70
3.6 Conclusion 71

4 Improving Probabilistic Models in Text Classification via Active Learning 73

4.1 Introduction 73
4.2 Machine Learning Approaches to Text Classification 75
4.3 The Method 79
4.4 Validation Performance 85
4.5 Reanalysis with Fewer Human Annotations 91
4.6 Discussion 95
4.7 Conclusion 100
5 Conclusion 101

APPENDICES 104

BIBLIOGRAPHY 205

LIST OF FIGURES

FIGURE

2.1 High level overview of the DQI annotation prompt. 13
2.2 Excerpt of DQI Theoretical Foundation Provided To The Model 16
2.3 Excerpt of DQI Dimension Definitions and Scoring Criteria Provided To The Model 17
2.4 Output Format for DQI Annotations Provided To The Model 18
2.5 Closing Instructions for Speech Annotation Prompt 19
2.6 Comparison of Model Performance for Different Numbers of In-Context Learning (ICL) Examples 22
2.7 Model Performance (Accuracy) vs Model Cost 23

3.1 **Structured Graph Representation of the Debate.** This graph visually represents the structured output generated by the LLM, capturing the relationships between parties (purple nodes) speakers (light blue nodes), arguments (yellow nodes), subjects (green nodes), debate (dark blue node), and topics (red nodes). Each edge represents a relationship between nodes: party-speaker relationships, speaker-argument relationships, argument-subject relationships, and inter-argument relationships. The edges are labeled to indicate the nature of the relationship, such as makes, addresses, contradicts, and supports. 49
3.2 **Distribution of High-Level Topics in Canadian Parliamentary Debates.** This plot shows the relative frequency of each high-level topics (such as Economic Policy, Social Policy, Procedure, etc) across the three parliaments in the sample. 59
3.3 **Most Frequently Discussed Subjects in Social Policy Debates.** This plot shows the most frequently discussed subjects within the Social Policy topic across all three parliaments in the sample. 60
3.4 **Discourse Quality Index (DQI) Analysis**. (a) shows the distribution of DQI scores across all speeches in the three sampled parliaments, while (b) illustrates the variability of DQI scores per debate. 62
3.5 **Discourse Quality Index (DQI) by Party**. This plot shows the average DQI scores for each party across the three parliaments in the sample. 64
3.6 Speaker-Argument Network for Parl. 12 66
3.7 Speaker-Argument Network for Parl. 24 67
3.8 Speaker-Argument Network for Parl. 34 68

4.1 **Labeling: Passive vs Active Learning.** Panel A presents a corpus where a classifier based on term frequencies of Spending" and Gridlock" is utilized to categorize unlabeled (U) documents as political (P) and non-political (N). Panel B depicts a passive learning approach where the next document to be labeled is randomly selected. In contrast, Panel C demonstrates active learning, where obtaining the true label of the U located in the region of uncertainty for the classifier (shaded region) is prioritized, as it provides more informative insights into learning the decision boundary between P and N. 77
4.2 **Comparison of Classification Results with Random and Active Versions of *activeText* and DistilBERT** 88
4.3 **Comparison of Classification and Time Results across *activeText*and DistilBERT** 89
4.4 **Classification Results of *activeText* with and without Keywords** 90
4.5 **Replication of Figure 3 in Gohdes (2020): Expected Proportion of Target Killings, Given Internet Accessibility and Whether a Region is Inhabitated by the Alawi Minority.** The results from *activeText* are presented in the left panel and those of Gohdes (2020) are on the right. 94
4.6 **Replication of Figure 1 in Park et al. (2020): The Relationship Between Information Density and Average Sentiment Score.** 95
4.7 **Bias of the Empirical Risk for Labeled Data (left panel) and Out-of-sample Classification Performance (right panel) of *activeText*, *activeText*+LURE. For each panel, the x-axis represents the number of documents labeled, and the y-axis represents the average bias and average out-of-sample F1 score across 100 Monte Carlo simulations. Shaded areas represent the 95% confidence intervals across Monte Carlo simulations.** 98

B.1 Example Mermaid Graph Representation of a Legislative Debate. The graph shows the structure of the debate, including topics, subjects, parties, speakers, arguments, and procedural motions. Purple nodes represent parties, blue nodes speakers, yellow nodes arguments, green nodes subjects, red nodes topics, and orange nodes procedural motions. Solid lines represent relationships between nodes, while dotted lines indicate support or contradiction between arguments. 149

C.1 **Graphical Representation of the Mixture Model with Two Classes.** Hyperparameters are represented by shaded rectangles, while unobserved random variables and parameters are depicted by unshaded circles. Observed data is displayed within shaded circles. Larger unshaded rectangular plates are used to denote the range of indices for different types of data and parameters. 182
C.2 **Graphical Representation of the Model for Multiclass Classification.** Hyperparameters are represented by shaded rectangles, while unobserved random variables and parameters are depicted by unshaded circles. Observed data is displayed within shaded circles. Larger unshaded rectangular plates are used to denote the range of indices for different types of data and parameters. 184

C.3 **Multiclass Classification Results.** The darker lines show the results with *activeText* and the lighter lines show the results with SVM. The solid lines use active sampling to decide the next set of documents to be labeled, and the dashed lines use random (passive) sampling. The y-axis indicates the out-of-sample F1 score and the x-axis show the number of sampling steps. The left column shows the results on BBC corpus, where the target classes are "Politics," "Entertainment," "Business," "Sports," and "Technology." "Politics" class has 5% of the total dataset, and the rest 95% is evenly split across the rest of classes. The right column shows the results on the Supreme Court corpus, where the target classes are "Criminal Procedure" (32.4% of the corpus), "Civil Rights" (21.4%), "Economic Activity" (22.2%), "Judicial Power" (15.4%), "First Amendment (8.6%)." In our model, we set the number of classes to be the same as the classification categories and linked each class to one classification category. *activeText* performs the best across the four specifications on both corpora. 187

C.4 **Binary Classification Results with 2 and 5 Classes.** The darker lines show the results with 5 classes and the lighter lines show 2 classes. The columns correspond to various proportions of positive labels in the corpus. The y-axis indicates the out-of-sample F1 score and the x-axis show the number of sampling steps. For binary classification, using multiple classes improves the classification performance when the number of classes matches the data generating process. 191

C.5 **Binary Classification Results with Multiple Classes and Keywords.** The rows correspond to different datasets and the columns correspond to various proportions of positively labeled documents in the corpus. The y-axis indicates the out-of-sample F1 score and the x-axis show the number of sampling steps. The linetype show whether keywords are supplied: the solid lines show the results with keywords and the dashed lines without keywords. The colors show the number of classes in the mixture model: the darker lines show the results with 5 classes and the lighter lines with 2 classes. Using 5 classes leads to as good or slightly better performance than using 2 classes. The performance improvement is the largest with the BBC corpus, which consists of 5 news topic categories. Likewise, our mixture models with keywords leads to as good or better performance than the models without keywords. The improvement is the largest with the human rights corpus, where the number of words per document is the smallest. 192

C.6 Bias of the Empirical Risk for Labeled Data (left panel) and Out-of-sample Classification Performance (right panel) of *activeText*, *activeText*+LURE. For each panel, the x-axis represents the number of documents labeled, and the y-axis represents the average bias and average out-of-sample F1 score across 100 Monte Carlo simulations. Shaded areas represent the 95% confidence intervals across Monte Carlo simulations. 197

C.7 Wikipedia Toxic Comments: Bias of the Empirical Risk for Labeled Data (left panel) and Out-of-sample Classification Performance (right panel) of *activeText*, *activeText*+LURE. The reference class is Toxic comment (9% of all documents). For each panel, the x-axis represents the number of documents labeled, and the y-axis represents the average bias and average out-of-sample F1 score across 100 Monte Carlo simulations. Shaded areas represent the 95% confidence intervals across Monte Carlo simulations. 198
C.8 **Classification Results with Mislabels in Active Document Labeling** The rows correspond to different datasets. The y-axis indicates the out-of-sample F1 score and the x-axis shows the number of sampling steps. 20 documents are labeled at each sampling step. The colors correspond to different levels of mislabels in the labeling of documents. We find that as the proportion of mislabels increases, the classification performance of *activeText* decreases. . . . 200
C.9 **Bias in the Proportion of Documents in the Reference Class as a result of Mislabels in Active Document Labeling** The rows correspond to different datasets. The y-axis indicates the absolute value of the out-of-sample bias and the x-axis shows the number of sampling steps. 20 documents are labeled at each sampling step. The colors correspond to different levels of mislabels in the labeling of documents. We find that as the proportion of mislabels increases, the magnitude of the bias increases as well. 202
C.10 **Classification Results with Mislabels in Active Keywords** The rows correspond to different datasets and the columns correspond to two values of γ, which controls the degree of keyword upweighting. The y-axis indicates the out-of-sample F1 score and the x-axis shows the number of sampling steps. The lines correspond to different levels of mislabels at the keyword labeling. 204

LIST OF TABLES

TABLE

2.1 DQI dimensions and scoring criteria 10
2.2 Model Comparison with Optimal In-Context Learning Performance for DQI Task 26

3.1 Speech-Level Annotations for the Debate on Economic Mismanagement 51
3.2 Descriptive Statistics of Parliamentary Debates by Parliament and Session . . . 56
3.3 Cohesion and Polarization Scores Across Selected Parliaments 69

4.1 Classification Performance: Comparison with Gohdes (2020) results 92

B.1 Discourse Quality Index (DQI) Criteria 154

C.1 **Confusion Matrix: Comparison of the Predictions of a Classifier to Documents' True Labels** 175

LIST OF APPENDICES

A Appendix for "Towards Qualitative Measurement at Scale: A Prompt-Engineering Framework for Large-Scale Analysis of Deliberative Quality in Parliamentary Debates"

A.1 Large Language Models for Text Classification 104
A.2 The Discourse Quality Index: Concept, Measurement, and Application . . . 107

B Appendix for "LegisGraph: A Graph-Based Method for Analyzing Legislative Debates Using Language Models"

B.1 Graph Generation Prompt Overview 111
B.2 DQI+ Annotation Prompt Overview 153
B.3 Extracting Information from the Neo4j database 165

C Appendix for "Improving Probabilistic Models in Text Classification via Active Learning"

C.1 Using Machine Learning for Text Classification 171
C.2 Supplemental Information for Validation Performance 177
C.3 Binary Classification: EM Algorithm 179
C.4 Multiclass Classification 183
C.5 Binary Classification With Multiple Classes 187
C.6 A LURE Approach to Remove The Bias of Active Learning 190
C.7 Classification Performance with Mislabels 199

ABSTRACT

This dissertation advances the frontier of computational political science by developing novel AI-driven methodologies for analyzing large-scale political discourse. It addresses three interconnected challenges in legislative and deliberative democracy research: 1) scaling qualitative measurements, 2) mapping complex argumentative structures, and 3) enhancing text classification efficiency.

The first study introduces a prompt-engineering framework leveraging large language models (LLMs) to automate the coding of deliberative quality in parliamentary speeches. I show that through a combination of detailed code book-style annotation instructions and examples drawn from a pre-validated collection of speeches, LLMs can achieve human-level performance when applying the Discourse Quality Index (DQI) to legislative debates from the US Congress.

Building on this, the second study presents LegisGraph, a new approach combining LLMs with network science to represent legislative debates as structured argument graphs, where nodes represents speeches, speakers, arguments and topics, and edges capture relationships between them. Applying it to a representative corpus of Canadian parliamentary debates, I show how this method can be scaled to analyze large corpora of parliamentary debates, enabling analysis of a wide range of dynamics, including topic distribution, discourse quality trends, and patterns of polarization.

The third study focuses on improving text classification efficiency by developing an algorithm that combines probabilistic modeling with active learning. By leveraging both labeled and unlabeled data, and focusing labeling efforts on challenging documents, this approach significantly reduces the need for labeled data while maintaining high classification accuracy. I demonstrate the effectiveness of this method through replication of two published studies with only a fraction of the original labeled data.

Collectively, these studies demonstrate the transformative potential of AI in political communication research, offering scholars powerful tools to analyze vast corpora of political text with unprecedented depth and efficiency. This work lays the groundwork for new research that can shed light on the complexities of legislative and deliberative processes, informing policy-making and democratic governance.

CHAPTER 1

Introduction

This dissertation demonstrates the transformative potential of artificial intelligence (AI), particularly large language models (LLMs), in advancing the study of political discourse. LLMs, a subset of generative AI, are designed to generate new content by drawing on vast amounts of textual data. These models are trained on extensive corpora, enabling them to predict and produce human-like text based on a given input. Unlike earlier natural language processing models that relied on fixed rules or limited datasets, LLMs such as GPT-4 and Claude can understand and generate coherent responses across a wide range of topics. This capability is a hallmark of generative AI, which refers to the broader class of models that can create novel outputs—whether text, images, or other forms of data—by identifying patterns in the data they have been trained on. The versatility of these models allows them to take on a variety of complex tasks, from summarization and translation to intricate content analysis, without the need for task-specific retraining. Through the three papers in this dissertation, I explore novel methodologies that harness the power of LLMs to push the boundaries of computational political science. These methodologies—prompt engineering for qualitative measurement, graph-based analysis of debate structures, and active learning for text classification—address long-standing challenges in the field, offering scalable solutions for the analysis of complex language phenomena that were previously beyond reach.

One of the central contributions of this work is the introduction and systematic development of prompt engineering as a methodological tool in political science. Prompt engineering, an emerging practice, is used to guide the outputs of generative AI models through carefully crafted, structured natural language prompts. While this practice has gained traction in various fields, my dissertation goes further by articulating a structured approach specifically for applied political science. In essence, a prompt can be understood as a program written in English, designed to guide an LLM in generating outputs that meet the analytical requirements of the task. Unlike less sophisticated prompt engineering approaches, which rely on simple instructions to nudge model behavior, my framework integrates detailed coding instructions,

contextual information, and illustrative examples to mimic the rigor of human coding practices. This approach also stands apart from traditional machine learning methods, where a model is trained from scratch or fine-tuned on task-specific data. Instead of retraining or fine-tuning, the LLM operates based on structured prompts, allowing it to perform complex analyses without the need for expensive and time-consuming model training. Further, by employing agentic workflows, I demonstrate how multiple AI agents—each equipped with a unique, specialized prompt—can be chained together to perform a series of connected tasks. This allows for richer and more detailed representations than any single model pass could achieve, leveraging the strengths of each agent to transform data progressively. Additionally, a key part of this contribution is the framework I develop for iteratively refining prompts at each stage of the workflow. By testing, evaluating, and adjusting the prompts based on outputs, I ensure that each agent in the chain is optimized for its task, thus improving the overall quality of the final output.

The first paper in this dissertation develops a novel framework for prompt engineering, automating the coding of deliberative quality in legislative debates. Drawing inspiration from the coding procedures used by human annotators, I design prompts that mirror the detailed instructions and context human coders typically rely on. The prompt for Discourse Quality Index (DQI) annotation includes theoretical context, detailed annotation criteria, illustrative examples, and the speech text itself, ensuring that the model receives the same kind of comprehensive information that a human would have. Empirically, this approach proves remarkably effective. The LLM-generated DQI scores closely approximate those produced by human coders, replicating their judgments across key dimensions such as participation, justification, and respect for counterarguments. These results, drawn from an analysis of U.S. Congressional debates, demonstrate that LLMs, when guided by well-designed prompts, can perform complex qualitative tasks with both accuracy and scalability. The implications are significant, allowing for the analysis of deliberative quality across large datasets—an endeavor previously constrained by the time and resource demands of manual coding.

Building on this foundation, the second paper generalizes the prompt-engineering framework to develop a more advanced system for structuring and analyzing debates: the LegisGraph framework. This framework transforms raw parliamentary text into graph-based representations of debates, capturing the relationships between speakers, arguments, and topics. Central to this process is the use of agentic workflows, where a series of AI agents are each assigned a unique, task-specific prompt that guides them through specialized data transformations. For example, the process begins with an AI agent transforming raw text into a structured summary. A second agent, equipped with a different prompt, then converts this summary into a graph representation. Finally, additional agents enrich the graph by

incorporating further details about the debate, such as contextual relationships or argument nuances. This layered, multi-agent approach enables the generation of increasingly detailed and structured representations of political discourse, far beyond what could be achieved by a single model in one pass. The contribution is not just in chaining these agents together, but in developing an iterative prompt refinement framework that ensures each agent's prompt is continuously improved based on output evaluation. The empirical application of LegisGraph to Canadian parliamentary debates reveals clusters of agreement and contradiction between parties, as well as patterns in how issue salience correlates with argument sophistication. These findings underscore the potential of graph-based analysis to uncover the relational dynamics of parliamentary debates, offering a new perspective on political interactions that traditional methods have struggled to capture.

This approach—combining the competence of highly sophisticated pre-trained AI models with detailed, task-specific prompts—represents a brand-new paradigm for tackling research tasks involving text data. Rather than relying on either manually intensive annotation or traditional machine learning methods that require task-specific training or fine-tuning, this framework allows researchers to condition LLMs to perform a wide range of tasks with minimal retraining. Beyond the annotation tasks discussed in this dissertation, this paradigm holds potential for diverse applications, including persuasion tasks, where LLMs can be prompted to structure the outputs of "chatbot" agents, or even simulation tasks, where AI models are conditioned to behave like participants in political systems. The flexibility and adaptability of this new approach open the door to a broader range of inquiries into how AI can assist in understanding and modeling complex social and political interactions.

The third paper shifts focus to the efficiency of text classification, presenting activeText, an algorithm that combines probabilistic modeling with active learning to minimize the need for labeled data in political text classification tasks. ActiveText directs the model to focus on the most informative documents, thus reducing the amount of manual labeling required while maintaining high classification accuracy. This approach is particularly valuable for political scientists who are working with large-scale datasets but have limited resources for extensive labeling. Empirical tests demonstrate that activeText can achieve high performance using only a fraction of the labeled data typically required, making it a practical solution for resource-constrained research environments. The algorithm's successful application to real-world political discourse datasets shows that efficient and accurate classification can be achieved even in data-scarce contexts, opening up new opportunities for large-scale political text analysis.

Taken together, these three papers demonstrate the potential of AI-driven methods to reshape the study of political communication. The first paper shows how prompt engineering

can be leveraged to scale qualitative judgments, providing a framework that enables LLMs to approximate human-level annotations across vast amounts of text. The second paper expands on this by introducing LegisGraph, a tool that visualizes and analyzes the structure of parliamentary debates, revealing the complex relationships between speakers and arguments in ways that traditional methods could not. The third paper offers a solution to the practical challenge of limited labeled data, demonstrating how activeText can ensure efficient text classification in large-scale political datasets. Each of these contributions is grounded in empirical results, which illustrate the real-world applicability of the methods and underscore their transformative potential.

By offering innovative, scalable methods for analyzing deliberative quality, polarization, and argumentative structures, this dissertation pushes the boundaries of computational political science. The methods developed here open new avenues for comparative research across different political systems and time periods, providing deeper insights into the dynamics of political communication in democratic systems. By bridging the gap between qualitative depth and quantitative scale, this work lays the foundation for a new era of political discourse analysis, where AI tools enable researchers to examine complex phenomena with an unprecedented level of both scope and precision.

CHAPTER 2

Towards Qualitative Measurement at Scale: A Prompt-Engineering Framework for Large-Scale Analysis of Deliberative Quality in Parliamentary Debates

2.1 Introduction

The ability to measure complex linguistic, psychological, and social constructs from textual data is central to many research questions in the social sciences and humanities. In political science, for example, understanding the quality of deliberative communication in parliamentary debates is essential for evaluating the health of democratic institutions and the legitimacy of political decisions (Habermas, 1996; Bächtiger et al., 2018b; Steenbergen et al., 2003). Similarly, tracking the evolution of social norms and cultural beliefs over time involves measuring latent dimensions like moral sentiment (Garg et al., 2018), gender stereotypes (Garg et al., 2018), and individualism-collectivism (Grossmann and Varnum, 2015) in large historical corpora.

Traditionally, measuring such rich, high-dimensional constructs from text has relied on qualitative methods like expert human annotation. Approaches like the Discourse Quality Index (DQI; Steenbergen et al. 2003) leverage the knowledge and interpretive abilities of trained coders to make judgments about multiple dimensions of textual data, from the logical coherence of arguments to the respect afforded to alternative viewpoints. While these methods can yield valid and reliable measurements of complex constructs, they are often prohibitively expensive and time-consuming to apply to large datasets. As a result, many researchers have turned to more scalable computational methods, such as dictionary-based sentiment analysis (Tausczik and Pennebaker, 2010), supervised machine learning on labeled data (Rudkowsky et al., 2018), and unsupervised topic modeling (Blei et al., 2003). However,

these automated methods typically measure relatively narrow and simplistic constructs, as they rely on predefined linguistic features or limited training data and cannot reason about the deeper meaning and context of the text.

In this paper, I propose a novel approach that uses large language models (LLMs) and prompt engineering to overcome the tradeoff between measurement richness and scalability in textual analysis. LLMs, such as GPT-4 (OpenAI, 2023), Claude (Anthropic, 2023), DeepSeek (Zhu et al., 2024), and Meta's LLaMA (Dubey et al., 2024), are deep neural networks that have been pretrained on vast amounts of textual data to perform open-ended natural language tasks. By virtue of their pretraining, LLMs have absorbed a broad knowledge base spanning science, history, culture, and current events (Petroni et al., 2019), and have developed strong capabilities for natural language understanding and generation (Brown et al., 2020). Crucially, LLMs can be "programmed" to perform specific tasks through prompt engineering—the process of designing natural language instructions that guide the model to produce the desired output (Liu et al., 2021).

My key argument is that prompt engineering can be used to imbue LLMs with the knowledge and analytical abilities of human experts, enabling them to make rich, contextually informed judgments about the linguistic, psychological, and social dimensions of text. By providing LLMs with detailed definitions and examples of the target constructs, along with step-by-step instructions for applying expert reasoning to the data, I show that we can generate high-quality annotations of complex latent variables directly from the raw text. This approach leverages the pretrained knowledge and generative capabilities of LLMs to scale up the process of expert content analysis, making it feasible to measure constructs across large, heterogeneous corpora.

In this paper, I empirically assess the ability of LLMs to automate the coding of the DQI in a sample of parliamentary speeches. To do this, I construct a validation dataset of 1000 human-annotated speeches from the 101st and 104th US Congress by manually combining the raw speeches from the Congressional Record (Gentzkow and Shapiro, 2017) with the corresponding DQI annotations from expert coders (Steenbergen et al., 2003). Using this dataset, I evaluate the performance of different LLMs in automating the coding of the DQI using a variety of prompting strategies, including zero-shot, few-shot, and many-shot in-context learning, as well as Chain-of-Thought reasoning where the model is guided to generate a logical argument chain for each annotation. I show that many-shot learning is particularly effective at reducing annotation error across most dimensions of the DQI, but that there are diminishing marginal returns on examples after a certain point, typicaly around 25-50 examples. I also find that while more expensive high-parameter models like GPT-4 perform better with few in-context examples, this advantage diminishes as the number of

examples increases, with less costly models like DeepSeek Coder outperforming GPT-4 when provided with sufficient examples.

In sum, this paper contributes to both established literature in political science that seeks to measure the quality of deliberative communication, and the broader field of computational social science by providing a novel framework for measuring complex constructs from text, and demonstrating the potential of LLMs to advance research on political communication, deliberative democracy, and social interaction. In addition, I provide a first step towards a general framework for using LLMs and prompt engineering to scale the measurement of complex constructs across diverse research domains, potentially transforming how we study and understand the dynamics of linguistic, psychological, and social interaction in the digital age. All of the code for my implementation is available as a Github repository[1], along with a web application that allows users to interact with the enriched network representation of the debates, and explore the relationships between speakers, arguments, and topics.

2.2 Previous Work

This paper builds on and integrates three research topics: computational methods for measuring latent constructs from text, large language models and their applications, and the Discourse Quality Index and its role in deliberative democracy research.

2.2.1 Computational Methods for Textual Measurement

A growing body of research in the social sciences and humanities uses computational methods to measure linguistic, psychological, and social constructs from large textual corpora. One common approach is dictionary-based text analysis, which counts the occurrence of predefined words and phrases associated with particular constructs (Grimmer and Stewart, 2013). For example, the Linguistic Inquiry and Word Count (LIWC) software uses dictionaries to measure dimensions like emotional tone, cognitive processes, and social relationships in text (Tausczik and Pennebaker, 2010). While dictionary methods are simple and transparent, they often lack precision and context-sensitivity, as they do not consider the deeper meaning or syntactic relations between words (Ribeiro et al., 2016).

More recently, researchers have applied supervised machine learning to textual measurement tasks, using labeled data to train classifiers to predict latent categories (Grimmer and Stewart, 2013). For instance, Rudkowsky et al. (2018) train a support vector machine on hand-coded examples to detect speeches related to economic policy in Austrian parliamentary

[1] https://github.com/mbosley/dqi-annotation-pipeline

records. Similarly, Card et al. (2015) use a logistic regression model to measure ideological bias in newspaper articles, based on human annotations. While supervised learning can achieve high accuracy in specific domains, it requires substantial upfront investment in data labeling and is difficult to generalize to new contexts or constructs.

Unsupervised learning techniques, such as topic modeling (Blei et al., 2003) and clustering using word embeddings (Mikolov et al., 2013), offer a more flexible approach to measuring latent dimensions in text. These methods uncover hidden semantic structures in the data, which can be used to track the evolution of themes (Chaney and Blei, 2012), ideologies (Sagi and Deignan, 2013), and cultural values (Garg et al., 2018) over time. However, unsupervised methods typically require careful post-hoc interpretation to map the learned structures onto meaningful theoretical constructs, and may not capture more complex, multi-dimensional concepts.

In contrast to these automated approaches, the method proposed here leverages the knowledge and reasoning capabilities of large language models to emulate the context-sensitive judgments of human experts in measuring latent constructs from text. By explicitly encoding the definitions, examples, and analytical logic of the target constructs in the prompt, I can guide the model to generate rich, theoretically grounded annotations at scale, without the need for extensive upfront labeling or post-hoc interpretation.

2.2.2 Large Language Models and their Applications

The approach is enabled by recent breakthroughs in natural language processing, particularly the development of large language models (LLMs) - deep neural networks that learn to model the statistical patterns in huge text corpora (Bommasani et al., 2021). LLMs like GPT-3 (Brown et al., 2020), PaLM (Chowdhery et al., 2022), and Claude (Anthropic, 2023) have achieved remarkable performance across a wide range of language tasks, including question-answering, summarization, and open-ended generation. By virtue of their broad pretraining, LLMs have acquired an extensive knowledge base spanning science, history, culture, and current events (Petroni et al., 2019), as well as strong capabilities for natural language understanding and reasoning (Rae et al., 2021).[2]

A key property of LLMs is their ability to perform new tasks without further training, simply by conditioning on natural language prompts that describe the desired behavior (Liu et al., 2021). This "in-context learning" capability has enabled researchers to adapt LLMs to a wide range of applications through prompt engineering - the process of designing instructions that elicit the desired model outputs (Reynolds and McDonell, 2021). For example, by

[2]For a more in-depth overview of the use of LLMs for measuring concepts from text, refer to Appendix A.1.

providing a prompt with a few example question-answer pairs, users can guide LLMs to answer open-ended queries about new topics (Brown et al., 2020). Similarly, by specifying the desired format and style in the prompt, users can control the model's generation of summaries, translations, and creative writing (Radford et al., 2019).

Prompt engineering is an emerging technique in natural language processing (NLP) and artificial intelligence (AI) that involves designing and optimizing textual prompts to elicit high-quality outputs from language models (Liu et al., 2021). By carefully structuring the prompt to provide context, instructions, examples, and formatting guidance, researchers can effectively "program" the model to perform complex tasks, such as data annotation and content analysis. Prompt engineering has been successfully applied in various domains, such as sentiment analysis (Shin et al., 2020), named entity recognition (Cui et al., 2021), and open-ended question answering (Kojima et al., 2022). Prompt engineering has been used to improve the performance of LLMs on benchmark NLP tasks like sentiment analysis (Shin et al., 2020), named entity recognition (Li et al., 2022), and textual entailment (Liu et al., 2022). Researchers have also developed techniques for optimizing prompts to achieve specific behaviors, such as reducing harmful outputs (Ouyang et al., 2022), improving truthfulness (Lin et al., 2021), and following instructions (Mishra et al., 2022). However, there has been little work on using prompt engineering to guide LLMs to perform open-ended content analysis of the kind traditionally done by human experts.

This paper contributes to this research by showing how prompt engineering can be used to generate context-sensitive judgments about the linguistic, psychological, and social dimensions of text from LLMs. By providing the model with definitions, examples, and step-by-step instructions for applying expert analytical frameworks, I can effectively "program" it to perform rich, theoretically grounded measurement of complex constructs at scale. This approach opens up new possibilities for leveraging the knowledge and reasoning capabilities of LLMs to accelerate research in the social sciences and humanities.

2.2.3 The Discourse Quality Index

The Discourse Quality Index (DQI) is a widely used framework for assessing the deliberative quality of political communication, which has played a central role in empirical research on deliberative democracy (Steenbergen et al., 2003; Bächtiger et al., 2018b). Deliberative democracy is a normative theory that emphasizes the importance of reasoned, inclusive, and respectful dialogue in political decision-making (Habermas, 1996; Gutmann and Thompson, 2004). It argues that legitimate and effective governance requires that citizens and their representatives engage in open-minded and mutually accountable communication, with the

Dimension	Definition	Scoring Criteria
Participation	The extent to which participants are able to engage in the debate without being constrained.	0: Participation impaired, 1: Normal participation
Justification (Level)	The extent to which arguments are supported by reasons.	0: No justification, 1: Vague assertion, 2: General justification, 3: Specific justification
Justification (Content)	The quality of the justification provided.	0: Lack of facts, 1: Oversimplification, 2: Relevant examples, 3: Balanced consideration
Respect (Groups)	The extent to which participants show respect towards other groups.	0: No respect, 1: Formal respect, 2: Substantial respect
Respect (Demands)	The extent to which participants show respect towards the demands of other groups.	0: No respect, 1: Formal respect, 2: Substantial respect, 3: Acknowledges legitimacy
Respect (Counterarguments)	The extent to which participants show respect towards counterarguments.	0: No respect, 1: Formal respect, 2: Substantial respect, 3: Acknowledges validity, 4: Engages with counterargument
Constructive Politics	The extent to which participants engage in positive sum politics and try to reach compromise solutions.	0: Positional politics, 1: Alternative proposal, 2: Consensus appeal, 3: Mediating proposal

Table 2.1: DQI dimensions and scoring criteria

aim of finding common ground and reaching justifiable decisions.

The DQI aims to operationalize these normative criteria into a set of measurable dimensions that can be used to evaluate the deliberative quality of real-world political discourse (Steenbergen et al., 2003). It consists of a detailed coding manual that guides trained annotators to score speech acts on seven key dimensions: participation, level of justification, content of justification, respect for groups, respect for demands, respect for counterarguments, and constructive politics. Each dimension is scored on an ordinal scale based on the degree to which the speech act fulfills the relevant deliberative criteria, with higher scores indicating more deliberative quality.

The DQI has been applied to a wide range of political contexts, including parliamentary debates (Bächtiger and Steiner, 2005), public consultation processes (Caluwaerts and

Deschouwer, 2014), and online discussions (Friess and Eilders, 2015). These studies have yielded important insights into the factors that shape deliberative quality, such as institutional design (Steiner et al., 2004), issue characteristics (Hangartner et al., 2007), and group diversity (Caluwaerts and Reuchamps, 2014). They have also explored the consequences of deliberative quality for outcomes like decision legitimacy (Steenbergen et al., 2003), opinion change (Luskin et al., 2002), and policy innovation (Bächtiger et al., 2010).

However, the application of the DQI to large-scale textual data has been limited by the time and effort required for manual annotation. Coding a single debate using the DQI can take several weeks and requires multiple trained annotators to ensure reliability (Steiner et al., 2004). This has constrained the scope and generalizability of previous DQI studies, which have typically focused on a small number of purposively selected cases. To date, there has been little work on automating the DQI coding process to enable large-scale deliberation research.

This paper addresses this gap by demonstrating how large language models can be used to generate high-quality DQI annotations at scale, through the use of prompt engineering. By providing the model with the coding criteria, examples, and analytical steps in the DQI manual, I can guide it to perform the same kind of context-sensitive evaluation of deliberative quality as human experts. This opens up new opportunities for systematic comparative research on the dynamics and impacts of deliberative communication across a wide range of contexts and over time. It also contributes to the broader enterprise of measuring the health and quality of democratic discourse in an era of polarization, misinformation, and digital transformation.

2.3 Methodology

In this section, I detail the methodology I use to generate my results, with which I am to evaluate the performance of large language models in automating the coding of the Discourse Quality Index (DQI) in a sample of parliamentary speeches. I show that by carefully designing prompts that provide LLMs with the knowledge and reasoning abilities of human experts, we can elicit high-quality annotations of complex constructs directly from textual data, at a fraction of the cost and time required for manual coding, and that many-shot in-context learning is particularly effective at reducing annotation error across most dimensions of the DQI.

2.3.1 Data

To establish a baseline measurement of the quality of the DQI annotations generated by LLMs, I construct a validation dataset of 1000 parliamentary speeches from the 101st and 104th US Congress from Steenbergen et al. (2003). To do so, I manually combined speech-level DQI annotations by expert coders[3] with the raw speeches from the Congressional Record (Gentzkow and Shapiro, 2017). The data spans several dozen distinct debates on a wide range of topics, including healthcare, abortion, and guns rights. Each debate is comprised of multiple speeches by different members of Congress, with each speech annotated on seven dimensions of the DQI: participation, level of justification, content of justification, respect for groups, respect for demands, respect for counterarguments, and constructive politics. As described in Table B.1, each dimension is scored on an ordinal scale with higher scores indicating higher deliberative quality.[4]

2.3.2 Prompt Structure

I develop a prompt template to guide the LLM annotation of parliamentary speeches using the DQI framework, the guiding philosophy of which is to provide the language model with all of the information that an expert human annotator would be provided with to conduct the same annotation task. Effectively, I aim to provide the model with a *codebook*, a set of instructions and guidelines for how to apply the DQI to a given speech, and the context that the speech occurs in.

An expert annotator tasked with applying the DQI to a series of legislative speeches would be provided with, for example, an overview of the measurement technique and a theoretical justification for its elements; strict guidelines for applying the measure to a given speech; instructions for how to structure their annotations; and the context that the speech that they are annotating occurs (either explicitly, or incidentally as a result of sequential labeling).

With the goal of replicating this same degree of detail for the AI model conducting the annotation, my prompt template consists of the following eight components represented in Figure 2.1, where each of the curly-braced components is replaced with the relevant information for the specific annotation task.[5]

[3]The DQI annotations were obtained using the Wayback Machine, a tool that automatically archives web pages for later retrieval, to access the original website which housed the DQI annotations, which is no longer available. The webpage can be found here.

[4]For an in-depth overview of the DQI and its annotation criteria, refer to Appendix A.2.

[5]A complete example of a prompt provided a language model can be found in accompanying `dqi-prompt.txt`, and is omitted from the paper due to its length.

```
You are an expert political scientist. Analyze the given
   parliamentary speech and annotate it on the following
   dimensions of the Discourse Quality Index (DQI), providing
   your reasoning, scores, and confidence levels in the specified
    format.

Here is a theoretical overview of the DQI:
{dqi-theory}

Here is the set of annotation criteria for each dimension:
{dqi-criteria}

Structure your annotations in the following format:
{output-format}

Here are examples of speeches with corresponding annotations:
{example_annotation-1}
{example_annotation-2}
...
{example_annotation-n}

Here are the preceding speeches in the debate:
{preceding-speeches}

Annotate the following speech:
{target-speech}

When deciding on scores, engage in the following reasoning
   process:
{reasoning-process}.
```

Figure 2.1: High level overview of the DQI annotation prompt.

High-Level Overview The prompt explains the expert role of the annotator (a political scientist), and briefly describing the task (analyzing a parliamentary speech using the DQI framework).

Theoretical Foundation As specified in `dqi-theory`, the prompt provides an overview of the DQI framework and its theoretical foundations in the context of deliberative democracy. See 2.2 for an excerpt of the theoretical foundation provided to the model. This formulation was taken directly from the appendix in Steenbergen et al. (2003) that provides the theoretical foundation of the DQI.

Dimension Definitions and Scoring Criteria The prompt details the dimensions and scoring criteria of the DQI, as outlined in `dqi-criteria`, providing descriptions and examples of the scoring levels for each dimension and guiding the model to consider evidence for each possible score before making a final decision. See 2.3 for an excerpt of the dimension definitions and scoring criteria provided to the model. As with the theoretical foundation, this information was taken directly from the appendix in Steenbergen et al. (2003) that provides the dimension definitions and scoring criteria of the DQI.

Output Format As described in `output-format`, the prompt establishes a structured output format for the annotations, including fields for each dimension and sub-dimension, as well as additional fields for overall notes and summary if specified. See Figure 2.4 for the JSON[6] output format provided to the model.

Sample Speeches and Annotations The prompt includes a set of sample speeches with corresponding annotations, in `example-annotation-1` to `example-annotation-n`, with the goal of both establishing the expected level of detail and reasoning in the evidence/score justifications and providing a model for the annotator to follow when coding the target speech. To generate these examples, I select a random subset of speeches from the validation dataset and for each speech provide the complete speech text along with the corresponding DQI annotations. As I discuss in the next section, I experiment with different numbers of randomly selected examples to determine how responsive the model is to demonstration of the desired annotation behavior.

Preceding Speeches The prompt provides the preceding speeches in the parliamentary debate, in `preceding-speeches`, to give the model necessary context for understanding

[6]JSON (JavaScript Object Notation) is a lightweight data-interchange format that is easy for humans to read and write and easy for machines to parse and generate.

and evaluating the speech to be annotated, including references to earlier speakers and arguments, and assessing the speech's responsiveness to other participants. When possible, I provide four speeches that sequentially precede the target speech in the debate, to give the model a sense of the ongoing conversation and the issues at stake. So that the previous speeches do not overwhelm the annotation task, I provide an excerpt of the first 500 characters of each previous speech, up to a maximum of 2000 characters for the four speeches.

Target Speech The prompt presents the target speech to be annotated, in `target-speech`. The target speech is the main focus of the annotation task, and the model is directed to provide detailed annotations for each dimension of the DQI based on the content of the speech. I provide the first 3000 characters of the speech to ensure that the model has sufficient context to make informed judgments.

Reasoning Process Finally, the prompt reiterates key instructions for the annotation task, and directs the model to respond with a particular reasoning process when deciding on scores, in `reasoning-process`. Figure 2.5 shows the baseline closing instructions provided to the model without any explicit attempt to guide the reasoning process.

It is worth noting that this prompt template was developed iteratively over several rounds of piloting and refinement in which I tested the effectiveness of different prompt components and formats in guiding the model to generate high-quality DQI annotations.

2.3.3 Model Selection and Parameter Configuration

Choosing the models There are four main considerations in selecting the models for the study: the size/power of the model, its cost, the length of the input context, and whether it is closed or open source. In general, larger models are expected to perform better on complex tasks like the DQI, but they are also more expensive to run and may require more data to fine-tune effectively. The length of the input context is important for capturing the full speech and debate context, while the cost of the model is a key factor in determining the feasibility of large-scale annotation tasks. Finally, the availability of the model's source code can be important for transparency, reproducibility, and customization—and so if all-else is equal, open-source models are preferred.

I evaluate the performance of several large language models in automating the coding of the DQI, including GPT-4o from OpenAI, Claude 3 Haiku from Anthropic, DeepSeek Coder 2 from DeepSeek, Llama 3 70B from Meta, Llama 3 8B from Meta, Qwen2 72B from Alibaba, and WizardLM 2 from Microsoft. These models, each of which are at or near the state-of-the-art, vary in size, cost, and training data, and training techniques, with parameters ranging

```
The DQI attempts to measure political deliberation in a general,
   valid and reliable way (Steenbergen et al. 2003). It mainly
   draws on Habermasian discourse ethics, but also incorporates
   elements of other deliberative models. The unit of analysis of
    the DQI is a speech act, i.e. the discourse by a particular
   individual delivered at a particular point in a debate. For
   each speech, we distinguish between relevant and irrelevant
   parts, and only the relevant parts are coded. A relevant part
   is one that contains a demand, i.e. a proposal on what
   decision should or should not be made. Our emphasis on demands
    stems from the fact that they constitute the heart of the
   deliberation. That is, demands stipulate what ought to be and
   what ought not to be, and this normative character puts them
   at the center of discourse ethics. The DQI is composed of
   seven indicators. Despite the considerable complexity of
   parliamentary debates, we attempt to keep the coding
   categories relatively easy, so as to ensure a high level of
   reliability. The following is an elaboration of these
   indicators, followed by an overview of the indicators and
   their codes. We discuss seven the seven indicators under four
   headings.

1. Participation. Participation constitutes a fundamental
   precondition for deliberation. In parliamentary settings of
   western democracies, this type of basic participation can
   usually be seen as given for the elected representatives.
   Normal participation is only assumed to be impaired if a
   speaker is cut off by a formal decision, or if she or he feels
    explicitly disturbed in the case of a verbal interruption by
   other actors.

...
```

Figure 2.2: Excerpt of DQI Theoretical Foundation Provided To The Model

```
1. Participation
0: participation was impaired - speaker was cut off or explicitly
    disturbed
1: normal participation was possible

2. Justification
2.1 Level of Justification
0: no justification
1: inferior justification - reason given but not properly linked
   to demand
2: qualified justification - one complete linkage between demand
   and reason
3: sophisticated justification (broad) - multiple complete
   justifications
4: sophisticated justification (in depth) - multiple
   justifications with one embedded in complete inference chain

2.2 Content of Justification
0: explicit reference to group/constituency interests
1: neutral statement, no explicit references to group interests
2: explicit reference to common good (utilitarian/collective)
3: explicit reference to helping least advantaged (difference
   principle)

...
```

Figure 2.3: Excerpt of DQI Dimension Definitions and Scoring Criteria Provided To The Model

```json
{
  "participation": {
    "reasoning": "Brief explanation for the score",
    "score": [0/1]
  },
  "justification": {
    "level": {
      "reasoning": "Brief explanation for the score",
      "score": [0/1/2/3/4]
    },
    "content": {
      "reasoning": "Brief explanation for the score",
      "score": [0/1/2/3]
    }
  },
  "respect": {
    "groups": {
      "reasoning": "Brief explanation for the score",
      "score": [0/1/2]
    },
    "demand": {
      "description": "Brief description of main demand",
      "reasoning": "Brief explanation for the score",
      "score": [0/1/2/3]
    },
    "counterargument": {
      "description": "Brief description of main counterargument",
      "reasoning": "Brief explanation for the score",
      "score": [0/1/2/3/4]
    }
  },
  "constructive_politics": {
    "reasoning": "Brief explanation for the score",
    "score": [0/1/2/3]
  }
}
```

Figure 2.4: Output Format for DQI Annotations Provided To The Model

```
Please provide your annotation in JSON format according to the
   schema provided in the [JSON SCHEMA] section above.
Format your JSON output with markdown-style backticks, like this:
"```json
{Your JSON here}```
Be sure to include ALL of the relevant information from the
   schema provided. Make sure that in your reasoning responses
   you consider step by step each of the possible annotation
   categories given the evidence at hand. Respond with only the
   JSON output.
```

Figure 2.5: Closing Instructions for Speech Annotation Prompt

from 8 billion to 236 billion, and token costs ranging from $0.05 to $5.00 per million tokens for input, and $0.25 to $15.00 per million tokens for output.[7]

These differences in price can be substantial: for example, a typical query with 20,000 input tokens (or roughly 7000 words) and 1000 output tokens (or roughly 300 words) would cost roughly $0.0125 with a model like DeepSeek Coder 2, but $1.15 with a model like GPT-4o, a difference of two orders of magnitude.

While each of these models has been trained on a large corpus of text data, they vary in the specific training data and techniques used, with some models being trained on a diverse range of text sources, and others being trained on more specialized or curated datasets. However, in the current landscape of large language models, labs and companies tend to keep the specifics of their training data and techniques proprietary, so it is difficult to make direct comparisons between models on this basis.

In presenting the results, I focus on the performance of the GPT-4o, Claude 3 Haiku, and DeepSeek Coder 2, as these models represent a range of sizes, costs, and training data, and are among the most widely used and well-known models in the field.

Parameter configuration In configuring the models for the DQI annotation task, I set the maximum token length of the input context to 128k tokens where possible, and the maximum token length of the output annotations to 4000 tokens. I also set the temperature

[7]Tokens are the basic units of text that the model processes, and the cost per token is a measure of how expensive it is to run the model. Each word is typically represented by three tokens, so the cost per token is a rough measure of the cost per word (divided by roughly 3). Input tokens refer to the tokens in the prompt that are used to condition the model, while output tokens refer to the tokens in the generated annotations that are used to evaluate the model's performance. Typically, API providers charge different rates for input and output tokens, with output tokens being more expensive than input tokens, due to the additional processing required to generate the model's response.

parameter to 0.7 for all models, which controls the randomness of the model's output, with lower temperatures leading to more deterministic responses. While there is as yet no consensus on the optimal temperature for fine-tuning LLMs, a temperature of 0.7 is a common choice in the literature, and has been found to produce high-quality outputs in a range of tasks. For all other hyperparameters, (e.g. top-k, top-p, nucleus sampling, etc.), I use the default settings provided by the model's API.

Prompt Variants As much as possible, I keep the prompt template consistent across models, to ensure a fair comparison of their performance on the DQI task. As I describe above, I principally vary the number of example speeches provided to the model, to see how responsive the model is to demonstration of the desired annotation behavior.

Because this is essentially the logic of supervised learning, I follow standard practices from that tradition, and split the set of validated speeches into a training set and a validation set, with 80% of the speeches used for training and 20% used for validation. I then randomly select 5, 10, 25, 50, and 100 example speeches from the training dataset, and provide these examples to the model in the prompt template, in addition to the theoretical overview, scoring criteria, and context information. For the Deepseek and Claude models, I conducted 5 random draws of the example speeches at each level, to ensure that the results are robust to the specific examples provided. I also construct a "zero-shot" variant of the prompt, where no examples are provided to the model, to see how well the model can perform the DQI annotation task with only the theoretical and scoring information provided in the prompt. I also experiment with a variant of DeepSeek Coder 2 that uses Chain-of-Thought reasoning, where the model is guided to generate a logical argument chain for each annotation, to see if this improves performance on the DQI task.

Evaluation Metrics To evaluate the performance of the models on the DQI annotation task, I use the Mean Absolute Error (MAE) of the model's annotations compared to the expert annotations in the validation dataset (which, as mentioned above, is a random set of 20% of the human annotated speeches that is not used to draw examples from for the training process). The MAE is calculated as the average absolute difference between the model's scores and the expert scores for each dimension of the DQI, across all speeches in the validation dataset.[8]

[8]Formally, the MAE is defined as:

$$\text{MAE} = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{M}\sum_{j=1}^{M}|y_{ij} - \hat{y}_{ij}| \tag{2.1}$$

While I also consider other metrics like accuracy and F1 score, the MAE is a natural measure of model performance because it naturally captures the ordinal nature of the DQI dimensions (i.e. a difference of 1 between scores 0 and 1 is more significant than a difference of 1 between scores 3 and 4), but without blowing up the error for large difference in scores the way that squared error would, and provides a single summary statistic for comparing the performance of different models.

To get a sense of how a given MAE score compares to how well a human annotator would perform on the DQI task, I take the reported inter-rater reliability of the DQI annotations in the literature as a benchmark. There, the inter-rater reliability of the DQI is typically reported as a Krippendorff's alpha of around 0.7 to 0.8 (Neblo et al., 2018), which is considered a good level of agreement for ordinal scales like the DQI.

We can translate this into a rough benchmark for the MAE by noting that the MAE is equivalent to the average absolute difference between two annotators' scores. For a single dimension of the DQI that is bounded between 0 and 4, we can say that the maximum possible difference between two annotators' scores is 4, and that the minimum possible difference is 0. Given this, we can say that an error of 2 for that dimension would be roughly equivalent to an inter-rater reliability of 0.5, which is a poor level of agreement for the DQI, and that an error of 1 would be roughly equivalent to an inter-rater reliability of 0.75, which the literature suggests is a good level of agreement for the DQI.

Therefore, if after aggregating the MAE scores across all dimensions of the DQI, we find that the MAE is roughly 1, we can say that the model is performing at a level that is roughly equivalent to a good human annotator on the DQI task. If the MAE is less than 1, we can say that the model is performing better than a good human annotator, and if the MAE is greater than 1, we can say that the model is performing worse than a good human annotator.

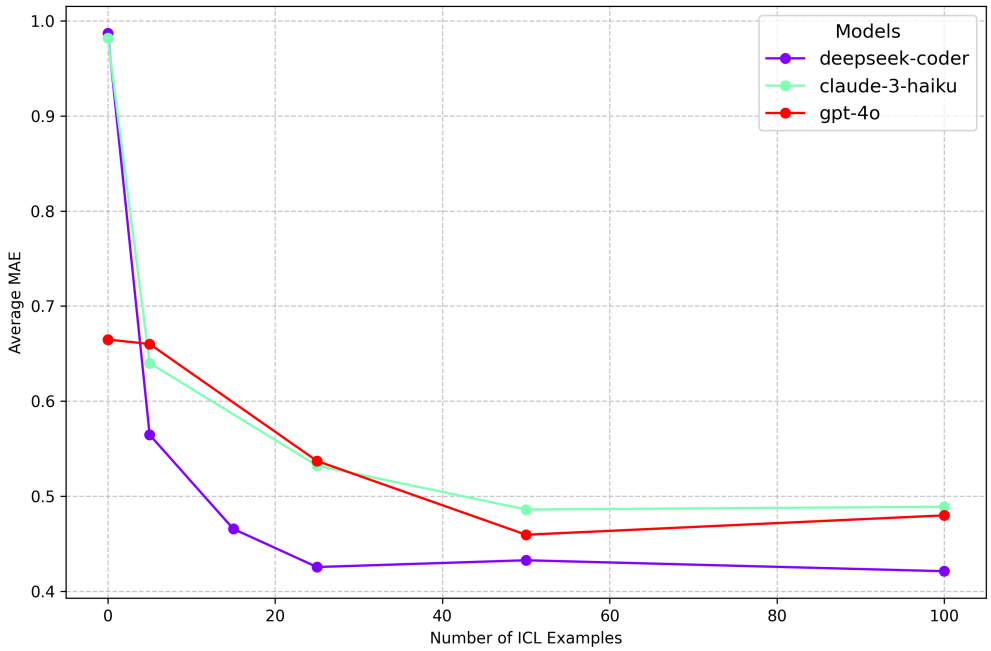
2.4 Results

2.4.1 Model Performance

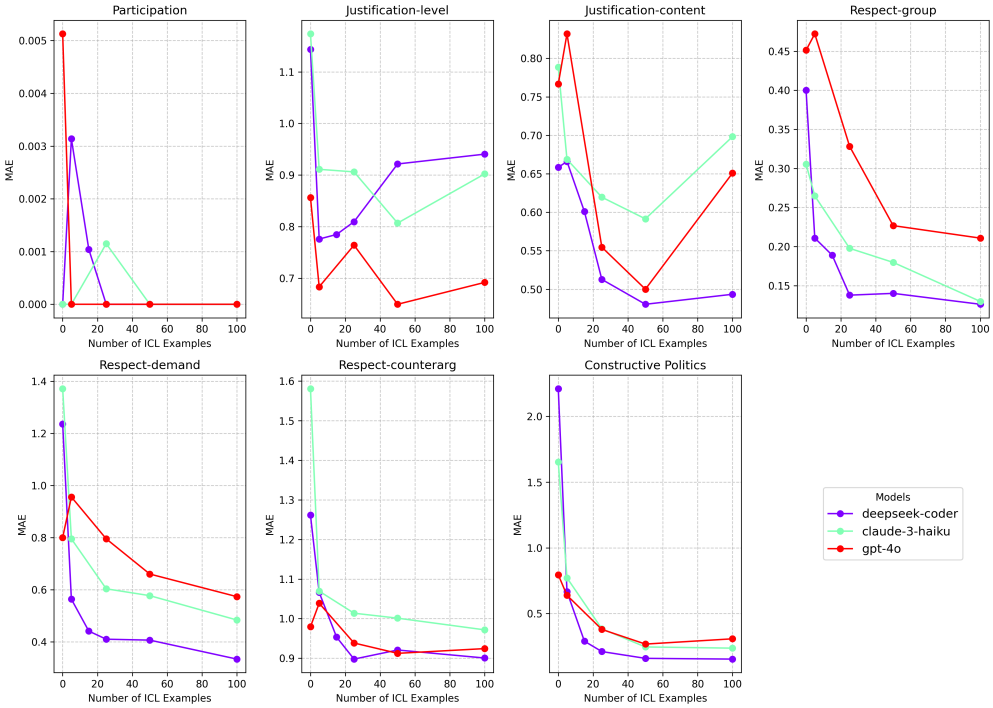
Figure 2.6 shows the performance of the models on the DQI annotation task as a function of the number of in-context learning (ICL) examples provided to the model (both in the aggregate 2.6a and by individual dimension 2.6b).

In both of these figures, the x-axis represents the number of in-context learning examples provided to the model, and the y-axis represents the mean absolute error of the model's

where N is the number of speeches in the validation dataset, M is the number of dimensions of the DQI, y_{ij} is the expert score for the j-th dimension of speech i, and $\hat{y}_{ij}$ is the model's predicted score for the j-th dimension of speech i.



(a) Model Performance (Mean Absolute Error) vs Number of ICL Examples



(b) Model Performance (Mean Absolute Error) vs Number of ICL Examples by DQI Dimension

Figure 2.6: Comparison of Model Performance for Different Numbers of In-Context Learning (ICL) Examples

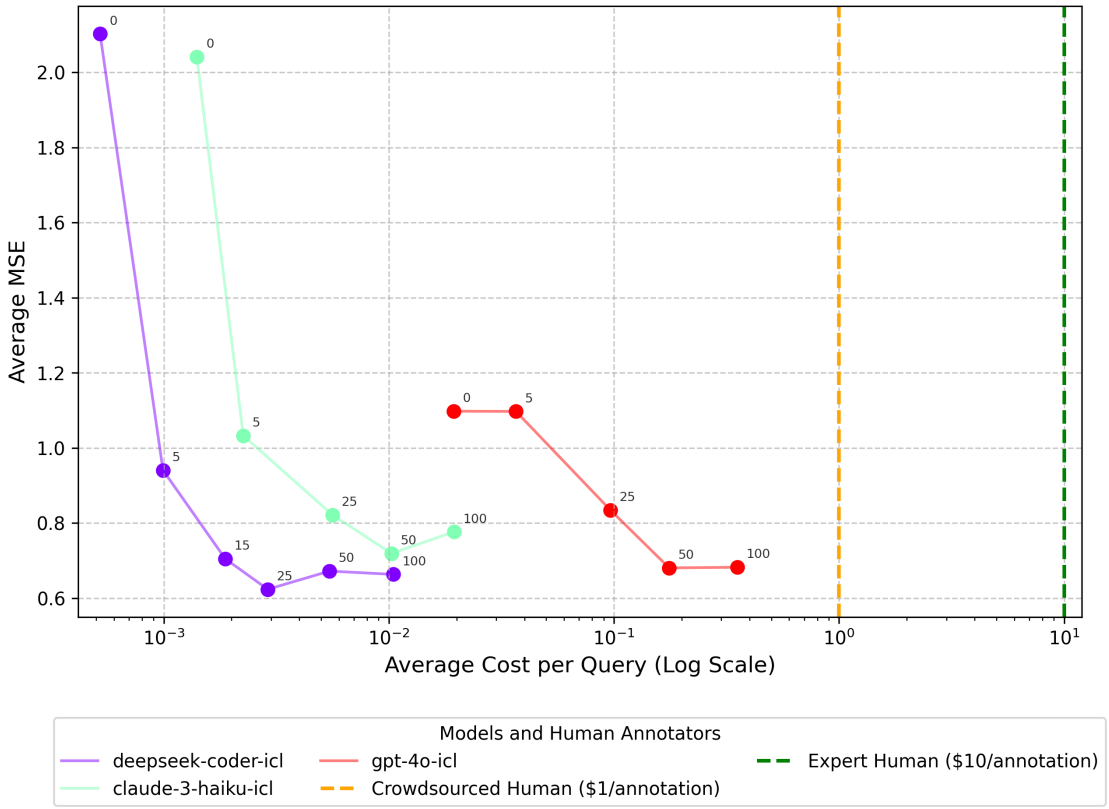


Figure 2.7: Model Performance (Accuracy) vs Model Cost

annotations compared to the expert annotations in the validation dataset. The colored lines represent the performance of different models on the DQI task (deepseek-coder, claude-3-haiku, and GPT-4o), with each line corresponding to a different model, and each point on the line corresponds to the performance for a given model with a given number of ICL examples.

On the left-most side of the figure we see the performance of the models with zero-shot learning, i.e. when no examples are provided to the model. In Figure 2.6a we see that there is substantial variation in the performance of the models with zero-shot learning: GPT-4o, the most expensive model in the study, performs best with an aggregate MAE roughly 0.67, whereas deepseek-coder and claude-3-haiku perform the worst with an aggregate MAE of just under 1.0.

As we move from left to right in the figure, we see that the performance of the models generally improves as the number of ICL examples increases, and that there is variation in the model's rate of improvement with additional examples. GPT-4o, which performed the best with zero-shot learning, surprisingly shows the least improvement with five examples (the point directly to the right of the zero-shot point) in comparison to the other models, which make large gains in performance moving from zero-shot to five examples. We see that as we increase the number of examples to 25, deepseek-coder finds its plateau in performance roughly 0.43, which is the best performance of any model at that level of examples. Comparatively, both GPT-4o and claude-3-haiku learn less efficiently from the examples, requiring 50 examples to reach a similar level of performance.

In Figure 2.6b, we see that the performance of the models varies across the different dimensions of the DQI, with some dimensions being easier for the models to learn than others, and some models performing better on certain dimensions than others. All models perform the best on the participation dimension, with MAE scores close to 0 for all models at all levels of examples. For the level and content of justification dimensions, we see that while there is a similar pattern to the overall performance with GPT-4o performing the best with zero-shot learning and the error reducing across models as we increase the number of examples to five, we see that not all models improve with additional examples. In particular, we see that the error actually increases as the number of examples increase for deepseek-coder on the level of justification dimension and for GPT-4o on the content of justification dimension, suggesting that the models may struggle with these dimensions in particular.

Most models perform better on the respect (Groups, Demands, Counterarguments) and constructive politics dimensions as the number of examples increases. As in the overall performance, we see that deepseek-coder performs the best on these dimensions, particularly compared to GPT-4o on the respect for groups and respect for demands dimensions. For the respect for counterarguments dimension, we see that there is a high error rate for the claude-

3-haiku and deepseek-coder model at zero examples, but that error decreases to around 0.9 for the deepseek coder model with around 20 examples. We see a similar pattern for the constructive politics dimension, with the error decreasing to less than 0.5 for all models at 25 examples, with deepseek-coder again performing the best.

2.4.2 Model Cost

Figure 2.7 shows the relationship between performance and average cost per API query, compared to an estimated cost of either an expert or a crowdworker annotating the same speech. As before, the y-axis represents aggregate error, but here the x-axis represents the average cost per query in USD, where a query generates a single set of DQI annotations for a single speech. It is important to emphasize that the x-axis is on a logarithmic scale, so the cost of annotation increases exponentially as we move from left to right.

On the far right-hand side of this graph is a green dashed vertical line corresponding to the expected cost of an expert annotator, which I estimate to be $10 USD per speech. To the left is a yellow dashed vertical line, corresponding to the expected cost of a crowdworker annotator, which I estimate to be roughly $1 USD per speech. This difference amounts to a single order of magnitude, which is represented on the logarithmic scale as a one-unit difference on the x-axis.

With this benchmark in mind, we can intuitively compare the cost to performance ratio of each of the GPT-4o, DeepSeek Coder 2, and Claude 3 Haiku models to the cost of an expert or crowdworker annotator. At 25 examples (i.e, where the red line intersects with 10^{-1} on the x-axis), we can see that GPT-4o is roughly 10 times as costly as a crowdsourced annotator, and 100 times as costly as an expert annotator. Claude 3 Haiku at 50 examples is in turn 10 times cheaper than that, and DeepSeek Coder 2 at 5 examples is 10 times still.

All in all, these results show that deepseek-coder is both the most cost-effective and the best-performing model on the DQI task. Table 2.2, which provides a summary of the optimal performance of each model on the DQI task, along with the associated cost per million tokens for input and output, reinforces this conclusion, showing that DeepSeek Coder 2 has the best performance on the DQI task at the lowest cost, with an aggregate MAE of 0.48 at a cost of $0.14 per million tokens for input and $0.28 per million tokens for output.

Model	Provider	Parameters	Context	Cost ($/M tokens)		Optimal In-Context Learning Performance			
			Length	Input	Output	Examples	Accuracy	F1 Score	MAE
GPT-4o	OpenAI	N/A	128k	5.00	15.00	50	0.6382	0.6454	0.4595
Claude 3 Haiku	Anthropic	N/A	200k	0.25	1.25	25	0.6366	0.6343	0.5322
DeepSeek Coder 2	**DeepSeek**	**236b**	**128k**	**0.14**	**0.28**	**100**	**0.6826**	**0.6525**	**0.4799**
DeepSeek Coder 2 (CoT)	DeepSeek	236b	128k	0.14	0.28	25	0.6635	0.6304	0.4294
Llama 3 70B	Meta	70B	8k	0.65	2.75	5	0.6120	0.6023	0.5452
Llama 3 8B	Meta	8B	8k	0.05	0.25	5	0.5422	0.5530	0.7124
Qwen2 72B	Alibaba	72B	32k	0.90	0.90	25	0.6466	0.6110	0.4560
WizardLM 2	Microsoft	176B	32k	1.20	1.20	25	0.6466	0.6110	0.4560

Table 2.2: Model Comparison with Optimal In-Context Learning Performance for DQI Task

Figure 2.7 shows the relationship between average model accuracy and average cost per API query.[9] Table 2.2 provides a summary of the optimal performance of each model on the DQI task, along with the associated cost per million tokens for input and output, complementing the information provided in Figure 2.7.

2.5 Discussion

There are several key takeaways from these results. The first is that even in these preliminary results, we see evidence that generative language models can be used to automate the coding of complex constructs like the DQI, with the best-performing models achieving MAE scores of around 0.45 to 0.55 on the DQI dimensions. Across all models, MAE scores averaged across each dimension range between 0.68 and 1.1 even when no examples are provided to the model, suggesting that the models are able to generate high-quality annotations based on the theoretical and scoring information provided in the prompt.

Second, we see that many-shot in-context learning is very effective at reducing annotation error across most dimensions of the DQI, and in aggregate, with the "sweet spot" for the number of examples typically being around 25, after which the benefits of additional examples diminish. However, it is important to note that model performance does not uniformly improve across all dimensions of the DQI—in the Justification-level dimension, which asks how well their demands are justified by reasoning, MAE scores *increased* as the number of examples increased in all models except for GPT-4o, suggesting that the models may struggle with this dimension in particular.

Third, using the benchmark for human performance on the DQI task discussed above, we can say that an average MAE of 1.0 is roughly equivalent to the performance of a good human annotator on the DQI task. With this as a reference point, we see that in aggregate, the models are able to generate human-level annotations on the DQI task with just a few examples, and that the performance improves to what we might expect from an expert annotator in the range of 25 to 50 examples.

Fourth, we see that while costlier models like GPT-4o tend to perform better with few in-context examples than smaller models, the performance of these models converges as the number of examples included increases. Moreover, the results show that Deepseek Coder 2, despite being two orders of magnitude cheaper than GPT-4o, outperforms it in all comparisons except for the zero-shot, suggesting that cost-effective models can be competitive with

[9]The total cost over all model runs and experimentation totaled roughly $500 USD, with the plurality of the cost coming from the GPT-4o model—one full pass of the GPT-4o model across the 200 evaluation examples with 100 examples each cost roughly $70 USD.

more expensive models on the DQI task, and for complex annotation tasks more generally.

These are promising results. Given the complexity and subjectivity of the DQI coding task, the fact that large language models can generate annotations that are close to expert annotations is a significant achievement, and suggests that LLMs can be a valuable tool for automating the analysis of deliberative quality in political discourse.

That said, there is much work to be done to determine the reliability and validity of LLM-generated annotations, and to understand the conditions under which LLMs can be used to automate complex coding tasks in political science and related fields. There are many opportunities for future research in this area, including exploring the performance of LLMs on other qualitative coding tasks in political science and related fields, investigating the impact of different prompt structures and training techniques on model performance, and developing methods for evaluating the reliability and validity of LLM-generated annotations.

2.6 Conclusion

In this paper, I have presented a novel approach to automating the analysis of political discourse in parliamentary debates using large language models (LLMs). I have developed a prompt template for fine-tuning LLMs on the Discourse Quality Index (DQI), a widely used measure of political deliberation, and have conducted a series of experiments to evaluate the performance of several state-of-the-art LLMs on the DQI annotation task.

I then systematically compared the performance of these models on the DQI task to a human-validated benchmark of the DQI annotations, and analyzed the relationship between model performance and cost, to determine the feasibility and effectiveness of using LLMs to automate the analysis of political discourse in parliamentary debates.

I showed that LLM models such as OpenAI's GPT-4o, Anthropic's Claude 3 Haiku, and DeepSeek's DeepSeek Coder 2 can be used to automate the coding of the DQI in parliamentary speeches, generating annotations that meet or exceed our expectations of human annotations at a fraction of the cost. I also showed that the performance of these models can be improved by providing in-context learning examples, and that the cost of using these models can be significantly reduced by selecting the most cost-effective model for the task: in this case, DeepsSeek Coder 2 performed better than the far more expensive GPT-4o model with just a handful of illustrative examples.

The results shown in this paper demonstrate the potential of large language models to automate the analysis of political discourse in parliamentary debates. The first study shows that LLMs can be used to automate the coding of the Discourse Quality Index (DQI) in parliamentary speeches, generating annotations that are close to expert annotations and

providing a new tool for analyzing the quality of political discourse. The second study shows that LLMs can be used to generate directed graph representations of legislative debates, capturing the structure and dynamics of political discourse in a way that is interpretable, scalable, and reliable, and providing a new method for analyzing and understanding political discourse.

More broadly this study demonstrates the power of large language models to automate the analysis of political discourse, providing new tools and methods for studying the structure and quality of political discourse in parliamentary debates. By combining the DQI annotations with the graph representations, we can generate new metrics for analyzing the structure and quality of political discourse, and gain new insights into the dynamics of political discourse in legislative debates. This research has the potential to transform the way we analyze and understand political discourse, and to open up new avenues for research in the field of political science.

Future work in this area could explore the performance of LLMs on other qualitative coding tasks in political science and related fields, investigate the impact of different prompt structures and training techniques on model performance, and develop methods for evaluating the reliability and validity of LLM-generated annotations. There are many opportunities for future research in this area, and the potential for LLMs to transform the way we analyze and understand political discourse is vast.

CHAPTER 3

LegisGraph: A Graph-Based Method for Analyzing Legislative Debates Using Large Language Models

3.1 Introduction

The study of legislative debates is central to understanding the functioning of democratic institutions. These debates are the primary arena where elected representatives engage in deliberation, agenda-setting, and decision-making on public policy issues. As such, analyzing the content, quality, and dynamics of legislative discourse can provide valuable insights into the health and performance of a democracy.

However, legislative debates pose significant challenges for both quantitative and qualitative analysis. The sheer scale and complexity of the textual data generated by legislatures make it difficult to manually code or close read more than a small sample of debates. Traditional quantitative text analysis methods, such as topic modeling or dictionary-based approaches, can help to identify broad themes and patterns but struggle to capture the nuanced arguments, relationships between ideas, and quality of discourse.

In particular, there are three key analysis tasks that are essential for understanding legislative debates but remain difficult with existing methods. First, characterizing the distribution and relationships between the topics, subjects, and granular arguments that constitute the content of debates is challenging. Topic models can identify broad themes but provide limited insight into the specific subjects and ideas discussed and how they interact. Second, measuring the quality of discourse, in terms of dimensions such as participation, justification, respect, and constructiveness, at scale and over time is infeasible through manual coding and has not been systematically explored using computational methods. Third, existing measures of political polarization based on voting patterns or ideological scaling do not capture the more fine-grained dynamics of polarization reflected in the agreements and disagreements

between arguments made by legislators.

In this paper, I introduce LegisGraph, a new approach that aims to address these challenges by leveraging recent advances in large language models and combining them with network science and computational text analysis techniques. LegisGraph uses state-of-the-art natural language processing models to extract the topics, subjects, and arguments from the raw text of legislative debates, and represents these elements and their relationships in a graph data structure. This enables the efficient analysis of the content and dynamics of debates at multiple levels of granularity. Further, by linking arguments to the speakers who make them and the parties they belong to, LegisGraph allows for the study of polarization within and between parties based on the network structure of argument agreement and disagreement.

I apply LegisGraph to a dataset of debates in the Canadian House of Commons from 1994 to 2018. Using this data, I demonstrate the utility of my approach by undertaking three key analysis tasks. First, I characterize the distribution and network of topics, subjects, and arguments across the debates, revealing the parliamentary agenda to be dominated by economic policy issues such as budgets, taxation, and trade, and social policy issues like pensions, healthcare and education. Second, I compute discourse quality measures for each debate using the Discourse Quality Index and track their evolution over time and across key covariates. Then, I show how we can use these measures to quantify concepts such as polarization and party cohesion, based on the network structure of agreements and disagreements between speakers' arguments.

Together, these analyses demonstrate the potential of LegisGraph to provide novel insights into legislative debates and polarization that were previously difficult or impossible to obtain at scale. By integrating the strengths of large language models for extracting granular argumentative content, network science for analyzing the relationships between ideas and actors, and discourse quality measures for evaluating the health of deliberation, LegisGraph offers a powerful new toolkit for legislative studies. More broadly, this approach opens up new avenues for studying the dynamics of political discourse and deliberation, with important implications for democratic theory and efforts to strengthen the functioning of representative institutions in an era of declining trust and rising polarization.

The remainder of this paper is organized as follows. Section 3.2 provides an overview of related work in the quantitative analysis of legislative debates and the application of language models to natural language processing tasks. Section 3.3 details the methodology of the LegisGraph approach, including the generation of structured graph representations from raw legislative text and the iterative refinement of prompts to guide the large language models. Section 3.4 presents the results of applying LegisGraph to the Canadian parliamentary

debates dataset, including the characterization of topics and arguments, the measurement of discourse quality, and the quantification of polarization. Section 4.6 discusses the implications of these findings and outlines future research directions. Finally, Section 4.7 concludes the paper with a summary of the key contributions and insights of the LegisGraph approach.

3.2 Previous Work

This paper builds upon and integrates two distinct but increasingly intersecting areas of research: the quantitative analysis of legislative debates in political science and the application of language models to natural language processing (NLP) tasks. This section provides an overview of key developments in these fields that inform and contextualize the LegisGraph approach.

3.2.1 Quantitative Analysis of Legislative Debates

The quantitative analysis of legislative debates has a rich history in political science, with researchers employing various methods to extract insights from the vast corpus of legislative texts. Early work in this field often relied on manual coding of speeches, a labor-intensive process that limited the scale of analysis (Mayhew, 1991). Similarly, qualitative content analysis methods were used to measure phenomena such as discourse quality Steenbergen et al. (2003), but these approaches faced challenges in terms of reliability and replicability when applied to large-scale textual data.

However, the advent of computational methods has dramatically expanded the scope and scale of legislative text analysis. One significant strand of research has focused on using text as data to measure policy positions and ideological leanings of legislators. Laver et al. (2003) pioneered the use of computerized content analysis to estimate policy positions from political texts, including legislative speeches. Building on this work, Slapin and Proksch (2008) developed Wordfish, an automated scaling model that estimates policy positions based on word frequencies in political texts. These methods have been widely applied to legislative debates to track ideological shifts over time and across different political systems (Lauderdale and Herzog, 2016).

Another important area of research has centered on measuring the substantive content of legislative speeches. Topic modeling techniques, particularly Latent Dirichlet Allocation (LDA) (Blei et al., 2003), have been extensively used to identify thematic structures in large corpora of legislative texts. For instance, Quinn et al. (2010) applied a dynamic topic model to U.S. Senate speeches to track the evolution of policy issues over time. Similarly, Grimmer

and Stewart (2013) developed expressed agenda models to capture the diversity of topics legislators discuss in their communications.

While these methods have greatly advanced our understanding of the content and structure of legislative debates, they face limitations in terms of capturing the granular arguments and deliberative quality of these discussions. Topic models, for example, can identify broad themes but provide limited insight into the specific ideas and arguments expressed within these themes and how they interact with one another. Similarly, automated scaling methods based on word frequencies may not fully capture the nuanced positions and justifications articulated by legislators. Moreover, traditional methods struggle to measure the quality of discourse and deliberation at scale, which is crucial for understanding the health and functioning of legislative bodies. The Discourse Quality Index (DQI) (Steenbergen et al., 2003) provides a framework for manually coding various dimensions of discourse quality, such as participation, justification, respect, and constructive politics. However, applying the DQI to large corpora of legislative debates is prohibitively time-consuming and costly. Computational adaptations of the DQI have been limited in their ability to fully capture the complex components of high-quality deliberation (Steiner et al., 2012).

Researchers have also sought to measure more abstract aspects of legislative behavior through text analysis. Proksch and Slapin (2010) used text scaling methods to analyze parliamentary questions and estimate the degree of government oversight. Lauderdale and Herzog (2016) developed a model to measure debate participation and polarization in the U.S. Congress, demonstrating how text analysis can shed light on legislative dynamics beyond just policy positions. However, these approaches still face challenges in terms of capturing the fine-grained interactions and polarization between legislators that occur at the level of individual arguments and ideas.

In recent years, there has also been growing interest in applying network analysis methods to study legislative behavior and political discourse. Co-voting networks have been used to analyze coalitions and polarization in roll call votes (Waugh et al., 2009), while cosponsorship networks have provided insight into patterns of legislative collaboration (Fowler, 2006). However, the application of network methods to the analysis of legislative debates has been more limited. Some studies have constructed co-occurrence networks based on interviews (Arnold et al., 2000) or analyzed the structure of citation networks in legislative documents (Wilkerson et al., 2015), but these approaches do not capture the semantic content and argumentative structure of debates.

To address these limitations, there is a need for new methodological approaches that can (1) extract the granular topics, subjects, and arguments from legislative debates at scale; (2) measure multiple dimensions of discourse quality in an automated and reliable way; and

(3) model the complex interactions and polarization between arguments and ideas. The LegisGraph approach aims to fill this gap by leveraging recent advances in natural language processing, network science, and discourse quality measurement.

3.2.2 Language Models in NLP Tasks

The field of Natural Language Processing (NLP) has been transformed by the development of large language models (LLMs) based on transformer architectures (Vaswani et al., 2017). These models, pre-trained on vast corpora of text, have demonstrated remarkable capabilities across a wide range of NLP tasks.

The introduction of BERT (Bidirectional Encoder Representations from Transformers) by Devlin et al. (2019b) marked a significant milestone in the development of LLMs. BERT's bidirectional training approach allowed it to develop a deeper understanding of language context, leading to state-of-the-art performance on various NLP benchmarks. Following BERT, a series of increasingly large and sophisticated models have been developed, including GPT (Generative Pre-trained Transformer) models (Radford et al., 2018; Brown et al., 2020), T5 (Raffel et al., 2020), and more recent models like GPT-3 (Brown et al., 2020) and PaLM (Chowdhery et al., 2022).

These LLMs have shown impressive performance in tasks ranging from text classification and named entity recognition to question answering and text generation. Of particular relevance to the current work is their ability to perform few-shot and zero-shot learning, allowing them to adapt to new tasks with minimal or no task-specific training data (Brown et al., 2020).

Traditionally, machine learning models, including those used in NLP, required extensive task-specific training data and fine-tuning to perform well on new tasks. However, the advent of large language models has introduced a new paradigm: prompt engineering. Prompt engineering is the practice of carefully crafting input prompts to guide the behavior of pre-trained language models without the need for additional training or fine-tuning. This approach leverages the vast knowledge and capabilities already embedded in these models, allowing researchers and practitioners to "steer" the models towards desired outputs or behaviors simply by formulating their queries or instructions in specific ways.

Researchers have begun to explore the application of LLMs and prompt engineering to various domains in the social sciences. For instance, Buchanan and et al. (2021) demonstrated the potential of GPT-3 for coding open-ended survey responses, and Haoran and et al. (2022) explored the use of LLMs for automated content analysis in communication research.

However, the application of LLMs to legislative text analysis remains relatively unexplored. While some researchers have used earlier transformer models for tasks like classifying legislative speeches (Davoodi and et al., 2020), the full potential of state-of-the-art LLMs and prompt engineering techniques for analyzing the complex, contextual nature of legislative debates has yet to be realized.

The LegisGraph approach aims to address this gap by leveraging the power of LLMs to extract granular arguments and their relationships from legislative debates, and combining this with network analysis and discourse quality measures to provide novel insights into the dynamics of legislative discourse and polarization. By integrating these cutting-edge methods from NLP, network science, and political communication, LegisGraph offers a unique and promising framework for advancing the computational study of political discourse.

By demonstrating the potential of LLMs and prompt engineering for extracting rich, structured data from unstructured legislative texts, this work contributes to ongoing methodological debates in political science about the role and value of computational methods for studying complex social and political phenomena. The ability to analyze the granular content and dynamics of legislative debates at scale using LLMs opens up new possibilities for testing and refining theories of deliberative democracy, political polarization, and legislative behavior.

The LegisGraph approach also highlights important challenges and limitations that need to be considered when applying LLMs to social science research, such as the interpretability and potential biases of these models (Weidinger et al., 2022), the role of human judgment and domain expertise in guiding their use, and the need for careful validation and robustness checks (Ji et al., 2023). By engaging with these methodological and substantive questions, this work aims to contribute to the growing interdisciplinary dialogue between computational social science, political methodology, and democratic theory.

3.2.3 Knowledge Graphs and Relational Databases

In addition to leveraging advances in natural language processing, the LegisGraph approach also builds upon work in knowledge representation and management using knowledge graphs and relational databases.

Knowledge graphs are structured representations of information that model entities and their relationships in a graph format, with nodes representing concepts and edges capturing the connections between them (Ji et al., 2021). This flexible and intuitive framework has been widely used in various domains, from search engines and recommendation systems to biomedical research and social network analysis, to represent and reason over large-scale,

heterogeneous data (Hogan, 2022).

In the context of computational social science, knowledge graphs have been applied to model and analyze complex social and political phenomena, such as studying the spread of misinformation (Cui and Lee, 2020). By representing entities and their relationships in a graph format, knowledge graphs enable the use of powerful network analysis techniques (e.g., centrality measures, community detection) to uncover patterns and insights that may not be apparent from unstructured text data alone.

Relational databases, on the other hand, provide a structured way to store, manage, and query large amounts of data using tables and predefined relationships between them (Codd, 1970). Widely used in various industries and research domains, relational databases enable efficient storage, retrieval, and analysis of structured data, and support complex queries and data manipulations using SQL (Structured Query Language) (Chamberlin and Boyce, 1974).

In the social sciences, relational databases have been used to manage and analyze large-scale datasets, such as surveys, administrative records, and text corpora. For example, the Comparative Agendas Project (Baumgartner and Jones, 2019) uses a relational database to store and organize coded data on policy agendas across multiple countries and time periods, enabling comparative analysis of policy priorities and attention. Similarly, the Database on Ideology, Money in Politics, and Elections (Bonica, 2015) integrates data on campaign finance, lobbying, and legislative behavior into a relational database to study the influence of money in U.S. politics.

However, the application of knowledge graphs and relational databases to the study of legislative debates has been limited. Overall, the potential of integrating knowledge graph representations of debates with relational databases for large-scale analysis remains largely unexplored.

The LegisGraph approach aims to fill this gap by combining the strengths of knowledge graphs for representing the complex relationships between speakers, arguments, and topics in legislative debates, with the power of relational databases for efficient storage, management, and querying of this structured data. By extracting a knowledge graph representation of debates using large language models and storing this in a relational database, LegisGraph enables researchers to perform complex analyses and queries to investigate the content, quality, and dynamics of legislative discourse at an unprecedented scale and granularity.

3.3 Methods

The LegisGraph approach integrates three key methodological components to enable novel and granular analyses of legislative debates: (1) extraction of argumentation graphs from

unstructured debate transcripts using large language models (LLMs) and prompt engineering; (2) representation of debates as knowledge graphs with nodes for speakers, arguments, topics, and other key entities, and edges capturing the relationships between them; and (3) storage and analysis of debate graphs using relational databases and graph-based measures of discourse quality and polarization.

This integration allows LegisGraph to address several limitations of existing methods for analyzing legislative discourse. First, by leveraging the power of LLMs to identify and extract granular arguments and their relationships, LegisGraph goes beyond the capabilities of traditional topic models or word-frequency based methods, which provide a more coarse-grained and less contextually nuanced representation of debate content (Quinn et al., 2010; Grimmer and Stewart, 2013). Second, by representing debates as knowledge graphs, LegisGraph can capture the complex, multi-relational structure of legislative discourse, including interactions between speakers, arguments, and topics, which are difficult to model using simple word co-occurrence or document similarity measures (Wilkerson et al., 2015). Finally, by storing and analyzing debate graphs in a relational database, LegisGraph enables efficient querying and computation of novel graph-based measures of discourse quality and polarization at scale, overcoming the limitations of manual content analysis or dictionary-based methods (Steiner et al., 2012).

Algorithm 1 outlines the LegisGraph Generation Procedure, which transforms raw legislative speeches into structured graph representations. The algorithm consists of four main steps: data preparation (Section 3.3.1) to format the speeches into debate-level documents, prompt refinement (Section 3.3.2) to construct the prompts each task, graph generation (Section 3.3.3) to create structured graph representations of debates, and DQI+ annotation (Section 3.3.4) to enrich the graphs with detailed discourse quality metrics. I also discuss the validation process (Section 3.3.6) to ensure the reliability and accuracy of the generated graphs. The resulting graph representations are then integrated into a relational database (Section 3.3.5) for further analysis.

3.3.1 Data Preparation

Before applying the LLM-based graph generation process, I organize the raw speech data into coherent debate structures. This preparation step ensures that the LLM receives a well-structured input that captures the full context of each debate. The process involves aggregating individual speeches and their associated metadata into debate-level documents.

The source data consists of individual speeches from parliamentary proceedings, each associated with metadata such as the speaker's name, party affiliation, position, speech ID,

Algorithm 1 LegisGraph Generation Procedure

Require: Set of legislative speeches S, LLM model M, Graph generation prompt P_{Graph}, DQI annotation prompt P_{DQI}

Ensure: Metagraph G representing all debates with DQI+ annotations

1. Data Preparation:
Group speeches in S by debate.
Format debates into standardized documents, denoted as *Debates*.
Initialize empty metagraph G.

2. Prompt Refinement:
Generate P_{Graph} and P_{DQI} using Algorithm 2.

for all debate d in *Debates* **do**
  3. Generate Graph:
  Apply M to d using graph generation prompt P_{Graph}.
  Store LLM output as initial representation R_d.

  4. DQI+ Annotation:
  for all speaker-argument edges e in R_d **do**
    Apply M to e using DQI+ prompt P_{DQI}.
    Update e with DQI+ annotations.
  end for

  5. Integration with Metagraph:
  Extract nodes (speakers, arguments, topics) and edges (relationships) from refined R_d.
  Construct debate graph G_d from extracted nodes and edges.
  Merge G_d into meta graph G.
  Update meta graph with debate-level metadata.
end for

return Populated metagraph G.

and the date of the speech. This data is typically stored in a structured format, such as a relational database or CSV files.

To transform data into debate-level documents, I developed a Python script that groups speeches by date and topic, allowing those that occur on the same date and share a common main topic to be considered part of the same debate. The script then extracts and compiles essential metadata for each speech group, including the debate date, parliament, session, main topic, subtopic (if applicable), subsubtopic (if applicable), the number of speeches in the debate, and the total word count across all speeches. Within each debate, speeches are ordered chronologically by their speech ID, which typically reflects the order of delivery. Finally, the script formats this information into a standardized structure, beginning with the debate-level metadata, followed by the individual speeches, each separated by a delimiter.

The resulting debate document follows this structure:

```
Debate Date: [DATE]
Parliament: [PARLIAMENT NAME]
Session: [SESSION NUMBER]
Main Topic: [MAIN TOPIC]
Subtopic: [SUBTOPIC OR "none"]
Subsubtopic: [SUBSUBTOPIC OR "none"]
Number of Speeches: [NUMBER]
Total Number of Words: [WORD COUNT]

----------------------

Speaker: [SPEAKER NAME]
Position: [POSITION OR "none"]
Party: [PARTY AFFILIATION]
Speech ID: [SPEECH ID]
Speech Text: [FULL TEXT OF THE SPEECH]

----------------------

[NEXT SPEECH...]
```

This format provides a comprehensive view of the debate, including its context and all constituent speeches, in a single document. The use of consistent delimiters ("——————————") between speeches allows for easy parsing by both humans and machines.

3.3.2 Prompt Refinement Procedure

In order to use LLMs to generate structured graph representations of legislative debates, a prompt must be designed to guide the model in extracting relevant information and transforming it into a graph format. Algorithm 2 outlines the iterative process used to refine the

Algorithm 2 Prompt Refinement Procedure

Require: Initial prompt P_0, Set of sample debates D, LLM model M, Evaluation criteria E

Ensure: Refined prompt P_f for graph generation

1. Initial Prompt Design:
Design P_0 based on Bosley (2024) findings
a. Detailed annotation instructions
b. Schema for output structure

2. Synthetic Example Generation:
Collaborate with M to create diverse synthetic examples S
Ensure S covers various debate structures and complexities
Add S to P_0

3. Iterative Refinement:
$i \leftarrow 0$
while performance not satisfactory **do**
$i \leftarrow i + 1$
3.1 Sample Output Generation:
for all debates d in D **do**
Apply M to d using current prompt P_i
Store output as O_d
end for

3.2 Manual Review:
Evaluate O_d against criteria E
Identify set of areas for improvement A

3.3 Prompt Revision:
$P_{i+1} \leftarrow P_i$
for all improvement areas a in A **do**
Update P_{i+1} based on type of improvement a:
- Modify instructions if a is an instruction-related issue
- Refine or add examples if a relates to examples
- Adjust schema for output if a requires schema changes
end for
end while
$P_f \leftarrow P_i$

return Refined prompt P_f

prompts for both graph generation and Discourse Quality Index (DQI) annotation.

At a high level, this procedure follows the approach articulated by Bosley (2024), which emphasizes the importance of detailed annotation instructions, high quality examples that cover a breadth of the potential cases that might be encountered, and a schema for the output structure. In that work, however, it was assumed that the research would be starting from an existing codebook and pre-validated set of examples to draw from. In this case, we are starting from scratch, and new techniques are required in order to develop the initial prompt and examples.

To this end, I propose a three-step process for prompt generation that mirrors the iterative process of codebook development in qualitative research, where iterative refinement is used to ensure that the codebook captures the full range of cases and is reliable and valid: initial prompt design, synthetic example generation, and iterative refinement based on manual review and a set of evaluation criteria.

1. Initial Prompt Design The algorithm begins with an initial prompt design, which serves as the foundation for subsequent iterations. These instructions provide a detailed framework for the LLM to follow in translating raw text into structured representations. This initial prompt is designed based on the researcher's understanding of the task and the capabilities of the LLM, and serves as a starting point for further refinement. The schema for the output structure should define the desired format of the structured output of the LLM, and can be specified in a variety of formats, such as a Markdown list, a JSON object, or a graph schema language like Mermaid (Knut, 2019).

2. Synthetic Example Generation Synthetic examples are then generated in collaboration with a high capacity LLM to ensure a diverse set of cases is covered. I chose Claude 3.5 Sonnet, a state-of-the-art LLM developed by Anthropic, for prompt refinement due to its advanced natural language understanding and ability to handle complex, multi-step instructions.[1] I accessed the model via Anthropic's chatbot interface[2] for prompt refinement, as it allows for interactive exploration of the model's capabilities and rapid prototyping of prompts.

The primary motivation for using synthetic examples rather than annotating real examples is to have precise control over the way that the model handles particular types of cases. By generating synthetic examples, we can ensure that the model encounters a diverse set of cases that cover the full range of potential scenarios it might encounter in real data without

[1] However, the approach outlined here is model-agnostic and can be applied to any LLM with similar capabilities.

[2] See https://claude.ai/

having to rely on the availability of such cases in the training data. Put another way, using synthetic examples allow us to build a robust, generalizable prompt that matches closely to the conceptual framework that the research is trying to capture, rather than being limited by the idiosyncrasies of the training data. Consequently, synthetic data can be generated abstractly enough that they are not overfit to specific topic areas, debate structures, or historical contexts. This allows the model to learn general principles of debate structure and discourse quality that can be applied across a wide range of legislative debates.

Using synthetic examples to guide the model's annotation process also has the advantage of naturally accommodating a sort of model distillation process, whereby the understanding of the construct that is being measured is encoded in a high level of detail by an expensive high-capacity model in collaboration with the researcher. This prompt can then be implemented at scale by a lower-capacity model that may have struggled to provide high quality annotations in the absence of this detailed guidance. Conceptually, this is not unlike the process of a human expert generating a codebook for a team of human coders to use, where the expert's deep understanding of the construct being measured is distilled into a set of detailed instructions that can be followed by less experienced coders.

3. Iterative Refinement After deciding on a set of instructions and preliminary set of synthetic examples, the next step is to iteratively refine the prompt through a process of manual review and evaluation. In each iteration, the current prompt is applied to a set of randomly chosen sample debates, and the outputs are manually reviewed against predefined evaluation criteria. For the DQI prompt, this involves assessing the accuracy and consistency of the LLM's annotations across various aspects of discourse quality. For the graph representation prompt, evaluation focuses on the clarity, completeness, and accuracy of the generated Mermaid graphs.

Based on this review, areas for improvement are identified, which could involve modifying instructions, refining examples, or adjusting the output schema. Common challenges addressed during this process include balancing detail and clarity in the instructions, handling edge cases such as unusual debate structures, maintaining consistency across different debates and LLM runs, and adapting to unexpected behaviors of the LLM.

The refinement process continues until satisfactory performance is achieved – i.e., the LLM consistently generates accurate and coherent graph representations on a diverse set of debates. The resulting prompts provide robust guidance to the LLM, including context consideration, detailed criteria explanations, scoring instructions, and output format specifications, and debate analysis instructions, and node and edge specifications for graph

outputs.[3]

3.3.3 Graph Generation

Creating the Graph Generation Prompt I use the approach outline above to develop the graph generation prompt (P_{Graph}) used in Step 3 of Algorithm 1. The prompt is designed to guide the LLM in transforming raw debate transcripts into structured graph representations that capture the relationships between speakers, arguments, topics, and procedural elements. The prompt includes detailed instructions, synthetic examples, and a schema for the output structure, ensuring that the LLM produces consistent and accurate graph representations across a wide range of debates.

While Appendix B.1 provides a detailed explanation of the graph generation prompt (including a complete annotated example of the prompt that is submitted to an LLM, the structure of the graph representation output, and an overview of the iterative process used to refine the prompt), I summarize the key components here. The prompt includes high-level instructions that outline the task and elicit a two-step analysis process: first, analyzing the debate transcript to identify key elements, and second, translating these elements into a Mermaid graph structure. The prompt also includes synthetic examples of debate analyses and their corresponding graph representations, and a schema for the output structure. The schema defines the types of nodes (e.g., speakers, arguments, topics) and edges (e.g., "makes", "addresses", "contradicts") that should be used in the graph representation. The prompt also includes guidance on using Mermaid syntax (Knut, 2019) to create the graph representation, ensuring that the model produces output that is compatible with existing visualization tools and can be easily integrated into the LegisGraph database system.

Implementing the Graph Generation Prompt at Scale The graph generation prompt is implemented by submitting it to the LLM model, which processes the debate transcripts and generates structured graph representations in Mermaid syntax. For this stage, I used the GPT-4o-mini model, a variant of the GPT-4o model developed by OpenAI, due to its strong performance when applied to a set of test debates, computational efficiency (each debate can be processed using the OpenAI API in under a second, and at a cost of roughly $0.005 USD per debate), and the high throughput of the OpenAI API, which allows for parallel processing of multiple debates.[4]

[3]While the process detailed here is somewhat *ad hoc*, it is intended to be a starting point for the development of more systematic methods for prompt refinement. In the discussion section, I discuss potential avenues for further research in this area, including the development of automated prompt refinement techniques that formalize the process described here and reduce the need for manual intervention.

[4]I set all hyperparameters to their default value.

To provide some intuition for the graph generation process, consider the following excerpt of a debate transcript between two speakers, Jean Chretien and Monte Solberg, in the Canadian Parliament. The transcript captures a heated exchange between the two speakers on the topic of gun control, with Chretien accusing Solberg's party of inconsistency in their stance on the issue. The LLM processes this text and generates a structured graph representation that captures the relationships between the speakers, their arguments, and the topics under discussion. The resulting graph provides a visual representation of the debate dynamics, highlighting the interactions between the speakers and the flow of arguments.[5]

```
Debate Date: 1967-11-20 00:00:00
Parliament: Twentieth Parliament
Session: 2nd
Main Topic: PROCEEDINGS ON ADJOURNMENT
    MOTION
Subtopic: SUBJECT MATTER OF QUESTIONS
   TO BE DEBATED
Subsubtopic: ALLEGED ECONOMIC
   MISMANAGEMENT BY GOVERNMENT
Number of Speeches: 13
Total Number of Words: 3877

----------------------

Speaker: Marcel Joseph Lambert
Position: none
Party: Progressive Conservative
Speech ID: 2515910
Speech Text: It was the first time
   that such a conference had been
   held.

----------------------

Speaker: Marcel Joseph Lambert
Position: none
Party: Progressive Conservative
Speech ID: 2515912
Speech Text: I caution, hon. members
   and Mr. Speaker, that most
   decisions of government ought not
   to be taken at dominion-provincial
    conferences. That pattern ought
   not to be set at such conferences,
    since this house would become a
   mere rubber stamp. This is the
   dilemma we face. I agree with what
    the Leader of the Opposition said
    this afternoon. There must be
   consultation with the provinces
   [...]

----------------------

Speaker: John Ross Matheson
Position: Parliamentary Secretary to
   the Prime Minister
Party: Liberal
Speech ID: 2515913
Speech Text: Mr. Speaker, in joining
   this debate I shall try to meet
   the challenges posed by the
   amendment of the Leader of the
   Opposition (Mr. Stanfield) and the
    subamendment of the hon. member
   for Burnaby-Coquitlam (Mr. Douglas
   ). In brief, the Leader of the
   Opposition charges the government
   with economic mismanagement.

----------------------

Speaker: Jay Waldo Monteith
Position: Progressive Conservative
   Party Caucus Chair
Party: Progressive Conservative
Speech ID: 2515914
Speech Text: Hear, hear.

----------------------

Speaker: John Ross Matheson
Position: Parliamentary Secretary to
   the Prime Minister
```

[5]This example was chosen randomly from the dataset, and was not cherry-picked for its quality or complexity.

```
Party: Liberal
Speech ID: 2515915
Speech Text: During his speech he said
   , as I recall, that he hoped to
   speak later and in greater depth
   on the budget. I take it that we,
   on this side, may regard ourselves
    as quite safe until the mini-
   budget debate.
When the Leader of the Opposition
   attacks the government for
   mismanagement I say, at the outset
   , that he is attacking the most
   professionally competent Minister
   of Finance we have had in Canada's
    history.

----------------------

Speaker: An hon. Member:
Position: none
Party: none
Speech ID: 2515916
Speech Text: Boo.

----------------------

Speaker: John Ross Matheson
Position: Parliamentary Secretary to
   the Prime Minister
Party: Liberal
Speech ID: 2515917
Speech Text: Certainly, he has had the
    best education in economics.

----------------------

Speaker: An hon. Member:
Position: none
Party: none
Speech ID: 2515919
Speech Text: And the ex minister of
   transport.

----------------------

Speaker: John Ross Matheson
Position: Parliamentary Secretary to
   the Prime Minister
Party: Liberal
Speech ID: 2515920
Speech Text: He has held a high
   position in industry and is well
   aware of the responsibilities of
   public office. From searching
   through budget debates sedulously
   I am satisfied that never have the
    Canadian people had as lucid and
   as technical explanations of
   Canada's financial affairs as we
   have had since our present
   Minister of Finance (Mr. Sharp)
   assumed his office.
I will be very brief in my remarks
   with respect to the subamendment
   moved by the hon. member for
   Burnaby-Coquitlam (Mr. Douglas)
   [...]
```

This debate is inserted into the prompt (P_{Graph}) and submitted to the LLM model, which processes the text and generates the following structured representation, which includes an analysis of the speakers, their arguments, the topics and subjects under discussion, the relationships between them, and a paragraph summarizing the key points of the debate.

```
## Analysis of the Debate

1. **Parties**: The debate involves
   three parties: Progressive
   Conservative, Liberal, and none (
   independent members).

2. **Speakers**:
  - Marcel Joseph Lambert (Progressive
      Conservative)
  - John Ross Matheson (Liberal)
  - Jay Waldo Monteith (Progressive
     Conservative)
  - Robert Lorne Stanfield (
     Progressive Conservative)
  - David Vaughan Pugh (Progressive
     Conservative)
  - An hon. Member (unnamed, none)

3. **Topics and Subjects**:
```

- **Topics**:
  1. Economic Policy
  2. Social Policy
- **Subjects**:
  1. Economic Mismanagement
  2. Productivity
  3. Government Spending
  4. Employment Rates

4. **Debate Node**: Proceedings on Adjournment Motion, 1967-11-20, Twentieth Parliament, 2nd Session

5. **Procedural Motions**: No specific procedural motions were identified in the provided text.

6. **Arguments**:
   - **Argument A**: The government has mismanaged the economy, leading to poor productivity and economic challenges. (Marcel Joseph Lambert, Speech ID: 2515912)
   - **Argument B**: The government has competent leadership and a professional Minister of Finance who has provided clarity on economic issues. (John Ross Matheson, Speech ID: 2515915)
   - **Argument C**: Canada has had notable successes in employment and trade, countering claims of mismanagement. (John Ross Matheson, Speech ID: 2515922)
   - **Argument D**: The government spending on defense and other areas should not be cut, as this could jeopardize national interests. (John Ross Matheson, Speech ID: 2515920)
   - **Argument E**: The opposition's claims lack substance and do not acknowledge the complexities of economic conditions in Canada. (John Ross Matheson, Speech ID: 2515924)

7. **Relationships**:
   - **Inter-Argument Relationships**:
     - Argument A contradicts Argument B by asserting mismanagement versus competent leadership.
     - Argument C supports Argument B by showcasing successes to refute claims of mismanagement.
     - Argument D supports Argument C by indicating that cutting spending could lead to negative outcomes.
     - Argument E supports Argument B by criticizing the opposition's lack of depth in their analysis.

   - **Argument-Subject Relationships**:
     - Argument A addresses Economic Mismanagement and Productivity.
     - Argument B addresses Government Spending and Economic Mismanagement.
     - Argument C addresses Employment Rates and Economic Mismanagement.
     - Argument D addresses Government Spending and National Security.
     - Argument E addresses Economic Mismanagement and the credibility of opposition claims.

8. **Summary**:
   This debate centered around the alleged economic mismanagement by the government. Marcel Joseph Lambert from the Progressive Conservative Party criticized the government's economic policies and claimed that Canada was following a path of mismanagement similar to that of Britain. In contrast, John Ross Matheson from the Liberal Party defended the government's actions, highlighting the professional competence of the Minister of Finance and the successes in employment and trade. The discussions revealed a divide between those who viewed the economic situation as dire and those who believed in the government's handling of economic affairs, emphasizing the complexities of economic policy and the importance of constructive criticism.

The model then generates a graph representation using Mermaid syntax, each node represents a speaker, argument, topic, or subject, and each edge represents a relationship between these elements. The resulting graph visually represents the structure of the debate, the topics discussed, the arguments made, and the relationships between them. Each node and edge is clearly labeled to reflect its role in the debate.

```
graph TD
   %% Debate and Topics
   D[Debate: Proceedings on
      Adjournment Motion<br
      >1967-11-20, Twentieth
      Parliament, 2nd Session]:::
      debate
   T1[Topic: Economic Policy]:::topic
   T2[Topic: Social Policy]:::topic
   D -->|debate about| T1
   D -->|debate about| T2

   %% Subjects
   S1[Economic Mismanagement]:::
      subject
   S2[Productivity]:::subject
   S3[Government Spending]:::subject
   S4[Employment Rates]:::subject
   S1 -->|discussed during| D
   S2 -->|discussed during| D
   S3 -->|discussed during| D
   S4 -->|discussed during| D

   %% Parties
   P1[Progressive Conservative]:::
      party
   P2[Liberal]:::party

   %% Speakers
   Spk1[Marcel Joseph Lambert]:::
      speaker
   Spk2[John Ross Matheson]:::speaker
   Spk3[Jay Waldo Monteith]:::speaker
   Spk4[Robert Lorne Stanfield]:::
      speaker
   Spk5[David Vaughan Pugh]:::speaker
   Spk6[An hon. Member]:::speaker

   %% Arguments
   A1[Government has mismanaged the
      economy]:::argument
   A2[Government has competent
      leadership and finance minister
      ]:::argument
   A3[Canada has had notable successes
       in employment and trade]:::
      argument
   A4[Defense spending should not be
      cut]:::argument
   A5[Opposition lacks substance in
      claims]:::argument

   %% Party-Speaker Relationships
   P1 -->|party of| Spk1
   P1 -->|party of| Spk3
   P1 -->|party of| Spk4
   P1 -->|party of| Spk5
   P2 -->|party of| Spk2

   %% Speaker-Argument Relationships
   Spk1 -->|makes -- 2515912| A1
   Spk2 -->|makes -- 2515915| A2
   Spk2 -->|makes -- 2515922| A3
   Spk2 -->|makes -- 2515920| A4
   Spk2 -->|makes -- 2515924| A5

   %% Argument-Subject Relationships
   A1 -->|addresses| S1
   A1 -->|addresses| S2
   A2 -->|addresses| S3
   A3 -->|addresses| S4
   A4 -->|addresses| S3
```

```
A5 -->|addresses| S1

%% Inter-Argument Relationships
A1 -.->|contradicts| A2
A2 -->|supports| A3
A4 -->|supports| A3
A5 -->|supports| A2

%% Styling
classDef debate fill:#C7CEEA,stroke
   :#4B0082,stroke-width:2px;
classDef topic fill:#FF9999,stroke
   :#FF0000,stroke-width:2px;
classDef subject fill:#A9DFBF,
   stroke:#27AE60,stroke-width:2px
   ;
classDef party fill:#8E44AD,stroke
   :#8E44AD,stroke-width:2px;
classDef speaker fill:#AED6F1,
   stroke:#3498DB,stroke-width:2px
   ;
classDef argument fill:#FAD7A0,
   stroke:#F39C12,stroke-width:2px
   ;
```

This graph can then be visually rendered using Mermaid or other graph visualization tools, providing a structured representation of the debate's dynamics. Each node is labeled with the corresponding speaker, argument, topic, or subject, and each edge is labeled to reflect the relationship between the nodes. This visual representation of the graph is shown in Figure 3.1.

3.3.4 DQI+ Annotation

Having generated a graph representation of the debate, where each node represents a speaker, argument, topic, or subject, and each edge represents a relationship between these elements, the next step is to enrich the graph with detailed discourse quality metrics. This is achieved through the Discourse Quality Index Plus (DQI+) annotation process, which evaluates the quality of discourse in the debate based on six criteria: participation, justification, respect, constructive politics, relevance to topic, and tone.

DQI+ Annotation Criteria The DQI+ criteria are designed to capture the key dimensions of discourse quality in legislative debates. I build on the original DQI framework (see e.g. Steenbergen et al. (2003)), which has been widely used in political science research to assess the quality of political discourse. The original DQI framework provides guidelines for evaluating the deliberative quality of political speeches based on four main criteria: participation, justification, respect, and constructive politics. Participation refers to the extent to which speakers are freely able to contribute to the discussion; justification evaluates the level and quality of the justifications for the speakers' claims; respect assesses the degree of respectfulness towards other groups, demands, and counterarguments; and constructive politics measures the extent to which speakers engage in constructive dialogue and seek common ground versus engaging in adversarial or unproductive behavior. Each of these criteria

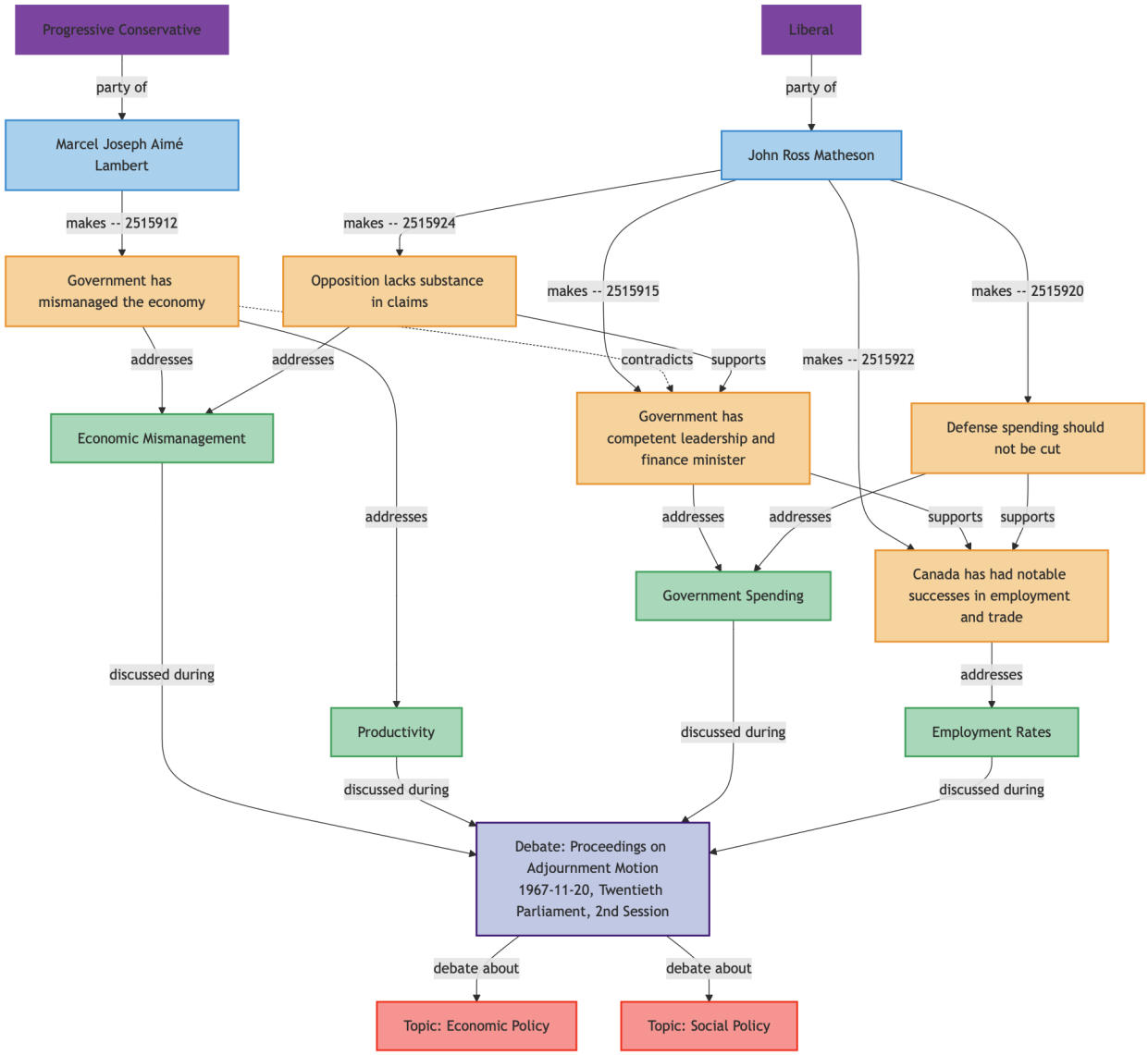


Figure 3.1: **Structured Graph Representation of the Debate.** This graph visually represents the structured output generated by the LLM, capturing the relationships between parties (purple nodes) speakers (light blue nodes), arguments (yellow nodes), subjects (green nodes), debate (dark blue node), and topics (red nodes). Each edge represents a relationship between nodes: party-speaker relationships, speaker-argument relationships, argument-subject relationships, and inter-argument relationships. The edges are labeled to indicate the nature of the relationship, such as makes, addresses, contradicts, and supports.

is scored on an ordinal scale, with higher scores indicating higher quality discourse.

I extend the original DQI framework by adding three additional criteria: respect towards in-party and out-party members, relevance to topic, and tone. Respect towards in-party and out-party members evaluates the level of respectfulness displayed by speakers towards members of their own party versus members of other parties, providing insights into the dynamics of inter-party relationships and cooperation. Relevance to topic evaluates the alignment of speakers' contributions with the main topics under discussion, ensuring that the debate remains focused and on-topic. Tone measures the overall tone of the debate, including the level of hostility, aggression, or politeness displayed by speakers. These additional criteria provide a more complete assessment of the quality of discourse in legislative debates, capturing nuances that may not be fully captured by the original DQI framework. The complete set of DQI+ annotation criteria can be found in Appendix B.2.

DQI+ Annotation Prompt In constructing the prompt for the DQI+ annotation process (P_{DQI} from Algorithm 1), I build on the approach employed by Bosley (2024) to guide the LLM in evaluating the discourse quality of the debate based on the six criteria. In that study, the author took advantage of pre-existing annotations in order to provide a set of examples that could be used to train the model to recognize the features of high and low quality discourse. Here, I take a different approach in line with the synthetic example generation process used for the graph generation prompt, using synthetic examples generated in collaboration with the LLM to illustrate the evaluation process and provide clear guidelines for scoring each criterion. In addition, that approach was designed for the annotation of a single speech at a time, whereas the approach employed here is designed to sequentially annotate a list of speeches from a debate in a single pass, allowing the model to build a cumulative understanding of the debate as it progresses.

The final prompt combines a detailed explanation of the DQI+ criteria, specific questions and guidelines for scoring each criterion, synthetic examples of a range of different speeches that illustrate a range of different annotation scores across the six criteria, along with explanations for the annotation decisions, the structured representation generated by the LLM in the graph generation step, and instructions to sequentially annotate the speeches corresponding to the "edges" between speakers and arguments in the graph representation.

The end result is a set of annotations that are structured in a way that can be easily integrated into the graph representation, providing even richer detail of the discourse quality in the debate.

DQI+ Annotation Output

Speaker	**Speech ID**	**Part.**	**Justification**		**Respect**					**Constr.**	**Relev.**	**Tone**
			Level	Content	Groups	Demand	Counter	Same-Party	Other-Party			
Lambert (PC)	2515912	1	3	2	1	3	1	-	1	2	4	2
Matheson (Liberal)	2515915	1	3	2	2	3	3	-	3	3	4	3
Matheson (Liberal)	2515922	1	3	3	2	3	2	-	2	3	4	3
Matheson (Liberal)	2515924	1	-	-	-	2	-	-	-	1	2	1

Table 3.1: Speech-Level Annotations for the Debate on Economic Mismanagement

The DQI+ annotations are generated in the following format, which includes detailed evaluations of each criterion for each speaker, along with an overall assessment of the discourse quality in the debate. As discussed above, each annotation is a numeric score on an ordinal scale, with higher scores indicating higher quality discourse. Table 3.1 provides a summary of the DQI+ annotations for the example debate on economic mismanagement.

```
## DQI Annotations

### 1. Marcel Joseph Lambert (
   Progressive Conservative)
- **Participation:** 1
  - *Reasoning:* Presents argument
     without interruption. "I caution,
      hon. members and Mr. Speaker,
     that most decisions of government
      ought not to be taken at
     dominion-provincial conferences."
- **Justification:**
  - **Level:** 3
    - *Reasoning:* Provides multiple
       reasons: examples of
       mismanagement from Britain and
       the need for increased
       productivity in Canada. "Our
       tendency, unfortunately, has
       been to follow Britain down
       that road."
  - **Content:** 2
    - *Reasoning:* Refers to broader
       societal concerns regarding
       productivity and economic
       management.
- **Respect:**
  - **Toward Groups:** 1
    - *Reasoning:* Mentions the general
        population neutrally.
  - **Toward Main Demand:** 3
    - *Reasoning:* Fully supports
       economic reform and improvement
       .
  - **Toward Main Counterarguments:**
     1
    - *Reasoning:* Acknowledges
       opposing views but does not
       engage substantively.
  - **Toward Other-Party Members:** 1
    - *Reasoning:* Criticizes other
       members' economic understanding
        without direct engagement.
- **Constructive Politics:** 2
  - *Reasoning:* Proposes productivity
      increases but does not address
     specific opposition concerns.
- **Relevance to Topic:** 4
  - *Reasoning:* Directly addresses
     economic mismanagement.
- **Tone:** 2
  - *Reasoning:* Critical yet measured
      tone, focusing on the need for
     reform.

### 2. John Ross Matheson (Liberal)
- **Participation:** 1
  - *Reasoning:* Presents argument
     fully without interruption. "Mr.
     Speaker, in joining this debate,
     I shall try to meet the
     challenges posed by the amendment
     ..."
- **Justification:**
  - **Level:** 3
    - *Reasoning:* Multiple reasons
       provided: defends the Minister
       of Finance, highlights economic
        successes. "He is attacking
       the most professionally
       competent Minister of Finance
       we have had in Canada's history
```

```
      ."
  - **Content:** 2
    - *Reasoning:* Focuses on broader
      societal benefits, emphasizing
      stability and competence.
- **Respect:**
  - **Toward Groups:** 2
    - *Reasoning:* Positive view of
      Canadian citizens benefiting
      from economic policies.
  - **Toward Main Demand:** 3
    - *Reasoning:* Fully supports
      government's economic
      management.
  - **Toward Main Counterarguments:**
    3
    - *Reasoning:* Acknowledges
      opposition's claims and
      critiques their validity.
  - **Toward Other-Party Members:** 3
    - *Reasoning:* Engages with
      opposition respectfully while
      defending government actions.
- **Constructive Politics:** 3
  - *Reasoning:* Seeks common ground
    by addressing successes while
    countering criticisms.
- **Relevance to Topic:** 4
  - *Reasoning:* Directly relevant to
    economic management and
    mismanagement.
- **Tone:** 3
  - *Reasoning:* Constructive and
    respectful tone, focusing on
    collaboration.

[...] {Additional annotations} [...]
```

3.3.5 Relational Database Integration

Once the graph representation and DQI+ annotations have been generated for a debate, the next step is to integrate this structured data into a relational database that can be queried and analyzed using standard database tools. To achieve this, I use the Neo4j graph database (Neo4j, 2022), which is well-suited for storing and querying graph data.[6] The graph representation generated by the LLM is translated into a set of nodes and relationships in the Neo4j database. The database structure includes key elements such as nodes that represent entities like debates, speakers, speeches, arguments, subjects, topics, sessions, and parliaments. Relationships capture connections between these nodes, including "GIVES" (between a speaker and a speech), "MAKES" (between a speech and an argument), "ADDRESSES" (between an argument and a subject), and "SUPPORTS" or "CONTRADICTS" (between arguments). Additionally, properties store metadata about nodes, with DQI+ annotations embedded as properties of speech nodes.[7]

This structure facilitates flexible and efficient querying of the debate data, enabling a

[6] I experimented with other database systems, including SQL databases and document-oriented databases, but found that Neo4j provided the most flexibility and performance for storing and querying the structured debate data, as well as for visualizing the nework structure of the debates.

[7] To populate the Neo4j database, I wrote a series of scripts that parse the structured graph representation and DQI+ annotations generated by the LLM and insert the data into the database. I used the Python library "py2neo" to interact with the Neo4j database and insert nodes and relationships.

wide range of potential analyses. As a demonstration of the capabilities of the database, I provide several example queries in Appendix B.3 that illustrate how researchers can extract insights from the structured debate data. These queries include tracking Discourse Quality Index (DQI) trends over time, analyzing inter-party argument dynamics, studying the evolution of topics, measuring argument complexity and interconnectedness, and calculating a polarization index based on the relationships between arguments from different parties. In Section 3.4, I implement these queries on a dataset of legislative debates spanning over a century from the Canadian parliament.

3.3.6 Validation

Because of the complexity of the multi-step LLM annotation approach detailed above, it is essential to validate the quality and reliability of the generated graph representations and DQI+ annotations. In this paper, I employ a random spot-checking approach, where I manually review a random sample of generated graphs and DQI+ annotations to assess their accuracy and coherence when compared to the debates they are derived from. This is admittedly a preliminary validation method that is intended to assess the face validity of the generated data. In future work, I plan to conduct more systematic validation studies, including inter-coder reliability tests, expert reviews, and comparison with human-coded data, to provide a more comprehensive evaluation of the LegisGraph approach.

3.4 Applications

In this section, I demonstrate the utility of the LegisGraph approach by applying it to a dataset of legislative debates from the Canadian parliament. I use the approach outlined in Section 3.3 to generate structured graph representations and DQI+ annotations for a set of debates, and integrate this data into a Neo4j graph database. I then conduct a series of analyses using the database to extract insights into the dynamics of legislative discourse, including trends in discourse quality, inter-party argument dynamics, topic evolution, argument complexity, and polarization. These analyses showcase the power of the LegisGraph approach to provide rich, structured data that can be used answer a wide range of research questions that would be difficult to address using existing methods, quantitative or qualitative.

3.4.1 Data Description

The dataset used in this analysis (retrieved from the LiPaD project[8])) consists of a collection of legislative debates from the Canadian parliament spanning over a century, from 1900 to 2020. The dataset includes transcripts of debates from the House of Commons, covering a wide range of topics and policy issues. The data is organized at the level of individual speeches, with each speech representing a statement made by a member of parliament during a debate. The debates are organized by parliamentary session, with each debate containing multiple speeches by different speakers, representing various parties and positions. The assignment of speeches to debates is based on the debate-level organization from the original Hansard transcripts, which group speeches by debate.

I selected a representative sample of 1,000 debates from the dataset, spanning three different parliaments: the 12th (1912-1917), 24th (1958-1962), and 34th (1988-1993) parliaments. I chose these parliaments to capture a diverse range of historical periods and policy contexts, with the goal of illustrating the ability of the LegisGraph approach to analyze debates across different time periods and political contexts. The sample includes debates from different sessions within each parliament, ensuring a diverse representation of debates over time. The breakdown of debates by parliament and session is shown in Table 3.2.

To create a representative sample of debates from the three parliaments (12th, 24th, and 34th), I employed a stratified sampling approach designed to account for the varying numbers of debates across different sessions within each parliament. This method ensures that each session is fairly represented according to its relative contribution to the total number of debates within its parliament.

For each parliament, I calculated the total number of debates across all its sessions. The weight for each session was determined by the following formula:

$$w_{i,j} = \frac{d_{i,j}}{\sum_{j=1}^{n_i} d_{i,j}}$$

where $w_{i,j}$ is the weight for session j in parliament i, $d_{i,j}$ is the number of debates in session j of parliament i, n_i is the total number of sessions in parliament i.

Using the calculated weights, I randomly sampled debates from each session, ensuring that sessions with more debates had a higher likelihood of contributing to the sample, while still giving a chance for sessions with fewer debates to be included. The sampling process was repeated across all three parliaments until a total of 1,000 debates were selected.

[8]See https://lipad.ca.

Table 3.2: Descriptive Statistics of Parliamentary Debates by Parliament and Session

Parliament	Session	Total Debates	Total Speeches	Total Words	Start Date	End Date
12	1	50	768	101,170	1912-01-26	1912-03-19
12	2	55	1,263	154,123	1913-04-25	1913-06-06
12	3	107	3,707	444,173	1914-01-22	1914-06-11
12	4	2	30	9,065	1914-08-20	1914-08-21
12	5	45	1,745	206,762	1915-02-18	1915-04-15
12	6	96	2,667	406,052	1916-01-24	1916-05-17
12	7	128	4,006	482,371	1917-01-25	1917-09-18
24	1	32	569	100,988	1958-05-13	1958-09-06
24	2	56	1,325	141,712	1959-01-30	1959-07-17
24	3	72	1,349	149,367	1960-01-19	1960-08-09
24	4	72	1,229	180,624	1960-11-22	1961-09-21
24	5	23	388	36,605	1962-01-26	1962-04-17
34	1	3	48	10,739	1988-12-12	1988-12-21
34	2	133	1,733	357,915	1989-04-04	1991-04-11
34	3	126	1,287	329,573	1991-05-22	1993-06-04

3.4.2 Applying the LegisGraph Approach

I apply the LegisGraph approach to the Canadian parliamentary debates dataset by generating structured graph representations and DQI+ annotations for a subset of debates. I employ the GPT-4o-mini model[9]from OpenAI's GPT-4 series to process the debate transcripts and generate the graph representations using the graph generation prompt (P_{Graph}), and the DQI+ annotation prompt (P_{DQI}) to evaluate the discourse quality of the debates based on the six criteria. Examples of complete annotated prompts for the graph generation and DQI+ annotation are provided in Appendices B.1 and B.2, respectively. The resulting output is integrated into a Neo4j graph database, which allows for flexible querying and analysis of the structured debate data.

3.4.3 Mapping the Distribution of Topics and Subjects

A key advantage of the LegisGraph approach is the ability to automatically extract and analyze the topics, subjects, and arguments discussed across a large corpus of debates. By leveraging the power of large language models to identify these elements and represent them in a graph structure, we can efficiently characterize what is being discussed and how different ideas connect and flow throughout the debates. Figure 3.2 shows the distribution of high-level topics in the Canadian parliamentary debates dataset across the three parliaments in the sample. By inspecting this figure we can see, for example, that the Labor and Employment topic (encoded as the red color) was frequently discussed in all three parliaments, but with varying levels of prominence. In addition, we can see that the Social Policy topic (encoded as the purple color) has become increasingly prevalent over time, indicating a shift in policy priorities towards social issues.

If we want to know more about how the topic of social policy was discussed, it is straightforward to query the database to retrieve the most frequently discussed subjects within the Social Policy topic. Figure 3.3 shows the top 10 most frequently discussed subjects within the Social Policy topic across all three parliaments in the sample. This analysis reveals that Pensions, Healthcare, and Education were the most prominent subjects within the Social

[9]For this task, I use the GPT-4o-mini model, which is a smaller, cost-effective version of the GPT-4 model, known for its robust performance in generating coherent and contextually relevant text. The choice of GPT-4o-mini is informed by several factor, including the ability to handle the complexity of the graph generation task, and the cost and of accessing it through OpenAI's API. GPT-4o-mini costs approximately $0.15 per million input tokens, and $0.60 per million output tokens, and is able to generate high-quality text with a maximum output of up to 16000 tokens, and with a context size of up to 128,000 tokens. All together, GPT-4o-mini strikes a balance between cost and performance, making it a suitable choice for the LegisGraph approach. In future work, I plan to explore the use of other open-source models, such as Meta's Llama series or DeepSeek.

Policy topic, providing insights into the specific policy areas that were the focus of legislative debates. In the 12th parliament, which took place during the onset of World War I, the subject of Canada's involvement in the war was particularly salient. We can see that in the 24th parliament, the subject of healthcare was a major focus, reflecting the changing social and economic landscape of the late 1950s and early 1960s, while in the 34th parliament, the subject of taxation, and particularly the Goods and Services Tax (GST), was a central topic of debate.

This represents a preliminary demonstration of the types of data exploration and analysis that can be conducted using the graph database generated by the LegisGraph approach. If we were interested in digging deeper, and exploring the debate over the GST in more detail, we could query the database to retrieve all speeches that address the subject of taxation and the GST, and analyze the arguments made by different speakers to gain a more nuanced understanding of the debate dynamics. Or, if we wanted to track the evolution of the discussion around healthcare policy over time, we could analyze the distribution of speeches and arguments related to healthcare in each parliament to identify trends and patterns in the discourse.

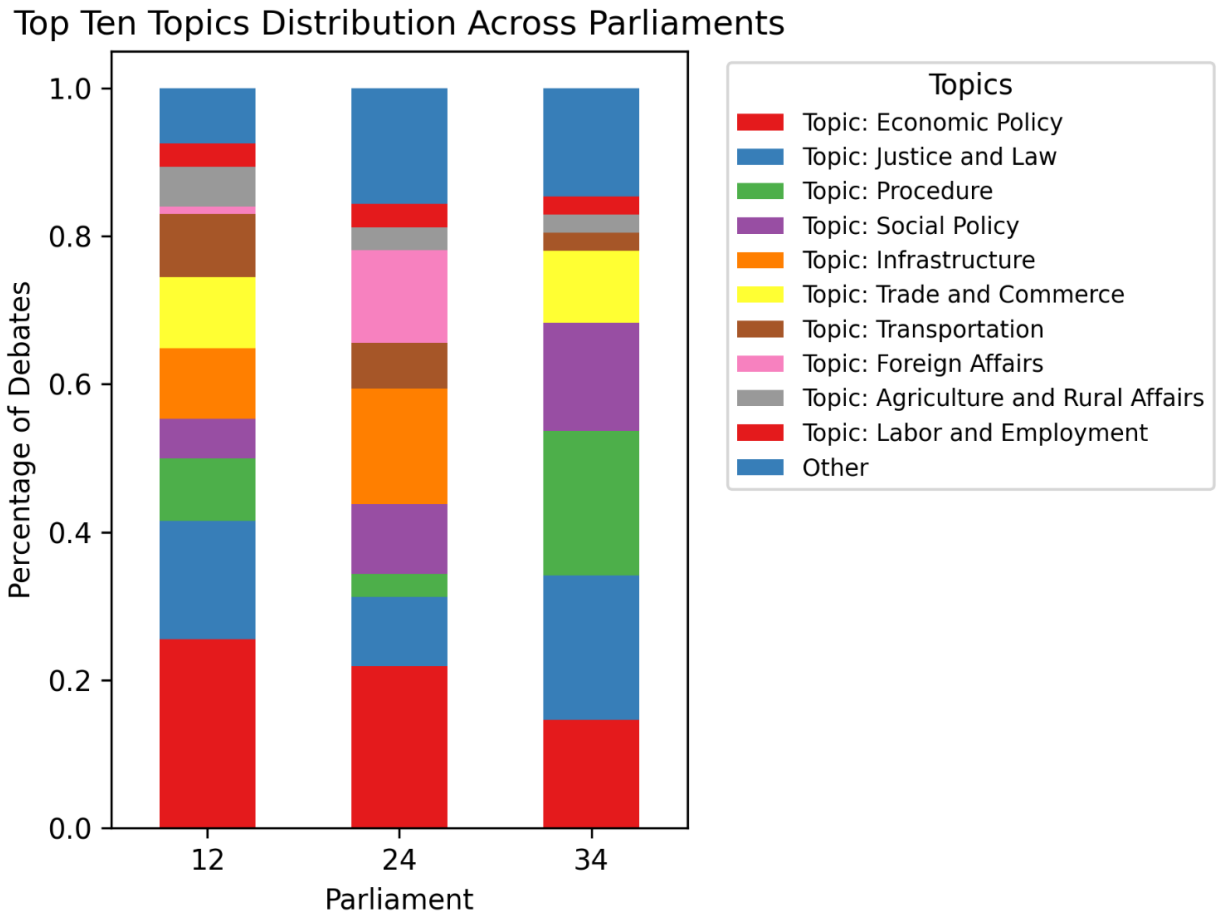


Figure 3.2: **Distribution of High-Level Topics in Canadian Parliamentary Debates.** This plot shows the relative frequency of each high-level topics (such as Economic Policy, Social Policy, Procedure, etc) across the three parliaments in the sample.

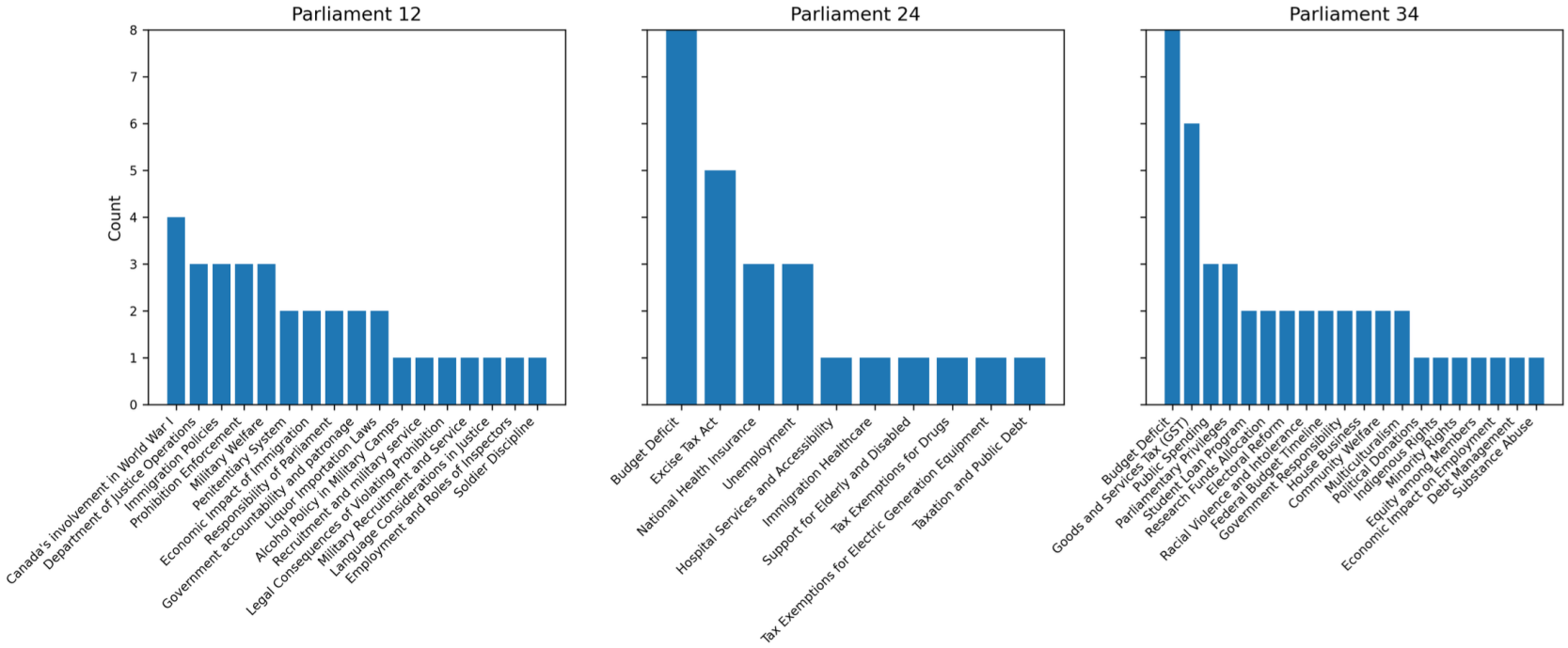


Figure 3.3: **Most Frequently Discussed Subjects in Social Policy Debates.** This plot shows the most frequently discussed subjects within the Social Policy topic across all three parliaments in the sample.

3.4.4 Mapping the Distribution of Discourse Quality

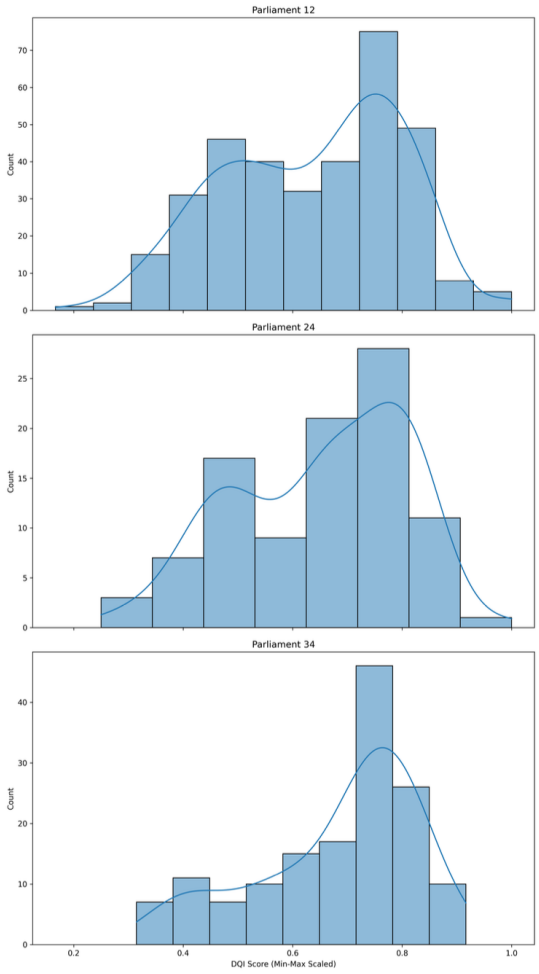
In addition to analyzing the distribution of topics and subjects in the debates, the LegisGraph approach enables us to assess the quality of discourse based on the DQI+ annotations. By leveraging the structured graph representations and DQI+ annotations generated by the LLM, we can analyze trends in discourse quality over time and by covariates such as speaker, party, and debate type, providing insights into the dynamics of legislative discourse and the factors that influence the quality of debate.

In order to more easily compare the quality of discourse across different debates, I first peformed a min-max normalization of the DQI scores for each speech, rescaling the scores to a range of 0 to 1. Then, I calculated the average DQI score for each debate by taking the mean of the normalized DQI scores for all speeches in the debate. This normalization allows us to compare the relative quality of discourse across different debates, providing a standardized measure of discourse quality that can be used to identify patterns and trends in the data.

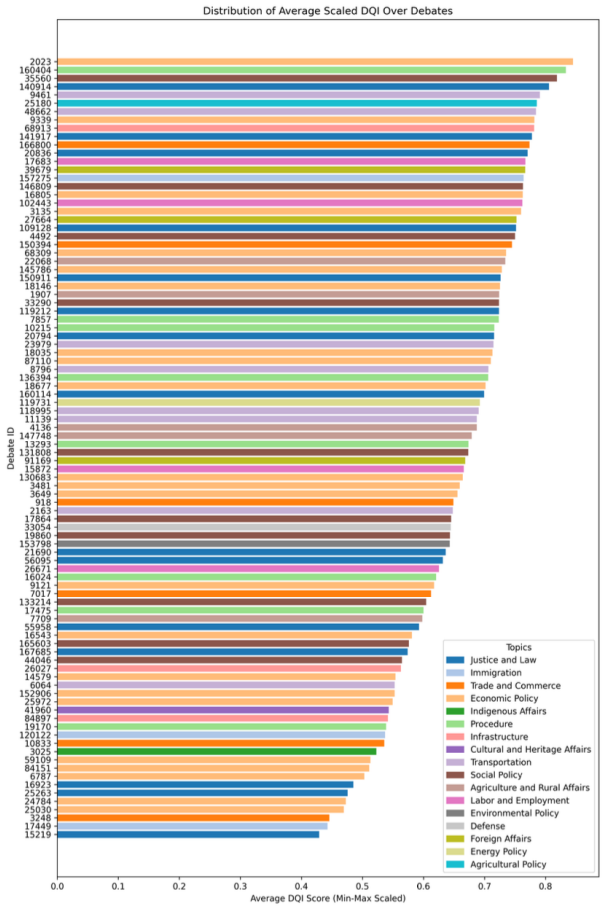
Figure 3.4 shows the distribution of the normalized and aggregated DQI scores across all speeches in the three sampled parliaments, and Figure **??** shows the distribution of DQI scores by debate, pooled across all three parliaments in the sample. These plots provide a high-level overview of the quality of discourse in the Canadian parliamentary debates dataset, allowing us to identify patterns and trends in discourse quality over time. From the histogram plots, for example, we can see that while the modal speech had an aggregate DQI score of around 0.7 in each parliament, there was significant variability in the quality of discourse, with some speeches scoring as low as 0.2 and others as high as 0.9 or even 1.0. This suggests that there is a wide range of discourse quality in the dataset, with some debates characterized by high-quality, substantive discourse, and others by lower-quality, more adversarial debate. We can see the same variability at the level of debate: the debates with the highest deliberative quality had average DQI scores around 0.8, while the debates with the lowest quality had average scores around 0.5.

When sampling from the low and high DQI debates, we can see that the high-quality debates were characterized by more substantive, respectful, and constructive discourse, while the low-quality debates were marked by more adversarial, less respectful, and less constructive dialogue. For example, the reported summary of analysis by the LLM annotator for Debate 35560, the third highest DQI debate in the sample, reads as follows:

> The debate on the Department of National Health and Welfare showcased a range of arguments focusing on healthcare provision, social security, and the importance of maintaining essential services for vulnerable populations. Participation was



(a) Distribution of DQI Scores Across All Speeches



(b) Distribution of DQI Scores by Debate

Figure 3.4: **Discourse Quality Index (DQI) Analysis**. (a) shows the distribution of DQI scores across all speeches in the three sampled parliaments, while (b) illustrates the variability of DQI scores per debate.

> consistent, with all speakers effectively delivering their messages without interruption. Justifications for arguments varied but were generally well-articulated, showing an understanding of broader societal implications.
>
> Respect towards affected groups and the main demands was notably present, particularly in the speeches advocating for improved healthcare and social services. Constructive politics improved as the debate progressed, with speakers proposing alternatives and acknowledging the contributions of previous administrations.
>
> Overall, the tone of the debate was mostly cordial and constructive, with speakers emphasizing collaboration and the necessity for improvements in health and welfare policy. The focus on vulnerable populations highlighted a collective concern for Canadian citizens' well-being, reflecting a responsible approach to governance in the context of national health policies.

In contrast, the summary of analysis for Debate 15219, the lowest DQI debate in the sample, reads as follows:

> This debate revolves around the accuracy of statements related to hay contracts and the implications of parliamentary conduct. Martin Burrell argues that Frank Carvell made misstatements regarding a hay contract, emphasizing the need for factual accuracy in parliamentary discourse. Carvell counters by asserting his readiness to take legal action against Burrell's claims. The discussion highlights tensions around freedom of speech within the context of parliament and the seriousness of accusations made between members.
>
> Overall, the debate is characterized by low levels of constructive engagement, high confrontational tones, and minimal respect for opposing views. The focus remains largely on personal attacks rather than the substantive issues at hand, indicating a low quality of discourse in this parliamentary setting.

While more work is needed to robustly validate the DQI+ annotations generated by the LLM, these examples provide a preliminary indication of the potential of the LegisGraph approach to capture and analyze the quality of legislative discourse in a systematic manner.

In addition to analyzing the distribution of discourse quality across all speeches and debates, we can also examine how the quality of discourse varies by party. Figure 3.5 shows the average DQI scores for each party across the three parliaments in the sample. There are substantial differences in the average quality of discourse between parties, with the Co-operative Commonwealth Federation (CCF) party scoring the highest on average in the 12th and 24th parliaments, and its successor the New Democratic Party (NDP) scoring at a

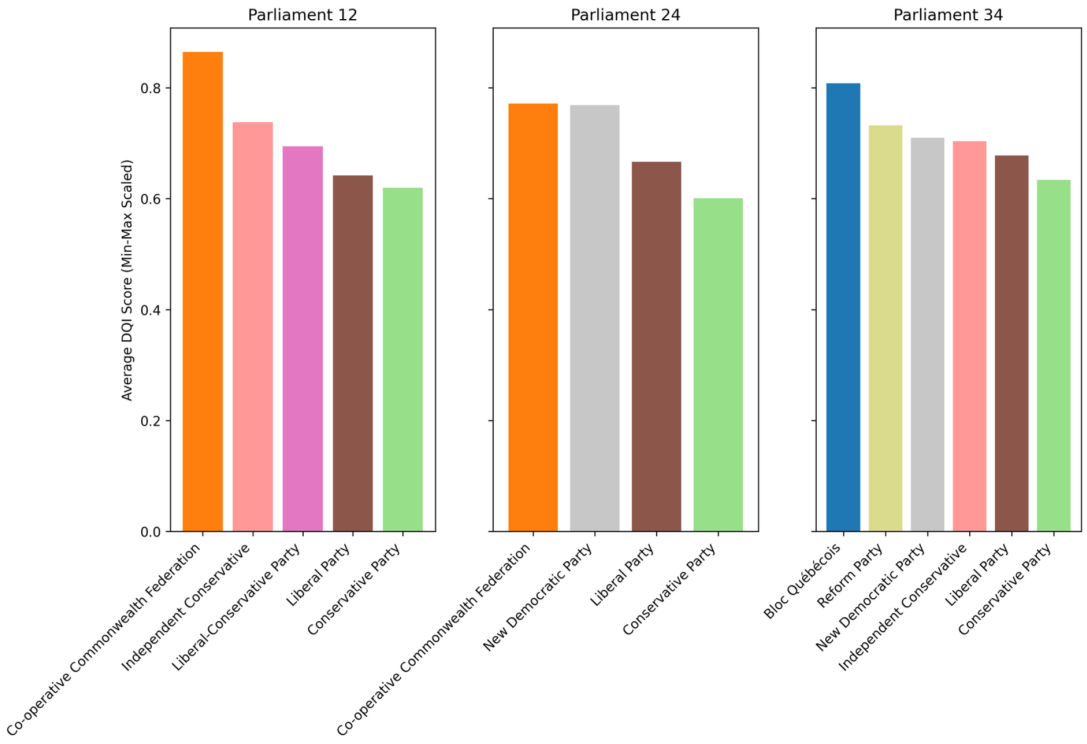


Figure 3.5: **Discourse Quality Index (DQI) by Party**. This plot shows the average DQI scores for each party across the three parliaments in the sample.

slightly lower level in the 34th parliament. The Conservative Party, in contrast, scored the lowest on average in all three parliaments.

As before, the graph database generated by the LegisGraph approach allows us to dig deeper into these patterns and explore the factors that drive differences in discourse quality between parties. For example, we could analyze the distribution of DQI scores by speaker within each party to identify patterns in the quality of discourse among individual members, or we could examine the relationship between party affiliation and discourse quality while controlling for other covariates such as debate topic or session.

3.4.5 Polarization Within and Between Parties

Finally, I show how we can leverage the structured graph representations generated by the LegisGraph approach to analyze the dynamics of political polarization within and between parties in the Canadian parliament. By extracting the relationships between arguments made by different speakers and parties, we can identify patterns of agreement and disagreement, track the formation of coalitions based on argument support, and calculate a global measure

of polarization based on the structure of the party-argument network, as well as the internal and external cohesion of each party.

Figures 3.6, 3.7, and 3.8 show the speaker-argument networks for each parliament in the sample, respectively, highlighting the clustering of speakers based on argument support. The size of the nodes represents the centrality of the speaker in the network, while the color indicates the party affiliation. The color of the edge indicates the nature of the relationship between arguments (support or contradiction), with e.g. a thick red edge indicating a strong contradiction and a thin blue edge indicating a weak support.

By visually inpecting these networks, we can identify patterns of polarization within and between parties, as well as the formation of coalitions based on argument support. For example, in the 12th parliament, we see that while there is some clustering of speakers based on party affiliation, there are also cross-party connections, both in terms of support and contradiction, suggesting that patterns of agreement and disagreement are more flexible than voting patterns, which are notoriously rigid in parliamentary systems such as Canada's. In the 24th parliament, we see a pattern of Liberal and NDP speakers supporting eachother more frequently and contradicting Conservative speakers, whereas the Conservative speakers frequently contradict eachother. This pattern essentially holds into the 34th parliament as well.

In an attempt to formalize these observations, I calculate a measure of polarization based on the ratio of arguments that contradict versus support arguments made by speakers from other parties, as well as the ratio of arguments that support versus contradict arguments made by speakers from the same party. Table 3.3 shows the calculated polarization and cohesion scores for each parliament in the sample, as well as the cohesion scores for each party. The polarization score is calculated as the ratio of arguments that contradict versus support arguments made by speakers from other parties, while the cohesion scores are calculated as the ratio of arguments that support versus contradict arguments made by speakers from the same party. However, because of the small sample size (this version of the draft uses a preliminary subsample of 100 debates), these results should be interpreted with caution.

It is once again worth noting that the LegisGraph approach offers a simple and powerful way to dig further into these patterns of polarization and cohesion, by allowing us to query the database for specific patterns of argument support and contradiction, track the evolution of these patterns over time, and identify the key actors and parties driving these dynamics. By combining the structured graph representations and DQI+ annotations generated by the LegisGraph approach with network analysis and machine learning techniques, we can gain a more comprehensive understanding of the complex dynamics of political communication and decision-making in the Canadian parliament.

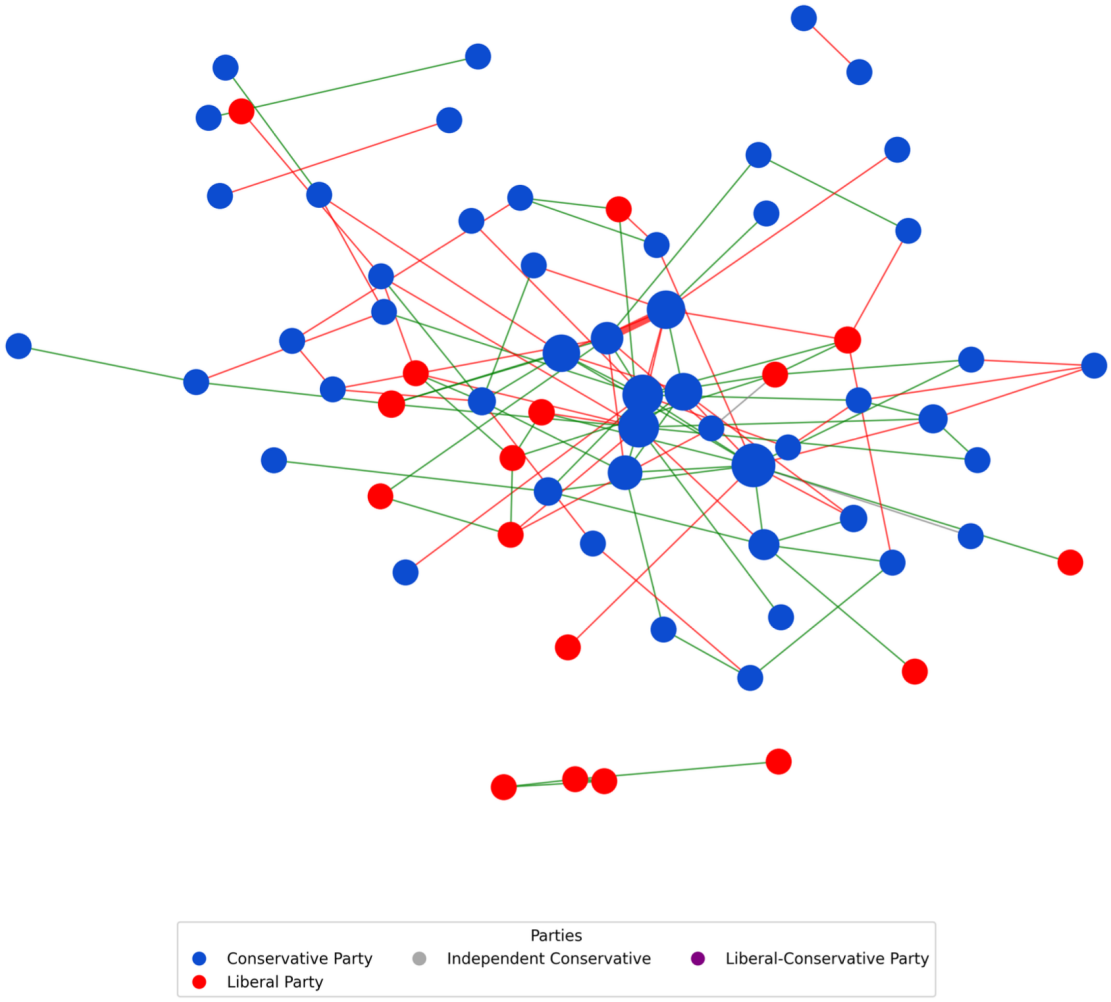


Figure 3.6: Speaker-Argument Network for Parl. 12

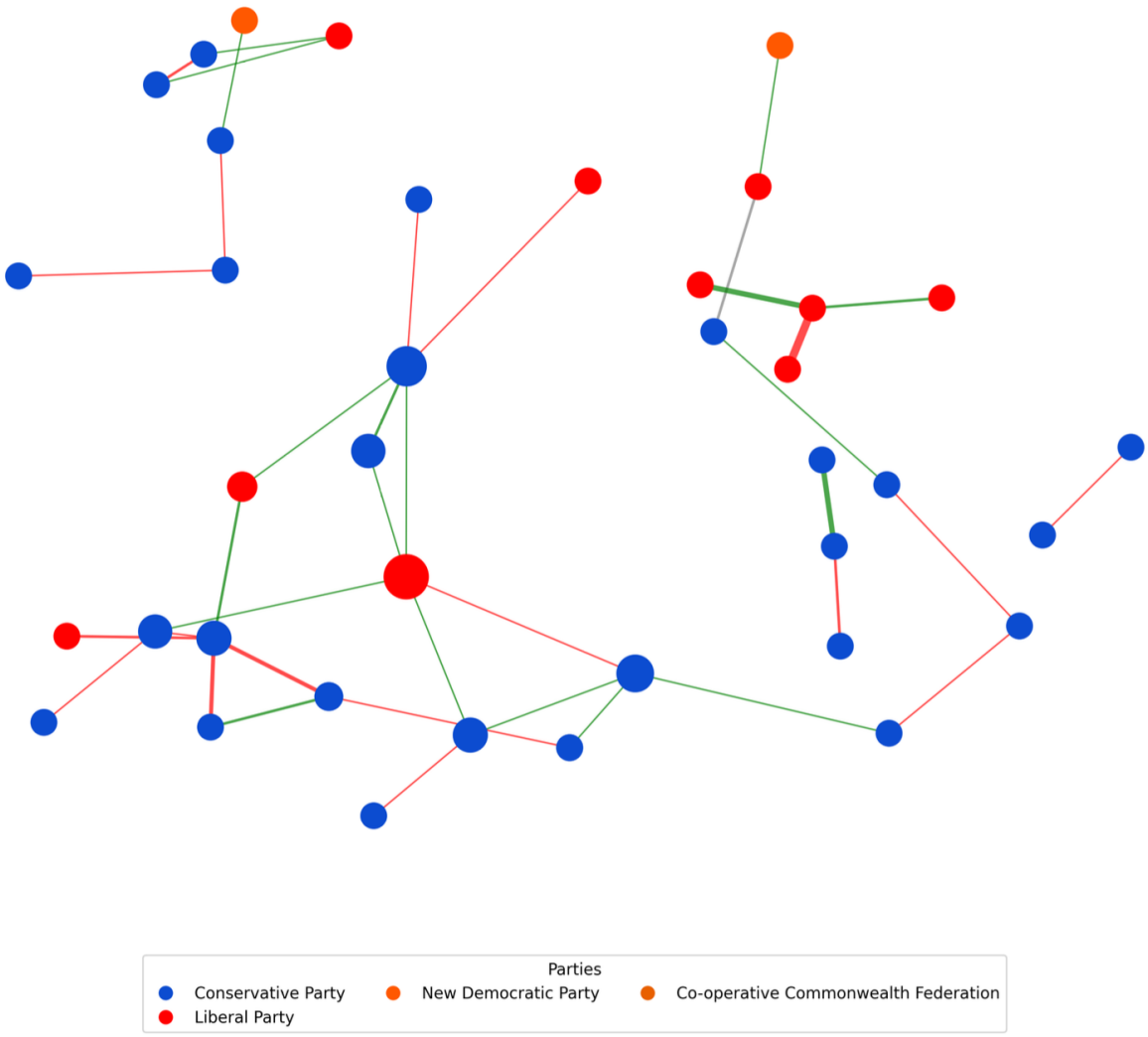


Figure 3.7: Speaker-Argument Network for Parl. 24

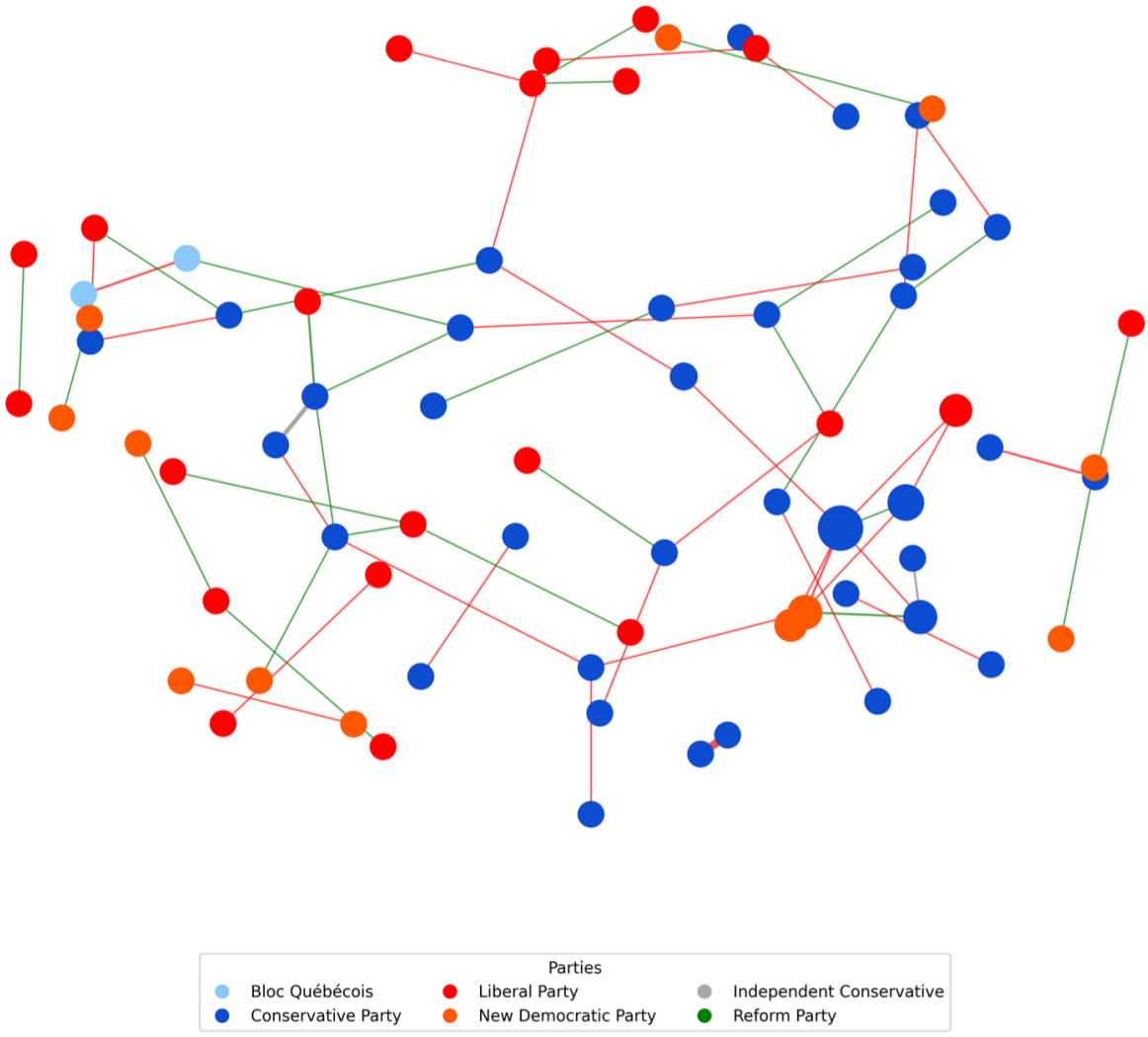


Figure 3.8: Speaker-Argument Network for Parl. 34

Parl.	Polarization	Party Cohesion							
	Score	Lib	Cons	Lib-Con	Ind. Con	CCF	NDP	Reform	Bloc
12	0.486	0.241	0.105	0.000	0.000	—	—	—	—
24	0.532	0.159	0.000	—	—	0.000	0.000	—	—
34	0.558	0.156	0.013	—	0.000	—	—	0.120	0.000

Table 3.3: Cohesion and Polarization Scores Across Selected Parliaments

3.5 Discussion

In Section 3.4, I have provided a demonstration of the utility of the LegisGraph approach by applying it to a dataset of Canadian parliamentary debates and conducting a series of analyses to extract insights into the dynamics of legislative discourse. The analyses showcased the power of the LegisGraph approach to provide rich, structured data that can be used to answer a wide range of research questions that would be difficult to address using existing methods, quantitative or qualitative. By leveraging the structured graph representations and DQI+ annotations generated by the LegisGraph approach, I have claimed that researchers can gain a deeper understanding of the complex dynamics of political communication and decision-making in the Canadian parliament.

The approach I articulate – from graph generation to DQI+ annotation to database integration – provides a systematic and scalable method for extracting structured data from unstructured text data, enabling rich insights into the dynamics of legislative discourse. By applying the LegisGraph approach to a dataset of Canadian parliamentary debates, I demonstrate the utility of the approach in generating structured graph representations and DQI+ annotations, and conducting a series of analyses to extract insights into the dynamics of legislative discourse.

However, more work is needed to validate the LegisGraph approach and address potential limitations. While the random spot-checking method provides a preliminary assessment of the quality and reliability of the generated data, more systematic validation studies are needed to ensure the accuracy and coherence of the structured data. Future work should include inter-coder reliability tests and a systematic comparison with human-coded data to provide a more comprehensive evaluation of the LegisGraph approach. Additionally, the more fundamental limitations of relying on LLMs, such as the closed-source nature of most models, the potential for hallucinations or biases, and the computational requirements of large-scale analysis, should be addressed through strategies for mitigating these risks, such as prompt refinement, human oversight, and optimization of performance and resource usage.

Despite these limitations, the LegisGraph approach has significant implications for legislative studies, offering the potential to uncover new insights into legislative behavior and discourse, enable comparative and longitudinal analysis at scale, and complement and extend existing methods, such as text scaling and topic modeling. The LegisGraph approach can be adapted for other types of political discourse, such as social media and news media, and applied to other deliberative contexts, such as legal proceedings and public consultations. By integrating the LegisGraph approach with other computational social science methods, such as network analysis and machine learning, researchers can gain a more comprehensive

understanding of the complex dynamics of political communication and decision-making.

3.6 Conclusion

In this paper, I have introduced the LegisGraph approach, a novel method for analyzing legislative debates using large language models to generate structured graph representations and DQI+ annotations. The LegisGraph approach combines the power of large language models with structured data generation prompts to extract key elements from legislative debates, including speakers, arguments, topics, and procedural elements, and to evaluate the quality of discourse based on a set of predefined criteria. The structured graph representations and DQI+ annotations generated by the LegisGraph approach provide rich, structured data that can be used to analyze the dynamics of legislative discourse, including trends in discourse quality, inter-party argument dynamics, topic evolution, argument complexity, and polarization.

Using this approach, I demonstrated the utility of the LegisGraph approach by applying it to a dataset of legislative debates from the Canadian parliament, generating structured graph representations and DQI+ annotations for a subset of debates, and integrating this data into a Neo4j graph database. I have conducted a series of analyses using the database to extract insights into the dynamics of legislative discourse, including trends in discourse quality, inter-party argument dynamics, topic evolution, argument complexity, and polarization. These analyses showcase the power of the LegisGraph approach to provide rich, structured data that can be used to answer a wide range of research questions that would be difficult to address using existing methods, quantitative or qualitative.

I have also provided a general framework for generating structured representations of political discourse from unstructured text data using large language models, which can be applied to a wide range of political texts, such as speeches, policy documents, interviews, and news articles. The LegisGraph method is defined by the prompts used to guide the LLM in generating structured data, and the same prompt refinement process proposed here can be applied to other types of political texts to generate structured data for analysis.

In future work, I plan to further validate the LegisGraph approach by conducting more systematic validation studies, including inter-coder reliability tests, expert reviews, and comparison with human-coded data. I also plan to explore the use of other open-source models, such as Meta's Llama series or DeepSeek, to generate structured data from political texts. Additionally, I aim to apply the LegisGraph approach to a wider range of political texts, including speeches, policy documents, interviews, and news articles, to generate structured data for analysis. By expanding the scope of the LegisGraph approach and refining the

prompts used to guide the LLM, I hope to provide a powerful tool for analyzing political discourse and generating insights into the dynamics of political communication.

Lastly, I plan to make the code and data used in this study publicly available to facilitate further research and development in the field of computational political science. By sharing the tools and methods developed in this study, I hope to encourage collaboration and innovation in the analysis of political discourse and the study of legislative dynamics using large language models. As the capabilities of these models continue to improve, the potential for generating structured data from unstructured text data will only increase, opening up new possibilities for generating insights into the complex dynamics of political communication and decision-making at a scale and depth that was previously unimaginable.

CHAPTER 4

Improving Probabilistic Models in Text Classification via Active Learning

4.1 Introduction

Text classification—the act of measuring underlying concepts by categorizing sequences of text into two or more categories—is a fundamental task in social science research. In political science, researchers have used this approach to classify a wide variety of textual data, including legislative speeches (Peterson and Spirling, 2018; Motolinia, 2021), correspondences to administrative agencies (Lowande, 2018), public statements of politicians (Airoldi et al., 2007; Stewart and Zhukov, 2009), news articles (Boydstun, 2013), election manifestos (Catalinac, 2016), social media posts (King et al., 2017), religious speeches (Nielsen, 2017), and human rights text (Cordell et al., 2022; Greene et al., 2019).[1]

Because manually labeling a large number of documents to classify text is too costly, researchers are increasingly turning to machine learning and Natural Language Processing (NLP) methodologies to automate this task. For example, to investigate the relationship between internet access and state repression in Syria, Gohdes (2020) manually labeled 2,000 out of 65,274 documents in order to train a machine learning model to predict the class of the documents in the rest of the corpus. Similarly, Park et al. (2020) train a classifier using 4,000 of the 2,473,874 documents in their corpus to analyze the association between Information Communication Technologies (ICTs) and the U.S. Department of State's human rights reports. Although these approaches are more efficient than manually labeling an entire corpus, labeling thousands of documents still demands considerable time and effort, as the authors of these studies acknowledge.

To help researchers reduce the amount of labeled data required to train an accurate classification model, we introduce *activeText*, a fast and easy-to-use algorithm for text classifi-

[1]See Grimmer et al. (2022) for a further overview of the application of text classification methods in political science research.

cation that can be run on a standard laptop. Our method combines probabilistic modeling, semi-supervised learning (Nigam et al., 2000; Zhu and Goldberg, 2022) and active learning (McCallum et al., 1998; Settles, 2011; Miller et al., 2020) to help guide researchers to label documents more efficiently.

Recently, deep learning models like Bidirectional Encoder Representations from Transformers (BERT, Devlin et al. 2019a) have become popular due to the impressive performance on many text classification tasks (Liu et al., 2019). However, they require substantial computational resources to train and can be prone to overfitting when labeled data is scarce (Yue et al., 2021). In addition, it is difficult to understand how their predictions are generated (Rudin, 2019). Consequently, deep learning models may not be the best choice when researchers have limited computational resources and labeled data. We present *activeText* as an additional option for researchers who face these challenges.

The simple mixture model we use at the core of *activeText* runs an order of magnitude faster than deep learning models by employing the Expectation-Maximization (EM) algorithm (Dempster et al., 1977a) for parameter estimation. Because of the computational efficiency of our model, we embed it in an active learning framework where the most informative documents are selected for human labeling. As a result, users of *activeText* can rapidly iterate between training the model to generate estimates of label uncertainty and selecting informative documents for human labeling to train a classification model with high accuracy using only a small fraction of the documents in their corpus.

Another key feature of our approach is that the parameters of our mixture model can be easily inspected to understand how the model is making its predictions. We leverage this interpretability to allow users with existing subject expertise to boost classification performance by upweighting keywords associated with each classification category. All these features make of *activeText* an attractive option for social science researchers for whom labeled data and computational resources are scarce.

We demonstrate the performance of *activeText* in three ways. First, we conduct a series of validation experiments to assess our performance on four common political science classification tasks using real text corpora: identifying news articles as political, identifying toxic hate speech, classifying the topic of supreme court rulings, and identifying mentions of physical integrity violations in human rights reports. Our validation experiments show that when there are few labeled documents, *activeText* generally outperforms alternatives, including DistilBERT (Sanh et al., 2019), a more computationally efficient variant of BERT, in terms of classification performance. We show that this benefit is most pronounced when the corpus is unbalanced, and that upweighting keywords can boost further model performance. We also show that in contrast to models such as BERT, which require substantial

computational resources to train, *activeText* can easily be run on a standard laptop using the statistical programming language **R**, with the prediction of classification labels typically completing in seconds for a corpus with tens of thousands of documents.

Second, we replicate Gohdes (2020) and Park et al. (2020) using *activeText* to show how researchers could have used our method to reach the same substantive conclusions—a higher level of internet access is associated with a larger proportion of targeted killings, and ICTs are not associated with the sentiment of the State Department's human rights reports, respectively—with far fewer labeled documents.

Third, we use simulations to explore the general conditions under which *activeText* performs well, and to evaluate the impact of mislabeling documents and the potential biases introduced by active learning on the classification performance of *activeText*. We show that *activeText* is robust to minor instances of mislabeling and that the in-sample bias introduced by active learning does not affect out-of-sample classification performance.

This paper proceeds as follows. In "Machine Learning Approaches to Text Classification," we introduce readers to the concepts of semi-supervised and active learning approaches to text classification. In "The Method," we describe both the semi-supervised and the active learning components of *activeText*, and how we combine the two. In "Validation Performance," we show the results from comparing our model to popular alternatives on validation data sets. Then, "Reanalysis with Fewer Human Annotations" presents the results of our replication studies. Finally, we discuss several practical concerns, directions for future research, and possible improvements to the algorithm in "Discussion." Code and data to reproduce the analysis of this paper is available at the American Political Science Review Dataverse (Bosley et al., 2024).

4.2 Machine Learning Approaches to Text Classification

In social science research, it is common for researchers to want to classify a large collection of text documents into two or more categories based on the content of the text in order to test a substantive hypothesis. The biggest impediment to this process is the cost of manually labeling a large collection of text documents. Rather than exhaustively labeling all documents, researchers often use machine learning techniques to automate the process, with supervised learning being the most common paradigm. In supervised learning, a model is trained on a labeled dataset to learn the relationship between text features and class labels (Kotsiantis, 2007), and a variety of supervised learning algorithms, such as Naive Bayes, Support Vector Machine (SVM), and Logistic Regression, have been applied to text classification tasks in political science research (Hillard et al., 2008; Colleoni et al., 2014).

Even though supervised learning reduces the amount of labeling required relative to hand-coding all documents, it still requires a substantial amount of labeled data to train a model that generalizes well to the entire corpus, typically in the order of several thousand labeled documents. In most applied political science applications, however, researchers start with little to no labeled data, making it laborious to label sufficient data to train accurate classifiers. In this section, we discuss two machine learning approaches that we leverage in our method to address the challenge of labeled data scarcity: semi-supervised learning and active learning.[2]

4.2.1 Solutions to Labeled Data Scarcity

To address the challenge of labeled data scarcity, several approaches have been proposed in the machine learning literature, including semi-supervised and active learning. Semi-supervised learning aims to leverage the structure of large amounts of unlabeled data to improve classification performance (Miller and Uyar, 1996; Nigam et al., 2000). In a semi-supervised setting, the model learns from both labeled and unlabeled data, using the labeled data as a foundation for measurement and incorporating patterns recovered from the unlabeled data to produce more accurate and robust predictions. This approach is particularly useful when labeled data is scarce, but unlabeled data is abundant.

Active learning, on the other hand, focuses on strategically selecting the most informative instances for labeling, minimizing the labeling effort while maximizing the model's performance (Settles, 2011). One of the most studied active learning approaches is uncertainty sampling (Lewis and Gale, 1994; Yang et al., 2015), where documents are chosen for labeling based on how uncertain the model is about their correct classification. By focusing labeling efforts on these informative documents, active learning can learn the decision boundary more efficiently than randomly selecting documents for labeling. In addition, active learning approaches have been shown to be particularly effective when the classification categories are imbalanced, which is a common occurrence in social science classification exercises (Miller et al., 2020).

An active learning algorithm typically involves a sequence of iterative steps applicable to any classification methodology. The first step is to estimate the probability that each document belongs to a specific classification outcome. The second step involves actively selecting the documents that the model is most uncertain about and focusing manual labeling efforts among those documents (Hoi et al., 2006). Then, the class probabilities are re-

[2]For an introduction of basic concepts in machine learning applied to text data for classification tasks, including topics like feature representation, supervised and unsupervised learning, discriminative versus generative models, and model evaluation metrics, please refer to Supplemental Information (SI) **??**.

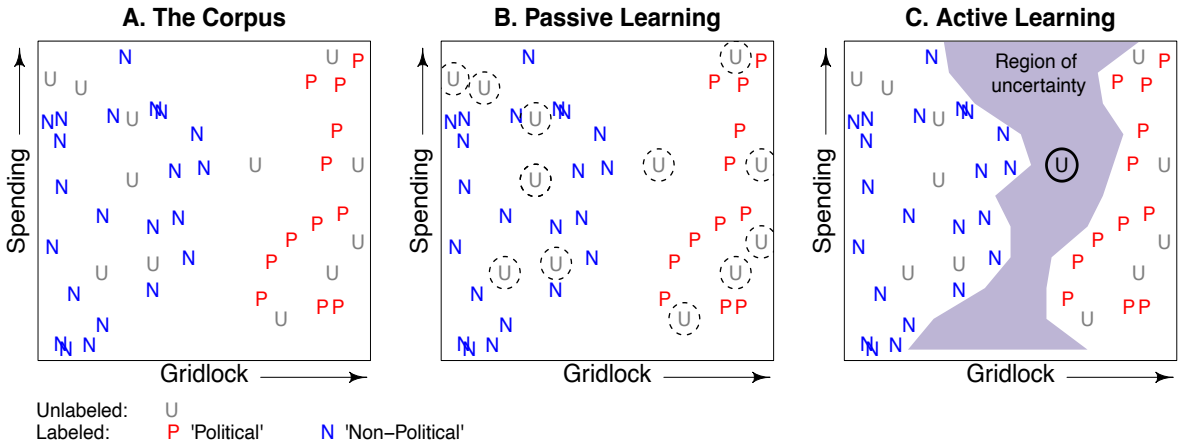


Figure 4.1: **Labeling: Passive vs Active Learning.** Panel A presents a corpus where a classifier based on term frequencies of Spending" and Gridlock" is utilized to categorize unlabeled (U) documents as political (P) and non-political (N). Panel B depicts a passive learning approach where the next document to be labeled is randomly selected. In contrast, Panel C demonstrates active learning, where obtaining the true label of the U located in the region of uncertainty for the classifier (shaded region) is prioritized, as it provides more informative insights into learning the decision boundary between P and N.

estimated using the newly labeled data. The algorithm cycles through these steps until a stopping criterion is met, such as a fixed budget for labeling (Ishibashi and Hino, 2020) or a threshold for improvement in accuracy metrics such as precision, recall, or F1 score (Altschuler and Bloodgood, 2019).

To illustrate the difference between passive and active learning for labeling a document, consider the scenario where a researcher aims to classify each unlabeled (U) document as either political (P) or non-political (N) based on the frequency of terms like Spending" and Gridlock" (Figure 4.1, Panel A presents the corpus). In passive learning (Panel B), the next document to be labeled is randomly selected, regardless of its position in the feature space. In contrast, active learning (Panel C) prioritizes labeling documents in the region of uncertainty (shaded region), where the model is less confident about their true labels. By focusing labeling efforts on these informative documents, active learning can learn the decision boundary more efficiently than passive learning

4.2.2 Deep Learning and *activeText*

In recent years, transfer learning, especially using deep learning approaches and pre-trained embeddings, has become popular. Transfer learning is a machine learning technique where knowledge gained from solving one problem is applied to a different but related problem,

often leading to improved performance and reduced training time compared to training from scratch (Ruder et al., 2019). This approach relies on leveraging large pre-trained models, such as BERT (Devlin et al., 2019a), which have been trained on vast amounts of unlabeled text data using the Transformer architecture (Vaswani et al., 2017) to learn rich, contextual representations of words and sentences. Rather than encode text as a sparse vector of word frequencies and learn the relationship between text features and class labels, as in the bag-of-words representation, these models learn embeddings—dense, vector representations of words that capture semantic and syntactic similarities between words and documents. Once the embedding representations of text data are learned, they can be fine-tuned towards a specific classification task using a collection of labeled data[3], allowing the model to adapt its learned representations to the specific domain and task at hand.

While transfer learning with deep learning models has been shown to excel at many text classification tasks (Devlin et al., 2019a; Liu et al., 2019), simpler models still have a place in the text classification toolkit, especially when labeled data and computing resources are scarce. Deep learning models require significant computational resources and can be time-consuming to train, even when fine-tuning pre-trained models (Strubell et al., 2019), and require substantial technical expertise in machine learning and natural language processing to implement relative to simpler models. They also have an extremely large number of parameters, making them more prone to overfitting when labeled data is limited (Yue et al., 2021).[4] In addition, their complex architectures and high-dimensional representations can make them difficult to interpret (Guidotti et al., 2018). For a model to be interpretable, we mean both that the model's predictions can be explained in terms of the input features and that the model's parameters can be used to gain insights into the underlying phenomena and test substantive theories, both of which are essential in political science research.[5]

Because of these limitations, we argue that when labeled data is scarce, computational resources are limited, and model interpretability is crucial—i.e., the conditions under which the typical political scientist operates—combining semi-supervised and active learning techniques with a simple mixture model[6] is a viable alternative to deep learning approaches at a fraction of the computational cost.[7]

[3]Fine-tuning involves adding a classification layer on top of the pre-trained model and training it on the target task while keeping the pre-trained model weights mostly fixed (Howard and Ruder, 2018).

[4]Overfitting occurs when a model learns to fit the noise or random fluctuations in the training data, rather than the underlying patterns, leading to poor performance on new, unseen data, and is a common problem in machine learning, particularly when the amount of labeled data is small and/or the model is complex (Hastie et al., 2009).

[5]See Rudin (2019) for a discussion of the importance of interpretability in machine learning.

[6] Mixture models are probabilistic models that can effectively capture the underlying structure of the data while remaining computationally efficient and interpretable (McLachlan et al., 2019).

[7] This is not to say that social scientists should not use deep learning models. To the contrary, we expect

In the following sections, we propose *activeText*, a novel method that combines semi-supervised learning and active learning with a generative mixture model based on bag-of-words representations of text data. Our approach leverages the EM algorithm to learn from both labeled and unlabeled data and incorporates uncertainty-based active learning to strategically select examples for labeling. We demonstrate the effectiveness of our approach through experiments and case studies on real-world political science datasets, highlighting its performance, interpretability, and computational efficiency compared to alternative methods.

4.3 The Method

In this section, we present our modeling strategy and describe our active learning algorithm. For the probabilistic model (a mixture model for discrete data) at the heart of the algorithm, we build on the work of Nigam et al. (2000), who show that probabilistic classifiers can be augmented by combining the information coming from labeled and unlabeled data. As we will discuss below, we insert our model into an active learning algorithm and use the EM algorithm to maximize the observed-data log-likelihood function and estimate the model parameters.

4.3.1 Model

Consider the task of classifying N documents as one of two classes (e.g., political vs. non-political). Let $\mathbf{D}$ be a $N \times V$ document feature matrix, where V is the number of features.[8] In most applications, features are words, but they can also be bi-grams, tri-grams, or other tokens such word embeddings, etc. We use $\mathbf{Z}$, a vector of length N, where each entry represents the class assigned to each document. If a document i is assigned to the kth class (out of K classes), then $Z_i = k$, where $k \in \{0, 1\}$ (e.g., $k = 1$ represents the class of documents about politics, and $k = 0$ those that are non-political). Because we use a semi-supervised approach, some documents are already hand-labeled. This means that the value of Z_i is known for the labeled documents and is unknown for unlabeled documents.

To facilitate exposition, we assume that the classification goal is binary, however, our approach can be extended to accommodate 1) multiclass classification, where $K > 2$ and each document is classified into one of the K classes e.g., classifying news articles into 3 classes: politics, business, and sports (see SI **??**); and 2) modeling more than two classes but

that in many cases, deep learning models will outperform simpler models, especially when labeled data is abundant and computational resources are not a constraint.

[8]Throughout the paper, we denote a row or a column of a matrix by using $\cdot$ in the subscript, where the subscript $\cdot a$ represents the ath column and the subscript $b\cdot$ represents the bth row.

keeping the final classification output binary (see SI **??**).[9]

Equations (4.1) to (4.7) summarize the model:

Labeled Data

$$Z_i = k \quad \rightarrow \quad \text{hand-coded}, \quad k \in \{0, 1\} \tag{4.1}$$

$$\boldsymbol{\eta}_{\cdot k} \overset{\text{i.i.d.}}{\sim} \text{Dirichlet}(\boldsymbol{\beta}_{\cdot k}) \tag{4.2}$$

$$\mathbf{D}_{i\cdot} | Z_i = k \overset{\text{i.i.d.}}{\sim} \text{Multinomial}(n_i, \boldsymbol{\eta}_{\cdot k}) \tag{4.3}$$

$\lambda \times$ Unlabeled Data

$$\pi \sim \text{Beta}(\alpha_0, \alpha_1) \tag{4.4}$$

$$Z_i = k \overset{\text{i.i.d.}}{\sim} \text{Bernoulli}(\pi), \quad k \in \{0, 1\} \tag{4.5}$$

$$\boldsymbol{\eta}_{\cdot k} \overset{\text{i.i.d.}}{\sim} \text{Dirichlet}(\boldsymbol{\beta}_{\cdot k}) \tag{4.6}$$

$$\mathbf{D}_{i\cdot} | Z_i = k \overset{\text{i.i.d.}}{\sim} \text{Multinomial}(n_i, \boldsymbol{\eta}_{\cdot k}) \tag{4.7}$$

If document i is unlabeled, we first draw the parameter $\pi = p(Z_i = 1)$, the overall probability that any given document belongs to the first class (e.g., political documents), from a Beta distribution with hyperparameters α_0 and α_1.[10] Similarly, for the other class (e.g., non-political documents), we have that $1 - \pi = p(Z_i = 0)$. Given π, for each document indexed by i, we draw the class assignment indicator Z_i from a Bernoulli distribution.[11] Then, we draw features for document i from a multinomial distribution governed by the

[9] In this second approach, we hierarchically map multiple sub-classes into one class e.g., collapsing the classification of documents that are about business and sports into a larger class (non-politics), and letting the remaining documents be about politics.

[10] An anonymous reviewer asked us to further justify our choice of the beta prior over other prior distributions such as the uniform distribution. We opted for a Beta distribution with hyperparameters α_0 and α_1 for a couple of reasons. First, it is conditionally conjugate in our model, allowing for efficient computation of posterior updates for π, as demonstrated in SI **??**. Conjugate prior distributions often provide good approximations and simplify computations, similar to standard likelihood models (Gelman et al., 2014, p. 36). This principle also applies to our model, since the model for Z_i given π is the Bernoulli distribution for which the Beta distribution is conjugate. We note that the uniform prior is a special case of the Beta prior with $\alpha_0 = \alpha_1 = 1$ (e.g., Blitzstein and Hwang, 2019, p. 380). Second, unless α_0 and α_1 are significantly large compared to the number of documents in each class, their selection has minimal impact on estimating π, as discussed in SI **??**. We set $\alpha_0 = \alpha_1 = 2$ in our study to avoid prior density on extreme values of π such as $\pi = 0$ and $\pi = 1$ while ensuring computational feasibility, but our package provides the option for setting the prior parameter values of the user's choice. We thank the anonymous reviewer for raising this point.

[11] An alternative approach would be to allow groups of documents to have distinct values of π. In such a setting, for each observation i in group g, we could have $\pi_g = p(Z_i = k | G_i = g)$, where G_i is a variable indicating the group assignment of document i and the total number of groups is smaller than N. This modeling strategy can be beneficial for datasets with inherent group structures like longitudinal data, especially when the group hierarchy is observed. Yet, the datasets utilized in this paper lack a clear pre-established group

total number of words in document i (n_i) and the vector $\boldsymbol{\eta}_{\cdot k}$ represents the kth column of the $V \times K$ matrix $\boldsymbol{\eta}$, where each entry of $\boldsymbol{\eta}_{\cdot k}$ is represented by the scalar $\eta_{vk} = p(D_{iv}|Z_i = k)$. The prior of $\boldsymbol{\eta}_{\cdot k}$ is the Dirichlet distribution with hyperparameter vector $\boldsymbol{\beta}_{\cdot k}$ (the kth column of the $V \times K$ matrix $\boldsymbol{\beta}$). Finally, $\mathbf{D}_{i\cdot}$ is a row vector of length V that represents the word counts of document i. Conditional on $Z_i = k$, $\mathbf{D}_{i\cdot}$ is drawn from a multinomial distribution with parameters n_i and $\boldsymbol{\eta}_{\cdot k}$. If document i has a label, the key distinction from the scenario with unlabeled data is that each Z_i is not drawn from a Bernoulli distribution. Instead, its value is manually determined through hand-coding.[12] Other than this point, the structure of the model remains unchanged for the labeled data.

Altogether, if we denote $\mathcal{L}_{\text{obs}}(\pi, \boldsymbol{\eta}|\mathbf{D}, \mathbf{Z}, \lambda)$ as the observed data log-likelihood for all the data, based on equations (4.1) to (4.7), then we can express it as:

$$\mathcal{L}_{\text{obs}}(\pi, \boldsymbol{\eta}|\mathbf{D}, \mathbf{Z}, \lambda) = \mathcal{L}_{\text{labeled}}(\pi, \boldsymbol{\eta}|\mathbf{D}_{\text{labeled}}, \mathbf{Z}_{\text{labeled}}) + \lambda \times \mathcal{L}_{\text{unlabeled}}(\pi, \boldsymbol{\eta}|\mathbf{D}_{\text{unlabeled}}, \mathbf{Z}_{\text{unlabeled}}) \tag{4.8}$$

In other words, the result of adding the information from the observed log-likelihoods for the labeled and unlabeled data, respectively, and where, $\lambda \in [0, 1]$ is a parameter that adjusts the influence originating from the unlabeled data on the observed data log-likelihood. The inclusion of such a parameter follows from the scarcity of labeled data compared to the abundance of unlabeled data, which is a significant challenge in implementing semi-supervised learning approaches, as the likelihood function of text in unlabeled documents is likely to overwhelm that of the labels. This is a common problem with combining information from text and other sources through likelihood-based methods, as text data usually contain an order of magnitude more observed variables—features—than other types of data. To ensure that a classifier effectively extracts information from labeled data and is not solely influenced by unlabeled data, it is crucial to enhance the relative importance of labeled data; otherwise, the signal from labeled data will be overshadowed by the overwhelming presence of unlabeled data. To address this, we weight information from unlabeled documents by utilizing a decision factor, λ (Nigam et al., 2000).[13] When λ equals 1, the model equally considers each document, irrespective of whether it is labeled by human supervision or labeled probabilistically by the algorithm. As λ moves from 1 to 0, the model reduces the importance of information contributed by probabilistically labeled documents in the estimation of $\boldsymbol{\eta}$ and

structure. Therefore, instead of specifying π_g, we opted for specifying π. While incorporating a hierarchical structure to π could be an interesting future extension of our model, we leave it for future research. We thank an anonymous reviewer for highlighting this point.

[12] In equation 4.1, we use $\rightarrow$ to represent a deterministic assignment of the classes to documents.

[13] Kim et al. (2018, pp. 217–8) use a similar strategy to balance the information from both a smaller dataset (roll calls) and a larger dataset (textual data) within a model designed for estimating ideal points.

π. When λ reaches 0, the model disregards the information from all probabilistically labeled documents, turning it into a supervised algorithm.

Additionally, note that an important advantage of the interpretability of the key model parameters facilitates augmentation of the model using additional information such as domain knowledge. For example, each element of $\boldsymbol{\eta}_{\cdot k}$ represents the probability of observing a unique feature given the class of the document. In the Section "Active Keyword Upweighting," we show how to augment the model to allow some features to be highly associated with an specific class and in that way improve performance.

Finally, because the observed data log-likelihood of our model is difficult to maximize, we use the EM algorithm to estimate the parameters.[14]

4.3.2 An Active Learning Algorithm

Our active learning algorithm (see Algorithm 3) can be split into the following steps: *estimation* of the probability that each unlabeled document belongs to the positive class, *selection* of the unlabeled documents whose predicted class is most uncertain, and *labeling* of the selected documents by human coders. The algorithm iterates until a stopping criterion is met. In this section, we also describe an optional keyword upweighting feature, where a set of user-provided keywords provide prior information about the likelihood that a word is generated by a given class to the model. These keywords can either be provided at the outset of the model or identified during the active learning process.

We now proceed to describe in detail each step of our algorithm:

4.3.2.1 Estimation

In the first iteration, the model is initialized with a small number of labeled documents.[15] The information from these documents is used to estimate the parameters of the model: the probability of a document being e.g., about politics, π, and $V \times 2$ matrix $\boldsymbol{\eta}$, represents the feature-class probabilities. If there is no labeled data, the model can be initialized by manually assigning initial values to the model parameters. These values can be set randomly or to a fixed value. From the second iteration on, we use information from both labeled and unlabeled documents to estimate the parameters using the EM algorithm, with the log-likelihood of unlabeled documents being weighted by λ, and with the $\boldsymbol{\eta}$ and π values from

[14]For a full derivation of the EM algorithm for our binary classification model and its graphical representation, see SI **??**. Furthermore, refer to SI **??** for the corresponding details (model description, estimation, and graphical representation) for our model extension to multiclass classification and to SI **??** for the details regarding our second extension i.e., binary classification with multiple classes.

[15]While we assume that these documents are selected randomly, the researcher may choose any subset of labeled documents with which to initialize the model.

Algorithm 3 Active learning with EM algorithm to classify text

Randomly select a small subset of documents, and ask humans to label them
Initialize $\mathbf{D}^l_{old}$ with the labeled subset and $\mathbf{D}^u \leftarrow \mathbf{D} \setminus \mathbf{D}^l_{old}$
[**Active Keyword**]: Ask humans to provide initial keywords
Initialize keyword matrix $\boldsymbol{\kappa}$
while Stopping conditions are not met yet **do**
    (1) [**Active Keyword**]: Up-weight the importance of keywords associated with a class
    Up-weight elements of the $\boldsymbol{\beta}$ prior using $\boldsymbol{\kappa}$
    (2) Predict labels for unlabeled documents using EM algorithm
    (3) Select documents with the highest uncertainty among unlabeled documents, and ask humans to label them
    Sample n most uncertain documents in $\mathbf{D}^u$ and have humans label them
    $\mathbf{D}^l_{new} \leftarrow n$ most uncertain documents in $\mathbf{D}^u$
    (4) [**Active Keyword**]: Select words most strongly associated with each class, and ask humans to label them
    Sample m non-keywords most associated with each class and have humans label to create $\boldsymbol{\kappa}_{new}$
    $\boldsymbol{\kappa} \leftarrow \boldsymbol{\kappa}_{new}$
    (5) Update sets of labeled and unlabeled documents for the next iteration
    $\mathbf{D}^l \leftarrow \mathbf{D}^l_{old} \cup \mathbf{D}^l_{new}$
    $\mathbf{D}^u \leftarrow \mathbf{D} \setminus \mathbf{D}^l_{old}$
end whilereturn Obtain predicted classes of all documents

the previous iteration as the initial values. Using the estimated parameters, we compute the probability that each unlabeled document belongs to the politics class.

4.3.2.2 Selection

Using the predicted probability that each unlabeled document belongs to the politics class, we use Shannon Entropy (that is, the level of uncertainty) to determine which of the probabilistically labeled documents it was least certain about. In the binary classification case, this is the equivalent of calculating the absolute value of the distance between the politics class probability and 0.50 for each document. Using this criterion, the model ranks all probabilistically labeled documents in descending order of uncertainty. The n most uncertain documents are then selected for human labeling, where n is the number of documents to be labeled by humans at each iteration.

4.3.2.3 Labeling

A human coder reads each document selected by the algorithm and imputes the "correct" label. For example, the researcher may be asked to label as political or non-political each of the following sentences:

> The 2020 Presidential Election had the highest turnout in US history ⟶ [Political]
>
> Argentina wins the 2022 FIFA World Cup, defeating France ⟶ [Non-political]

These newly-labeled documents are then added to the set of human-labeled documents, and the process is repeated from the estimation stage.

4.3.2.4 Stopping Rule

Our method is highly modular and supports a variety of stopping rules. This includes an internal stability criterion, where stoppage is based on small amounts of change of the internal model parameters, as well as the use of a small held-out validation set to assess the marginal benefit of labeling additional documents on measures of model evaluation such as accuracy or F1. With either rule, the researcher specifies some bound such that if the change in model parameters or out-of-sample performance is less than the pre-specified bound, then the labeling process ends. For example, we use the out-of-sample validation stopping rule with a bound of 0.01 for the F1 score in Section "Reanalysis with Fewer Human Annotations."

4.3.2.5 Active Keyword Upweighting

The researcher also has the option to use an active keyword upweighting scheme, where a set of keywords is used to provide additional information. This is done by incrementing elements of the $\boldsymbol{\beta}$ (the prior parameter of $\boldsymbol{\eta}$) by γ, a scalar value chosen by the researcher. In other words, we impose a tight prior on the probability that a given keyword is associated with each class.[16] To build the set of keywords for each class, 1) *activeText* proposes a set of candidate words, 2) the researcher decides whether they are indeed keywords or not,[17] and 3) *activeText* updates the parameters based on the set of keywords.

To select a set of candidate keywords, *activeText* calculates the ratio that each word was generated by a particular class using the $\boldsymbol{\eta}$ parameter. Specifically, it computes $\eta_{vk}/\eta_{vk'}$ for $k = \{0, 1\}$ with k' the opposite class of k, and chooses top m words whose $\eta_{vk}/\eta_{vk'}$ are

[16] See Eshima et al. (2024) for a similar approach for topic models.

[17] The researcher may also provide an initial set of keywords, and then iteratively adds new keywords.

the highest as candidate keywords to be queried for class k.[18] Intuitively, words closely associated with the classification classes are proposed as candidate keywords. For example, words such as "vote," "election," and "president," are likely to be proposed as the keywords for the political class of documents in the classification between political vs. non-political documents.

After *activeText* proposes candidate keywords, the researcher decides whether they are indeed keywords or not. This is where the researcher can use her expertise to provide additional information. For example, she can decide names of legislators and acronyms of bills as keywords for the political class.

Using the set of keywords for each class, *activeText* creates a $V \times 2$ keyword-class matrix $\boldsymbol{\kappa}$ where each element κ_{vk} takes the value of γ if word v is a keyword for class k, otherwise 0. Before we estimate parameters in each active iteration, we perform a matrix sum $\boldsymbol{\beta} \leftarrow \boldsymbol{\kappa} + \boldsymbol{\beta}$ to incorporate information from keywords. The keyword approach therefore effectively upweights our model with prior information about words that the researcher thinks are likely to be associated with one class rather than another.

4.4 Validation Performance

This section shows the performance comparisons between *activeText* and other classification methods. First, we show comparisons between active and passive learning. Then, we compare classification and time performance between *activeText* and a version of BERT called DistilBERT, a state-of-the-art text classification model using word embeddings as vector representations of the data.[19] Finally, we show how keyword upweighting can improve classification accuracy.

We compare the classification performance on each of the following sets of documents: internal forum conversations of Wikipedia editors (class of interest: toxic comment), BBC News articles (political topic), the United States Supreme Court decisions (criminal procedure), and Human Rights allegations (physical integrity rights allegation).[20] We use 80% of each dataset for the training data and hold out the remaining 20% for evaluation. Documents to be labeled are sampled only from the training set, and documents in the test set are not included to train the classifier, even in our semi-supervised approach. The out-of-sample F1

[18] Words are excluded from candidate keywords if they are already in the set of keywords, or if they are already decided as non-keywords. Thus, no words are proposed twice as candidate keywords.

[19] We trained the BERT models using Nvidia V100 Graphics Processing Units (GPUs) on an High-Performance Computing (HPC) platform.

[20] SI **??** presents a comprehensive description of the validation data and the preprocessing required for analyses.

score is calculated using the held-out testing data.[21]

4.4.1 Classification Performance

Figure 4.2 shows the results from three model specifications: *activeText* (denoted by the solid line); *Random Mixture*, a version of *activeText* that uses passive instead of active learning (denoted by the dotted line); and DistilBERT (denoted by the dashed line).

Each panel corresponds to a unique combination of a dataset and the proportion of documents associated with the class of interest, with the rows corresponding to the datasets and the columns corresponding to the proportions. The parentheses beside the name of each corpus represent the proportion of positive labels in the population configuration i.e., the proportion of documents in the corpus that are labeled as the class of interest.[22] Within each panel, the x-axis represents the number of documents labeled, and the y-axis represents the average out-of-sample F1 score averaged across 50 and 10 Monte Carlo simulations in the case of the *activeText* models and the DistilBERT model, respectively. In the *activeText* models, 20 documents are labeled in each iteration.[23]

There are two key takeaways from Figure 4.2. First, we show that *activeText* is either equivalent to or outperforms its random sampling counterpart in nearly all cases, and the benefit from active learning is larger when the proportion of documents in the class of interest is smaller. The exception is the Human Rights corpus, where the benefit of active learning is marginal, and where at the 50% proportion, random sampling slightly outperforms active learning with less than 200 labeled documents.

Second, we show that in nearly all cases, *activeText* either outperforms or performs comparably to the DistilBERT model. As in the comparison between the active and random versions of *activeText*, the advantage of *activeText* is larger when the proportion of documents in the class of interest is small. This is particularly true in the case of the BBC, Supreme Court, and Wikipedia corpora for the 5% and Population specifications. This advantage is not permanent, however: as the number of labeled documents increases, DistilBERT (as expected) performs well and even exceeds the F1 score of *activeText* in the case of Wikipedia. As before, the exception is the Human Rights corpus, where DistilBERT outperforms *activeText* at the 50% and Population levels.[24]

[21] See SI **??** for a detailed description of the F1-score and other commonly used model evaluation metrics.

[22] See SI **??** for more details on how we generate validated data with class-imbalance.

[23] Table **??** in SI (Dataverse-only) **??** presents similar evidence, based on other evaluation metrics (precision and recall). In addition, Figure **??** in SI (Dataverse-only) **??** includes comparisons of our generative approach, *activeText*, in terms of predictive performance against SVM, a popular discriminative method used for classification tasks.

[24] Figure **??** of SI **??** illustrates that the multiclass version of *activeText* performs better than other alternative models for the BBC and Supreme Court datasets. Additionally, our findings in Figures **??** and

The early poor performance of *activeText* on the Human Rights corpus may be due to the fact that documents are short. Short labeled documents provide less information, making it more difficult for the model to distinguish between classes. We discuss how the information can be augmented using keywords to improve our method's classification performance in Section "Benefits of Keyword Upweighting." The keyword upweighting we propose takes advantage of the substantive interpretability of the feature-class matrix $\boldsymbol{\eta}$ in our generative model.

4.4.2 Runtime

In Figure 4.3, we compare computational runtime for *activeText* and DistilBERT. For this analysis, our goal was to compare how long it would take a researcher without access to a High-Performance Computing (HPC) platform or an expensive GPU to train these models. To this end, we trained the *activeText* and DistilBERT models on a base model M1 Macbook Air with 8 GB of RAM and 7 GPU cores. While the *activeText* models were trained using a single central processing unit (CPU), we used the recent implementation of support for the GPU in M1 Macs in PyTorch[25] to parallelize the training of the BERT model using the M1 Mac's GPU cores.[26] We also computed the time values *cumulatively* for *activeText* since it is expected that model will be fit over and over again as part of the active learning process, whereas for a model like BERT we expect that the model would only be run once, and as such do not calculate its run-time cumulatively. For the Human Rights and Wikipedia corpora, which each have several hundred thousand entries, we used a random subsample of 50,000 documents. For the Supreme Court and BBC corpora, we used the full samples. Finally, we present the time results in logarithmic scale to improve visual interpretation.

Figure 4.3 shows that using DistilBERT comes at a cost of several orders of magnitude of computation time relative to *activeText*. Using the Wikipedia corpus as an example, at 500 documents labeled the baseline *activeText* would have run to convergence 25 times, and the sum total of that computation time would have amounted to just under 100 seconds. With DistilBERT, however, training a model with 500 documents and labeling the remaining 45,500 on an average personal computer would take approximately 10,000 seconds (2.78 hours).

?? in SI **??** indicate that even in binary classification tasks, *activeText* excels when considering the presence of multiple classes. Again, this is especially noticeable in datasets such as the BBC and Supreme Court corpora, where the number of underlying classes exceeds 2.

[25]See https://pytorch.org/blog/introducing-accelerated-pytorch-training-on-mac/.

[26]Specifically, we trained a DistilBERT model (see Sanh et al., 2019) for three epochs (the number of passes of the entire training dataset BERT has completed) using the default configuration from the Transformers and PyTorch libraries for the Python programming language and used the trained model to predict the labels for the remaining documents for each corpus.

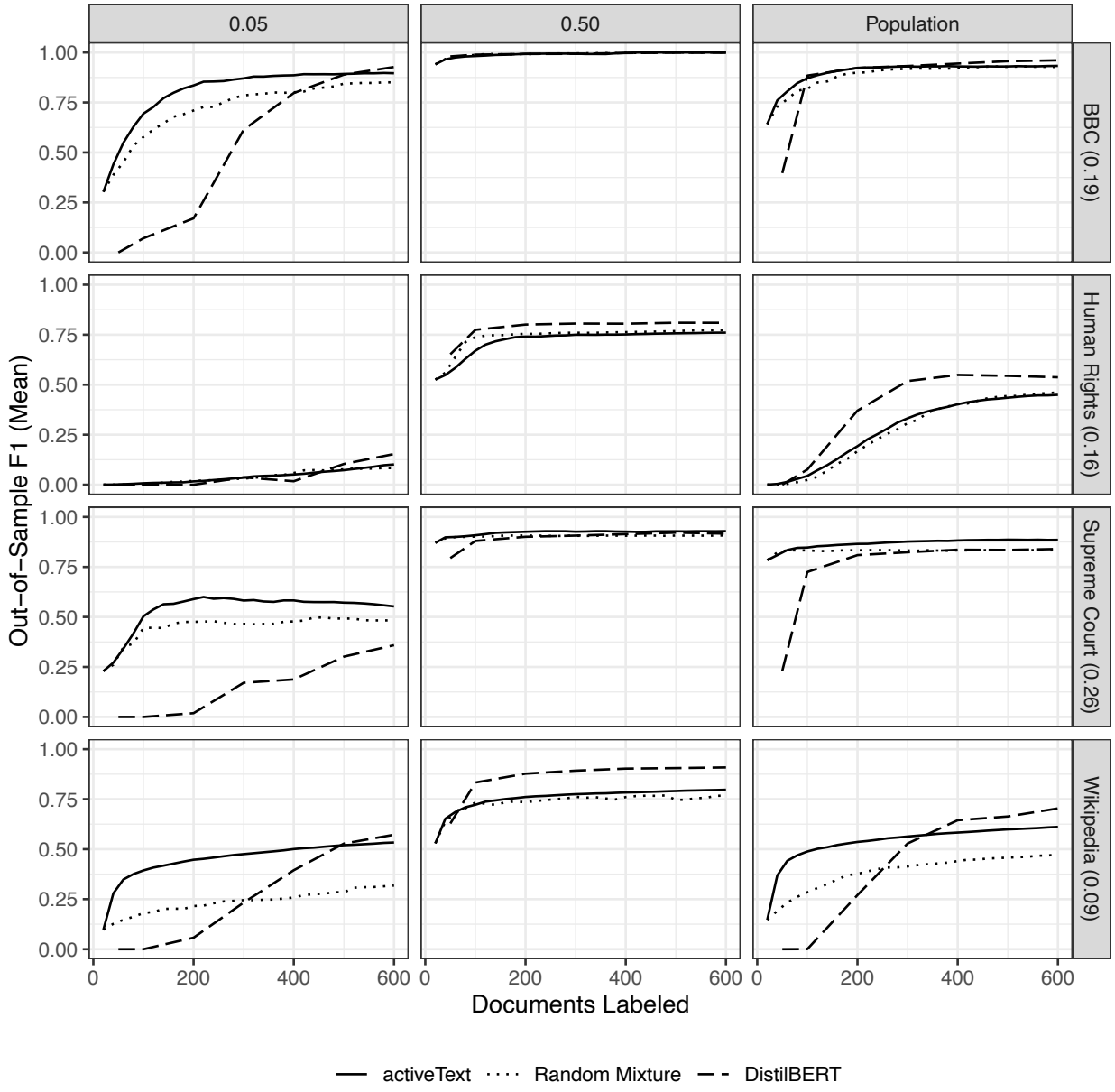


Figure 4.2: **Comparison of Classification Results with Random and Active Versions of *activeText* and DistilBERT**

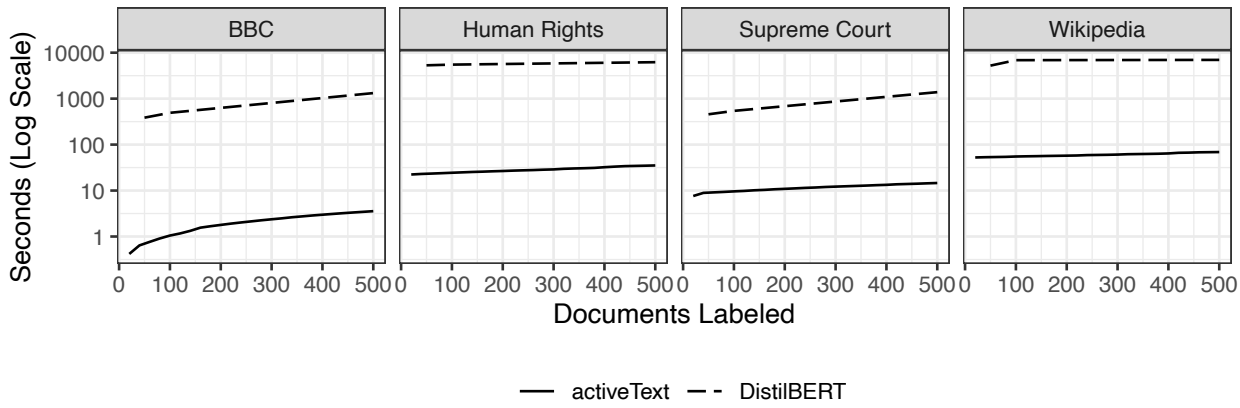


Figure 4.3: **Comparison of Classification and Time Results across *activeText*and DistilBERT**

4.4.3 Benefits of Keyword Upweighting

In Figure 4.2, active learning did not improve the performance on the human rights corpus, and the F1 score was lower than other corpora in general. One reason for the early poor performance of *activeText* may be the length of the documents. Because each document of the human rights corpus consists of one sentence only, the average length is shorter than other corpora.[27] This means that the information the models can learn from labeled documents is less compared to the other corpora with longer documents.[28] In situations like this, providing keywords in addition to document labels can improve classification performance because it directly shifts the values of the feature-class probability matrix, $\boldsymbol{\eta}$, even when the provided keywords is not in the already labeled documents.

Figure 4.4 compares the performance with and without providing keywords. The darker lines show the results with keywords and the lighter lines without. The columns specify the proportion of documents associated with the class of interests: 5%, 50%, and the population proportion (16%). As in the previous exercises, 20 documents are labeled at each sampling step, and 100 Monte Carlo simulations are performed to stabilize the randomness due to the initial set of documents to be labeled. We simulated the process of a user starting with no keywords for either class and then being queried with extreme words indexed by v whose $\eta_{vk}/\eta_{vk'}$ is the highest for each class k, with up to 10 keywords for each class being chosen based on the estimated $\boldsymbol{\eta}$ at a given iteration of the active process. To determine whether

[27]With the population data, the average length of each document is 121 (BBC), 17 (Wikipedia), 1620 (Supreme Court), and 9 (Human Rights)

[28]In our simulation studies described in Section "The Bias of Active Learning" and SI **??**, we confirmed that the classification performance is poor when the document length is short. Please refer to SI (Dataverse-only) **??** for the full set of results.

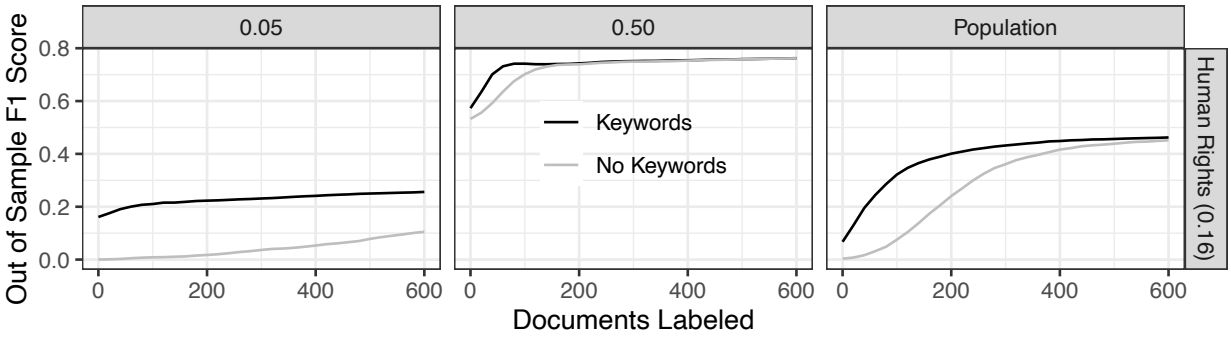


Figure 4.4: **Classification Results of *activeText* with and without Keywords**

a candidate keyword should be added to the list of keywords or not, our simulated user checked whether the word under consideration was among the set of most extreme words in the distribution of the 'true' $\boldsymbol{\eta}$ parameter, which we previously estimated by fitting our mixture model with the complete set of labeled documents.[29]

The results suggest that providing keywords improves performance when the proportion of documents is markedly imbalanced across classes. The keywords scheme improved the performance when the number of labeled documents is smaller on the corpus with 5% or 16% (population) labels associated with the class of interest. By contrast, it did not on the corpus where both classes were evenly balanced. These results highlight that our active keyword approach benefits the most when the dataset suffers from serious class imbalance problems.[30]

One caveat is that we provided 'true' keywords, in the sense that we used the estimated $\boldsymbol{\eta}$ from a fully labeled dataset. In practice, researchers have to decide if candidate keywords are indeed keywords using their substantive knowledge. In this exercise, we believe that the keywords supplied to our simulation are what researchers with substantive knowledge about physical integrity rights can confidently adjudicate. For example, the keywords, such as "torture," "beat," and "murder," match our substantive understanding of physical integrity right violation. Nevertheless, as we explain in Section "Labeling Error" and SI **??**, humans can make mistakes, and some words may be difficult to judge. Thus, we examined the classification performance with varying degrees in the amount of error at the keyword labeling step. In SI **??**, we show that the active keyword approach still improves the classification

[29]Specifically, the simulated user checked whether the word in question was in the top 10% of most extreme words for each class using the 'true' $\boldsymbol{\eta}$ parameter. If the candidate word was in the set of 'true' extreme words, it was added to the list of keywords and upweighted accordingly in the next active iteration.

[30]Figure **??** in SI (Dataverse-only) **??** demonstrates how active keyword works by visualizing the feature-class matrix, $\boldsymbol{\eta}$, at each active iteration. In particular, we show how the keyword scheme accelerates the learning process of the feature-class matrix $\boldsymbol{\eta}$.

performance compared to the no-keyword approach – even in the presence of small amounts (less than 20%) of "honest" (random) measurement error in keyword labeling.

4.5 Reanalysis with Fewer Human Annotations

To further illustrate the benefits of our proposed approach for text classification, we conduct reanalyses of two recently published aritcles: Gohdes (2020) and Park et al. (2020). We show that with *activeText*, we can arrive at the same substantive conclusions advanced by these authors but using only a small fraction of the labeled data they originally used.

4.5.1 Internet Accessibility and State Violence (Gohdes, 2020)

In the article "Repression Technology: Internet Accessibility and State Violence," Gohdes (2020) argues that higher levels of Internet accessibility are associated with increases in targeted repression by the state. The rationale behind this hypothesis is that through the rapid expansion of the Internet, governments have been able to improve their digital surveillance tools and target more accurately those in the opposition. Thus, even when digital censorship is commonly used to diminish the opposition's capabilities, Gohdes (2020) claims that digital surveillance remains a powerful tool, especially in areas where the regime is not fully in control.

To measure the extent to which killings result from government targeting operations, Gohdes (2020) collects 65,274 reports related to lethal violence in Syria. These reports contain detailed information about the person killed, date, location, and cause of death. The period under study goes from June 2013 to April 2015. Among all the reports, 2,346 were hand-coded by Gohdes, and each hand-coded report can fall under one of three classes: 1) government-targeted killing, 2) government-untargeted killing, and 3) non-government killing. Using a document-feature matrix (based on the text of the reports) and the labels of the hand-coded reports, Gohdes (2020) trained and tested a state-of-the-art supervised decision tree algorithm (extreme gradient boosting, `XGboost`). Using the parameters learned at the training stage, Gohdes (2020) predicts the labels for the remaining reports for which the hand-coded labels are not available. For each one of the 14 Syrian governorates (the second largest administrative unit in Syria), Gohdes (2020) calculates the proportion of biweekly government targeted killings. In other words, Ghodes collapses the predictions from the classification stage at the governorate-biweekly level.

We replicate Gohdes (2020) classification tasks using *activeText*. In terms of data preparation, we adhere to the very same decisions made by Gohdes (2020). To do so, we use

the same 2,346 hand-labeled reports (1,028 referred to untargeted killing, 705 to a targeted killing, and 613 a non-government killing) of which 80% were reserved for training and 20% to assess classification performance. In addition, we use the same document-feature matrices.[31] As noted in Section "An Active Learning Algorithm," because *activeText* selects (at random) a small number of documents to be hand-labeled to initialize the process, we conduct 100 Monte Carlo simulations and present the average performance across initializations. As in "Validation Performance," we set $\lambda = 0.001$. The performance of *activeText* and `XGboost` is evaluated in terms of out-of-sample F1 score. Following the discussion above, we stopped the active labeling process at the 30th iteration when the out-of-sample F1 score stopped increasing by more than 0.01 units (our pre-specified threshold). Table 4.1 presents the results.[32] Overall, we find that as the number of active learning steps increases, the classification performance of *activeText* is similar to the one in Gohdes (2020). However, the number of hand-labeled documents that are required by *activeText* is significantly smaller (around one-third) if compared to the ones used by Gohdes (2020).

Table 4.1: Classification Performance: Comparison with Gohdes (2020) results

			Ouf-of-sample F1 Score per class		
Model	Step	Labels	Untargeted	Targeted	Non-Government
activeText	0	20	0.715	0.521	0.800
	10	220	0.846	0.794	0.938
	20	420	0.867	0.828	0.963
	30	**620**	**0.876**	**0.842**	**0.963**
	40	820	0.879	0.845	0.961
Gohdes (2020)		1876	0.910	0.890	0.940

In social science research, text classification is often not the end goal but a means to quantify a concept that is difficult to measure and make inferences about the relationship between this concept and other constructs of interest. In that sense, to empirically test her claims, Gohdes (2020) conducts regression analyses where the proportion of biweekly government targeted killings is the dependent variable and Internet accessibility is the main independent variable – both covariates are measured at the governorate-biweekly level. Gohdes (2020) finds that there is a positive and statistically significant relationship between Internet access and the proportion of targeted killings by the Syrian government. Using the predictions from *activeText*, we construct the main dependent variable and replicate the main regression

[31] Gohdes (2020) removed stopwords, punctuation, and words that appear in at most two reports, resulting in 1,342 features and a document-feature matrix that is 99% sparse. The median number of words across documents is 13.

[32] The values in the bottom row are based on Gohdes (2020), Table A9.

analyses in Gohdes (2020).[33]

Tables **??** and **??** in SI (Dataverse-only) **??** report the estimated coefficients, across the same model specifications in Gohdes (2020). The point estimates and the standard errors are almost identical whether we use `XGboost` or *activeText*. Moreover, Figure 4.5 presents the expected proportion of targeted killings by region and Internet accessibility, using the preferred regression specification by Gohdes. This model, as detailed in column V of Tables **??** and **??**, incorporates the interaction between region and Internet accessibility. Gohdes finds that in the Alawi region, which is recognized for its loyalty to the regime, higher levels of Internet access correspond to a significantly lower expected proportion of targeted killings compared to other regions in Syria. In the absence of the Internet, however, there is no discernible difference across regions (see Figure 4.5, right panel). Our reanalysis does not change the substantive conclusions by Gohdes (2020) (Figure 4.5, left panel), however, it comes just at a fraction of the labeling efforts (labeling 620 instead of 1876 reports).

4.5.2 Human Rights are Increasingly Plural (Park et al., 2020)

Park et al. (2020) investigate how the rapid growth (in the last four decades) of information communication technologies (ICTs) has changed the composition of texts referring to human rights, and show that the average sentiment with which human rights reports are written has not drastically changed over time. They claim that if one wants to really understand the effect of changes in the access to information on the composition of human rights reports, it is necessary to internalize the fact that human rights are plural (i.e., bundles of related concepts). In other words, the authors argue that having access to new information has indeed changed the taxonomy of human rights over time, even when there has not been a change in tone.

To empirically test such a proposition, Park et al. (2020) conduct a two-step approach. First, by training an SVM for text classification with three classes (negative, neutral, and positive sentiment), the authors show that the average sentiment of human rights reports has indeed remained stable even in periods where the amount of information available has become larger.[34] Second, they use a network modeling approach to show that while the average sentiment of these reports has remained constant over time, the taxonomy has drastically

[33]The results presented in SI (Dataverse-only) **??** demonstrate two main findings. First, the classification results of *activeText*, as shown in Table **??**, are almost identical to that of Gohdes (2020). Second, the proportion of biweekly government targeted killings from *activeText*, depicted in Figure **??**, is also highly consistent with the same measure by Gohdes (2020).

[34]As explained in Appendix A1 of Park et al. (2020), negative sentiment refers to text about a clear ineffectiveness in protecting or to violations of human rights; positive sentiment refers to text about clear support (or no restrictions) of human rights; and neutral sentiment refers to stating a simple fact about human rights.

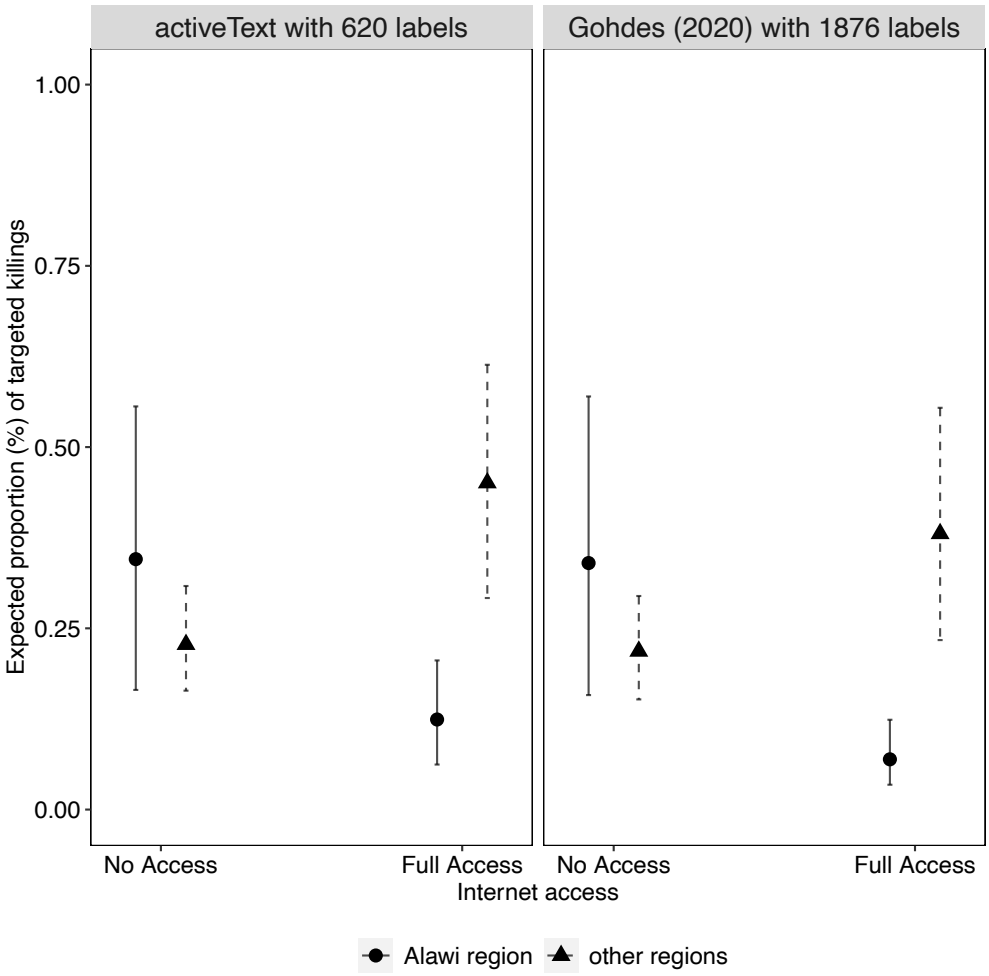


Figure 4.5: **Replication of Figure 3 in Gohdes (2020): Expected Proportion of Target Killings, Given Internet Accessibility and Whether a Region is Inhabitated by the Alawi Minority.** The results from *activeText* are presented in the left panel and those of Gohdes (2020) are on the right.

changed. In this section, using *activeText*, we focus on replicating the text classification task of Park et al. (2020), which is key to motivating their puzzle.

As in the reanalyses of Gohdes (2020), we adhere to the same pre-processing decisions made by Park et al. (2020) when working with their corpus of Country Reports on Human Rights Practices from 1977 to 2016 by the US Department of State. In particular, we use the same 4000 hand-labeled human rights reports (1182 are positive, 1743 are negative, and 1075 are neutral) and use the same document-feature matrices (which contain 30,000 features, a combination of unigrams and bigrams). Again, we conduct 100 Monte Carlo simulations and present the average performance across initializations. As shown in Figure **??** in SI (Dataverse-only) **??**, we stopped the active labeling process at the 25th iteration of our

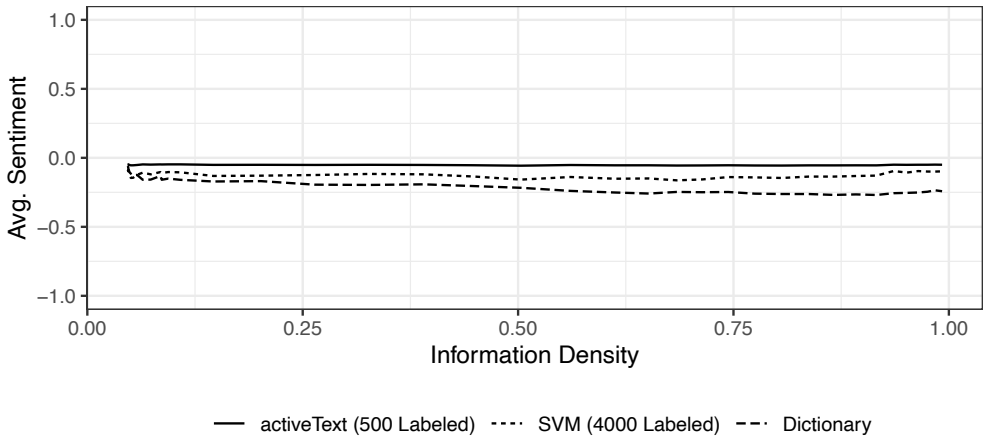


Figure 4.6: **Replication of Figure 1 in Park et al. (2020): The Relationship Between Information Density and Average Sentiment Score.**

algorithm as the out-of-sample F1 score (from an 80/20 training/test split) does not increase by more than 0.01 units.[35] Using the results from the classification task via *activeText*, the sentiment scores of 2,473,874 documents are predicted. With those predictions, we explore the evolution of the average sentiment of human rights reports per average information density score.[36]

Figure 4.6 shows that by labeling only 500 documents with *activeText*, instead of 4000 labeled documents used by Park et al. (2020) to fit their SVM classifier, we arrive at the same substantive conclusion: the average sentiment of human rights reports has remained stable and almost neutral over time. In Figure **??** of SI (Dataverse-only) **??**, we also show that this result is not an artifact of our stopping rule and it is robust to the inclusion of additional label documents (e.g, labeling 1000, 1500, and 2000 documents instead of just 500).

4.6 Discussion

In this section, we address three key issues concerning our proposed approach: 1) the (in-sample) bias that occurs when actively selecting observations to train a model; 2) the impact of mislabeling documents and keywords; and 3) the practical considerations about down-

[35]The only point where we depart from Park et al. (2020) is that we use an 80/20 split for training/testing, while they use k-fold cross-validation. Conducting k-fold cross-validation for an active learning algorithm would require over-labeling because the labeling process should be repeated k times as well. Because of this difference, we refrain from comparing our model performance metrics to theirs.

[36]Information density is a proxy for ICTs based on a variety of indicators related to the expansion of communications and access to information, see Appendix B in Park et al. (2020).

weighting unlabeled data.

4.6.1 The Bias of Active Learning

As highlighted by Dasgupta (2011); Dasgupta and Hsu (2008); Farquhar et al. (2021), active learning introduces in-sample statistical bias due to training non-i.i.d data. As described above, active learning involves selecting the most informative data points for training. This selection can introduce bias because the newly labeled documents may not represent the entire of the target population.

To illustrate the nature of this bias, let's denote the population risk as $r = \mathbb{E}_{\mathbf{Z},\mathbf{D}}[L(Z_i, \widehat{Z}_i)]$. Here, L represents a loss function, such as the L1 or L2 norms. For each observation indexed by i, Z_i denotes its true label, and $\widehat{Z}_i$ denotes the predictions made based on the model parameters and the data's features ($\mathbf{D}$). It is important to note that $\mathbb{E}_{\mathbf{Z},\mathbf{D}}$ indicates the expectation calculated over the joint distribution of $\mathbf{Z}$ and $\mathbf{D}$. In the training stage, we want to find the model parameters that minimize the population risk. However, the population risk is not observed, and we can only estimate it using the labeled data. Hence, the empirical risk, derived from an i.i.d sample of the population with a size denoted as N^l, can be computed by the formula: $\widehat{R} = \frac{1}{N^l}\sum_{j=1}^{N^l}[L(Z_j, \widehat{Z}_j)]$. This calculation provides an unbiased estimate of r because the labeled data is randomly selected from the population. With active learning, the empirical risk is $\widetilde{R} = \frac{1}{M}\sum_{m=1}^{M}[L(Z_m, \widehat{Z}_m)]$, where M is the number of labeled data selected actively. Because these M data points are not a random sample of the population, $\widetilde{R}$ is no longer an unbiased estimator of r.

Farquhar et al. (2021) propose an unbiased estimator of the population risk with active learning, called Leveled Unbiased Risk Estimator (LURE), $\widetilde{R}_{\text{lure}} = \frac{1}{M}\sum_{m=1}^{M}[v_m L(Z_m, \widehat{Z}_m)]$, where v_m is a function of the sampling probabilities of the actively selected documents (refer to SI **??** for more details). They show that $\widetilde{R}_{\text{lure}}$ is an unbiased estimate with minimum variance in its class of weighted estimators for the empirical risk. In other words, the in-sample bias in the active learning process can be corrected by $\widetilde{R}_{\text{lure}}$ (refer to Theorems 3 and 4 in Farquhar et al. 2021).

Importantly, according to Farquhar et al. (2021), one factor that determines whether the in-sample bias correction improves out-of-sample predictive performance is the complexity of the model. For example, they empirically show that the in-sample bias correction with LURE improves the out-of-sample classification performance for a simple linear regression, but not for a neural network. Farquhar et al. (2021) argue that for overparameterized models, correcting the in-sample bias from active learning might not be advantageous to improve out-of-sample classification because active learning can serve as a regularization mechanism

against overfitting bias. While *activeText* does not possess as many parameters as a neural network, it has many more parameters than linear regressions depending on the number of features of the data. This implies that it is an open question whether the in-sample bias correction improves the out-of-sample predictive performance of *activeText*.

To examine whether the in-sample statistical bias has adverse effects on the out-of-sample classification performance, we conducted a series of simulation studies involving 108 different configurations. In our simulation studies, we manipulated various aspects of the simulated data, such as the number of unique words, the average length of the documents (measured in number of words), the difficulty of classification, and the proportion of positive class documents in the corpus.

Figure 4.7 presents the results of implementing the LURE bias correction to *activeText* under a simulation setup. This setup involves generating simulation data with 1000 documents, 500 unique features, an average of 50 features per document, and the reference class accounting for 10% of the corpus. We perform 100 Monte Carlo simulations in this context. The left panel presents the in-sample bias of the $\widetilde{R}$ and $\widetilde{R}_{\text{lure}}$, and the right panel presents the corresponding out-of-sample F1 scores. The bias is calculated as the difference between the population risk and the empirical risk. For active learning, this is represented as $r - \mathbb{E}[\widetilde{R}]$, while for its bias-corrected version, it is $r - \mathbb{E}[\widetilde{R}_{\text{lure}}]$.

Figure 4.7 shows that regarding the in-sample bias of the empirical risk, $\widetilde{R}$ exhibits an upward bias in the early stages of the labeling process, which gradually decreases as more documents are labeled. This bias arises because the most uncertain documents are labeled first in active learning, causing the empirical risk to initially surpass the population risk. In contrast, we find that LURE weights effectively eliminate the in-sample bias of *activeText*. However, as the right panel of Figure 4.7 shows, the unadjusted version of *activeText* demonstrates better out-of-sample classification performance compared to its bias-corrected counterpart. Consequently, our findings indicate that addressing in-sample bias does not necessarily improve the out-of-sample classification performance of *activeText*. These results hold across different simulation settings and in our validation data. For a more detailed overview of the simulation results, interested readers can refer to SI **??**.

4.6.2 Labeling Error

Although our main findings operate under the assumption that labelers are accurate, it is important to acknowledge that human labelers can still make errors. We will now investigate how mislabeling of documents and keywords that are actively selected impacts the classification performance of *activeText*. Specifically, we aim to demonstrate the potential impact

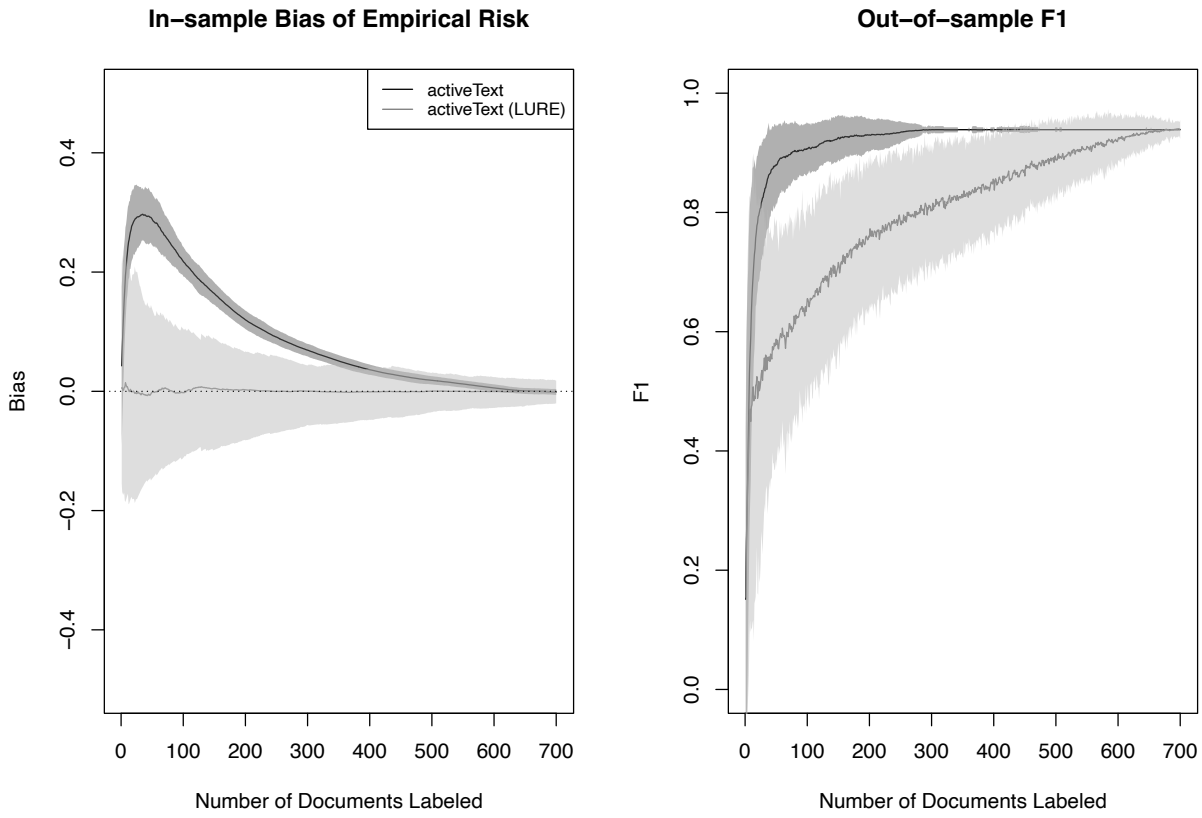


Figure 4.7: **Bias of the Empirical Risk for Labeled Data (left panel) and Out-of-sample Classification Performance (right panel) of *activeText*, *activeText*+LURE. For each panel, the x-axis represents the number of documents labeled, and the y-axis represents the average bias and average out-of-sample F1 score across 100 Monte Carlo simulations. Shaded areas represent the 95% confidence intervals across Monte Carlo simulations.**

of measurement errors in labeling, particularly focusing on the effect of "honest" mistakes (classical measurement error) on classification performance and basic downstream analysis.

In the case of mislabeling actively selected documents, our results show that random perturbations from true document labels do hurt the classification performance.[37] For example, in SI **??**, we find that in the case of the BBC News articles dataset described above, when about 20 out of 200 documents are labeled with the incorrect label (10% mislabeling), the out-of-sample F1 score remains high at around 0.87. However, when the mislabeling of documents exceeds 1 in 5 documents ($\geq$ 20% mislabeling), there is a significant decrease in the out-of-sample F1 score (the F1 is less than 0.75). This pattern holds across all validation

[37]As outlined in SI **??**, in binary classification, honest mistakes in labeling documents entails choosing (at random) the opposite label from the true one.

datasets (refer to Figure ?? in SI ??).

To illustrate how mislabeling affects downstream analyses, we consider a simple example. Suppose we are trying to predict the number of documents related to a specific category, like politics in the BBC data. In our validation studies, we already know the actual proportions of documents in the categories we are interested in. For instance, in the BBC dataset, 19% of the articles cover politics, while in the Wikipedia corpus, 9% of the documents are deemed toxic. Similarly, 26% of Supreme Court cases involve criminal procedure, and 16% of Human Rights reports include allegations of physical integrity rights violations. To gauge the impact of mislabeling, we assess the bias in the predictions for the proportion of documents in the target class made by our model. As illustrated in Figure ?? in SI ??, the bias increases as the rate of mislabeling rises. For example, in the Wikipedia comments dataset, if we accurately label around 200 documents, the bias is minimal. However, introducing 30% mislabeling results in a bias increase of 0.25 units. This trend is consistent across all datasets, mirroring what we observed with the F1 score. Essentially, even minor unintentional errors (honest mistakes) diminish accuracy when computing simple summary statistics such as a sample mean.

In contrast to our results for the mislabeling of documents, if compared to the no-keyword approach, a small amount of classical measurement error on keyword labeling does not hurt the classification performance.[38] For example, the results presented in Figure ?? in SI ?? shows that when random noise is introduced, the classification performance of *activeText* for binary classification decreases slightly as the proportion of mislabeled keywords increases to 30% or more. This trend remains consistent across different validation datasets and values of γ (the upweight assigned to each keyword).

In light of these results, we believe that future research focus on developing new active learning algorithms that prioritize assigning labelers based on their labeling expertise and adapt to various types of labeling errors. One approach could involve allocating the most skilled labelers to annotate the most uncertain or challenging documents, while assigning simpler tasks to less proficient labelers. This strategy could optimize the efficiency of the labeling process. Additionally, as we discussed above, inaccurate predictions can introduce bias, particularly in downstream tasks. This bias can be exacerbated by departures from classical measurement error, making it difficult to determine its direction. Therefore, as recently recognized by authors such as Knox et al. (2022) and Fong and Tyler (2021) further investigation is necessary to directly tackle these potential biases, especially in settings such as a poplar inference methods such as a regression framework.[39]

[38] As described in more detail SI ?? mislabeling keywords implies randomly labeling non-keyword as keywords, and vice versa.

[39] Refer to Egami et al. (2023) for a recent approach to recover regression estimates when noisy predictions

4.6.3 Tuning the value of λ

As noted above, we downweight the information from unlabeled documents as we typically have more unlabeled than labeled documents. Moreover, since the labeled documents have been classified by an expert, we want to rely more on the information they bring for prediction.

An important practical consideration is how to select the appropriate value of λ. One possible approach would be to adopt popular model selection methods (e.g. cross-validation) to choose the appropriate λ value during the model initialization process. However, cross-validation may not be practical when the labeled data is scarce (or absent at the beginning of the process). We have consistently observed across a variety of applications that very small values (e.g., 0, 0.001 or 0.01) work the best on the corpora we used (see Figures **??** and **??** in SI (Dataverse-only) **??**). However, more work is needed to clearly understand the optimality criteria needed to select λ. We leave this question for future research.

4.7 Conclusion

Human labeling of documents is the most labor-intensive part of social science research that uses text data. For automated text classification to work, a machine classifier needs to be trained on the relationship between text features and class labels, and the labels in training data are given manually. In this paper, we have described a new active learning algorithm that combines a mixture model and active learning to incorporate information from labeled and unlabeled documents and better select which documents to be labeled by a human coder. Our validation and simulation studies showed that a moderate number of documents are labeled, and the proposed algorithm performed at least as well as state-of-the-art methods such as BERT at a fraction of the cost. We replicated two published political science studies to show that our algorithm lead to the same conclusions as the original papers but needed much fewer labeled documents. In sum, our algorithm enables researchers to save their manual labeling efforts without sacrificing quality.

Machine learning techniques are becoming increasingly popular in political science, but the barrier to entry remains too high for researchers without a technical background to make use of advances in the field. As a result, there is an opportunity to democratize access to these methods. We believe that our model and **R**-package will provide applied researchers with a tool that they can use to efficiently categorize documents in corpora of varying sizes and topics.

from a model are employed as outcomes in regression analysis.

CHAPTER 5

Conclusion

This dissertation has demonstrated the transformative potential of artificial intelligence, particularly large language models (LLMs), in advancing the study of political discourse. Through three papers, I applied emerging AI techniques—prompt engineering for qualitative measurement, graph-based analysis of debate structures, and active learning for text classification—that significantly push the boundaries of computational political science. These methodologies address long-standing challenges in the field by providing scalable solutions for analyzing complex language phenomena at scale, enabling new types of analysis that were previously impractical.

In the first paper, I developed a scalable approach for measuring deliberative quality in legislative debates by leveraging prompt engineering. This method allows LLMs to generate Discourse Quality Index (DQI) scores that closely approximate human annotations, enabling the analysis of deliberative quality across large corpora. The broader implications of this approach extend beyond the immediate application to U.S. Congressional debates. It opens the door to comparative studies across national parliaments, longitudinal analyses of how deliberative quality evolves over time, and even analyses of online discourse, where similar frameworks can be applied to assess the quality of public deliberation. This paper demonstrates that prompt engineering, as a technique, offers a powerful way to scale qualitative analysis without sacrificing the depth required for meaningful insight.

The second paper extended the use of prompt engineering by applying it to the problem of mapping argumentative structures in parliamentary debates. By developing LegisGraph, a graph-based framework that captures the relationships between speakers, arguments, and topics, I provided a novel way to visualize and analyze the structure of legislative debates. When applied to Canadian parliamentary debates, LegisGraph revealed patterns of argumentation that would be difficult to detect using traditional methods. For instance, it identified clusters of agreement and contradiction across party lines and uncovered a relationship between issue salience and argument sophistication. The findings of this paper highlight the

broader potential of using AI-driven methods to analyze the structure of political discourse, offering new opportunities to study polarization, coalition dynamics, and the evolution of political debate.

The third paper focused on active learning, demonstrating how text classification can be made more efficient with minimal labeled data. By developing activeText, an algorithm that combines probabilistic modeling with active learning, I showed that high classification accuracy can be achieved with only a fraction of the labeled data typically required. This method is particularly valuable for researchers working with large-scale political text datasets but who have limited resources for extensive manual labeling. The broader implications of this approach include the potential to streamline classification tasks across various political domains, enabling efficient analysis of political texts in resource-constrained settings.

While these contributions represent significant advances, further validation is necessary, particularly in the case of LegisGraph. Although LegisGraph offers a novel framework for visualizing the structure of debates, more work is needed to validate whether the argumentative structures it identifies align with established theoretical expectations. Future research should compare LegisGraph's outputs to traditional methods of debate analysis, such as manual coding and network analysis, to assess its accuracy. For example, testing whether the argument clusters detected by LegisGraph correspond with known party coalitions or ideological groupings will help determine its robustness as a tool for understanding legislative discourse.

In the case of the first paper, while LLMs were able to approximate human DQI annotations, further validation is also needed to assess the generalizability of this approach across different contexts and datasets. Future research should systematically compare the automated DQI scores generated by LLMs with human-coded data across various legislative bodies, such as parliaments in different countries or deliberative bodies operating under different institutional frameworks. Moreover, it is important to understand potential biases in the model's output. Conducting error analysis across different types of debates, rhetorical styles, or speaker behaviors will help refine the approach and ensure that the models produce reliable and unbiased results.

A crucial next step is to take the measurements proposed in this dissertation and compare them with other attempts to measure similar concepts. For example, comparing LegisGraph to other methods for mapping argumentative structures, such as manual coding or traditional network analysis, is essential not only for validation but also for uncovering new insights. The AI-driven methods presented in this dissertation offer unique advantages, such as scalability and precision, that can lead to surprising findings. Future research could explore whether LegisGraph reveals previously undetected shifts in party dynamics, new forms of cross-party

agreement, or unexpected coalitions that other methods might overlook.

Error analysis remains an important avenue for future work. While the models in this dissertation generally perform well, they likely make systematic errors in certain contexts, such as highly technical debates, emotionally charged discussions, or debates featuring rhetorical complexity. Understanding these errors is crucial for improving model accuracy and reliability. Cross-validation with human coders in debates involving specialized jargon or politically sensitive topics could provide valuable insights into the limitations of LLMs. By identifying where models struggle, researchers can refine their methodologies to better handle these challenging cases.

Beyond technical challenges, the integration of AI into political discourse analysis raises important ethical and theoretical considerations. As AI tools become more prevalent in the social sciences, issues of transparency, accountability, and bias must be addressed. Ensuring that the outputs of LLMs and AI-driven methods are interpretable, unbiased, and transparent is essential for their responsible use. In particular, scholars must remain vigilant about the potential for these tools to reinforce existing biases or obscure critical nuances in political discourse. The theoretical implications of AI-driven analysis also warrant further exploration. By automating tasks traditionally performed by human coders, AI changes how we conceptualize deliberation, argumentation, and the dynamics of political debate. Understanding how these changes affect our theoretical models of political communication is crucial for the continued development of this field.

In conclusion, this dissertation makes significant contributions to the field of computational political science by introducing new methodologies for scaling and automating the analysis of political discourse. These methodologies—prompt engineering, LegisGraph, and active learning—provide scalable, efficient tools that have the potential to revolutionize how political texts are analyzed. However, much work remains in terms of validation, comparison with existing methods, and error analysis. By addressing these challenges, future research can refine these tools and unlock new insights into the dynamics of legislative debates and democratic governance. Ultimately, this work lays the foundation for a new paradigm in political discourse analysis, offering exciting opportunities for future exploration in both theoretical and practical applications. The methodologies developed here could be applied to comparative studies of deliberative quality across different political systems, to the analysis of online deliberation, or to policy analysis, contributing to a deeper understanding of political communication in both traditional and digital arenas.

APPENDIX A

Appendix for "Towards Qualitative Measurement at Scale: A Prompt-Engineering Framework for Large-Scale Analysis of Deliberative Quality in Parliamentary Debates"

A.1 Large Language Models for Text Classification

A.1.1 Introduction

Large language models (LLMs) have revolutionized the field of natural language processing (NLP), enabling significant advancements in various tasks, including text classification. This appendix provides a technical introduction to LLMs, focusing on the Transformer architecture and its applications in fine-tuning and in-context learning.

A.1.2 The Transformer Architecture

The development of LLMs has been driven by innovations in machine learning, such as word embeddings, neural networks, and the Transformer architecture. Word embeddings are dense vector representations that capture the semantic meaning of words in a continuous space Mikolov et al. (2013). Neural networks have provided the foundation for complex models capable of learning from and generating text data Elman (1990); Hochreiter and Schmidhuber (1997); Kim (2014).

The Transformer architecture Vaswani et al. (2017) combines the strengths of previous neural network models to create a powerful, scalable, and effective neural network for various NLP tasks. It consists of an encoder, which processes the input text, and a decoder, which generates the output text. The key components of the Transformer are:

- **Self-attention mechanisms**: These allow the model to attend to different parts of the input sequence, capturing the relationships between words regardless of their position.
- **Feed-forward layers**: These layers process the information in a single direction, transforming the representations learned by the self-attention mechanisms.
- **Positional encoding**: This technique incorporates word order information into the input representations, enabling the model to capture the relative positions of words in the sequence.

The Transformer's parallelizable design enables training on large datasets, which is crucial for performance Halevy et al. (2009). By stacking multiple layers of self-attention and feed-forward networks, the Transformer can learn complex relationships between words and generate highly contextual representations.

A.1.3 Fine-tuning with BERT

BERT (Bidirectional Encoder Representations from Transformers) is a powerful language model based on the Transformer encoder, designed for various NLP tasks, including text classification Devlin et al. (2019b). BERT is pre-trained on large corpora using two objectives:

- **Masked Language Modeling (MLM)**: A certain percentage of input tokens are masked, and the model is trained to predict the original tokens based on the context provided by the unmasked tokens.
- **Next Sentence Prediction (NSP)**: The model is trained to predict whether two sentences follow each other in the original text, helping it learn relationships between sentences.

The pre-training process allows BERT to learn bidirectional contextual representations, which can then be fine-tuned for specific tasks. Fine-tuning involves adding a task-specific output layer on top of the pre-trained BERT model and training the entire model with a smaller learning rate. This process adapts the pre-trained representations to the target task, often requiring only a small amount of task-specific training data.

Variations of BERT, such as RoBERTa Liu et al. (2019) and DistilBERT Sanh et al. (2019), have introduced improvements in training methodology and model compression, further advancing the state-of-the-art in text classification and other NLP tasks.

A.1.4 In-context Learning with Generative LLMs

Generative LLMs, such as GPT-3 Brown et al. (2020) and GPT-4 OpenAI (2023), are state-of-the-art autoregressive language models that excel in various NLP tasks, including text classification. GPT models differ from BERT in several key aspects:

- **Computational scale**: GPT models are much larger than BERT, with GPT-3 having 175 billion parameters compared to BERT's 340 million. This increased capacity allows GPT models to capture more information during pre-training.
- **Context window size**: GPT models have larger context windows, enabling them to process longer sequences and handle long-range dependencies more effectively.

The combination of the causal language modeling objective, increased computational scale, and larger context windows has made GPT models particularly adept at in-context learning. In-context learning involves providing examples of desired input-output pairs (few-shot learning) or even just a task description (zero-shot learning) as part of the prompt. The model uses this context to generate appropriate responses without the need for explicit fine-tuning.

To perform in-context learning for text classification, users provide examples of texts and their corresponding labels in the prompt. The model learns the task format and the relationship between the texts and labels, enabling it to classify new texts based on the provided examples. The prompt can also include additional instructions or context to guide the model's output.

GPT-3 and GPT-4 have demonstrated remarkable performance in zero-shot and few-shot learning scenarios across a wide range of NLP tasks Bubeck et al. (2023). This capability has the potential to significantly reduce the need for large labeled datasets and task-specific fine-tuning, making it easier to apply LLMs to new text classification problems.

A.1.5 Choosing Between Fine-tuning and Prompt Engineering

Fine-tuning and prompt engineering represent two distinct methods for "steering" the output of a language model given some input. Fine-tuning involves adapting the model's parameters to a specific task using labeled data, essentially aligning the model's behavior with the desired output for that task. This approach has been highly effective with models like BERT, where the pre-trained representations are fine-tuned to perform text classification and other tasks with high accuracy.

On the other hand, prompt engineering focuses on designing effective prompts that guide the language model to generate the desired output without modifying its parameters. This

approach is particularly relevant for large language models like GPT-3 and GPT-4, which have demonstrated remarkable zero-shot and few-shot learning capabilities. By carefully crafting prompts that include task instructions, examples, and relevant context, users can align the model's output with their intended goals.

Prompt engineering techniques for AI alignment in text classification tasks may include:

- Providing clear instructions and guidelines for the classification task
- Including representative examples of texts and their corresponding labels
- Specifying the desired format for the model's output (e.g., label only, or label with confidence score)
- Incorporating additional context or constraints to guide the model's decision-making process

The choice between fine-tuning and prompt engineering depends on various factors, such as the availability of labeled data, the complexity of the task, and the specific language model being used. Fine-tuning may be preferred when there is sufficient labeled data and the task requires a high level of customization. Prompt engineering, on the other hand, can be more efficient and flexible, especially when working with large language models that have strong zero-shot and few-shot learning capabilities.

As language models continue to evolve, the development of more sophisticated AI alignment techniques will be crucial to ensure that these models can be effectively applied to a wide range of text classification tasks while maintaining consistency with human values and goals. This may involve a combination of fine-tuning, prompt engineering, and other emerging approaches, such as reinforcement learning with human feedback Christiano et al. (2017) and value alignment Soares and Fallenstein (2016).

A.2 The Discourse Quality Index: Concept, Measurement, and Application

The Discourse Quality Index (DQI) is a theoretically grounded and empirically validated instrument for measuring the quality of deliberation in political speech (Steenbergen et al., 2003). It was developed to advance the empirical study of deliberation by providing a reliable and flexible tool for quantifying deliberative quality across a range of contexts.

A.2.1 Theoretical Foundations

The DQI is firmly rooted in Habermasian discourse ethics, which conceptualizes deliberation as a process of rational argumentation aimed at reaching understanding and agreement (Habermas, 1996). According to Habermas, the key elements of an ideal deliberative process include the free and equal participation of all affected parties, the justified exchange of arguments, respect for opposing views, appeals to the common good rather than narrow interests, and a cooperative search for consensus (Habermas, 1981, 1992).

While recognizing the counterfactual nature of these ideals, the DQI attempts to translate them into observable indicators that capture the essential features of deliberative quality in real-world political debates (Steenbergen et al., 2003). The index focuses on the *speech act* as the unit of analysis, coding each relevant part of a debate according to multiple criteria derived from deliberative theory.

A.2.2 Dimensions and Measurement

The current version of the DQI consists of seven indicators, each measured on an ordinal scale. These indicators and their theoretical justifications are as follows:

1. **Participation**: This codes whether a speaker's participation is impaired by interruptions or formal constraints. It is a binary variable, reflecting the Habermasian ideal of free and equal access to deliberation.

2. **Level of Justification**: This captures the extent to which arguments are supported by reasons, on a scale from 0 (no justification) to 4 (sophisticated justification with multiple linked reasons). It operationalizes the Habermasian emphasis on rational argumentation.

3. **Content of Justification**: This measures whether justifications appeal to narrow group interests (0), neutral considerations (1), or the common good, either in utilitarian (2a) or Rawlsian "difference principle" terms (2b). It reflects the deliberative ideal of public-spirited reasoning.

4. **Respect for Groups**: This codes the degree of respect shown toward affected groups, from explicitly negative (0) to explicitly positive (2) statements. It translates Habermas's notion of empathy and reciprocity.

5. **Respect for Demands**: This measures respect toward the demands of other speakers, using the same scale as above but with an additional top category (3) for explicitly

agreeing with a demand. It captures the deliberative aim of recognizing the merit in others' claims.

6. **Respect for Counterarguments**: This assesses how speakers engage with counterarguments, from ignoring (1) or degrading (0) them to neutrally acknowledging (2), valuing (3), or agreeing with them (4). It operationalizes the weighing of competing arguments in deliberation.

7. **Constructive Politics**: This codes the degree to which speakers propose constructive solutions, from pure positional politics (0) to mediating proposals (3). It reflects the deliberative goal of cooperatively seeking consensus or compromise.

To illustrate the coding scheme, consider the following example speech act from a debate on immigration policy:

> While I understand the economic concerns raised by the opposition, I believe we have a moral duty to prioritize the humanitarian needs of refugees. Numerous studies show that asylum seekers pose little threat to our social cohesion or welfare system. Therefore, while the opposition's views are valid, I maintain that moderately increasing our refugee intake is the right policy for the common good.

This speech would be coded as follows:

- Participation: 1 (normal)
- Level of justification: 3 (multiple complete justifications)
- Content of justification: 2a (appeal to the common good in utilitarian terms)
- Respect for groups: 1 (neutral toward refugees)
- Respect for demands: 2 (explicit respect for opposition demands)
- Respect for counterarguments: 3 (values opposition's economic arguments)
- Constructive politics: 1 (defends current agenda but acknowledges other views)

A.2.3 Validity and Reliability

The theoretical validity of the DQI has been established through its close alignment with key concepts from deliberative theory, particularly Habermasian discourse ethics (Steenbergen et al., 2003). Its empirical reliability has been demonstrated through inter-coder agreement tests, with Cohen's kappa scores ranging from 0.881 to 0.954 across the seven indicators, suggesting excellent consistency between independent coders (Steiner et al., 2004).

A.2.4 Evolution and Application

Since its initial development, the DQI has undergone some refinements to improve its precision and applicability. For example, the original coding protocol assigned ignoring counter-arguments the lowest score on the respect scale. However, recognizing that non-engagement is also common in respectful, non-conflictual debates, the authors adjusted this to a higher (though still suboptimal) category (Bächtiger and Steiner, 2005).

The DQI has now been applied to study deliberation in a wide variety of political contexts, including parliamentary debates, public consultation processes, international negotiations, and online discussions (Bächtiger et al., 2018a). By enabling the quantitative comparison of discourse quality across settings, it has yielded important insights into the institutional, cultural, and issue-specific drivers of deliberative politics.

For instance, Bächtiger and Steiner (2005) found that consensual political systems like Switzerland's tend to produce higher DQI scores than competitive systems like Germany's or the UK's, especially on the respect dimensions. However, this effect interacts with institutional publicity: in consensus systems, the difference between public and non-public debate is minimal, while in competitive systems, closed-door negotiations score much higher than open sessions. Other work has shown the DQI to predict outcomes like unanimous agreement in elite negotiations (Spörndli, 2004).

While the DQI has greatly advanced the empirical study of deliberation, it is not without limitations. Its focus on observable speech means it cannot directly measure certain deliberative ideals like truthfulness or authenticity (Steenbergen et al., 2003). Moreover, its aggregation of complex speech acts into numeric scores necessarily simplifies the qualitative richness of discourse.

Nonetheless, by providing a systematic and reliable means of evaluating discourse against normative standards, the DQI remains an invaluable tool for understanding the realities and possibilities of deliberative democracy. As the field continues to evolve, the DQI will no doubt be further refined and adapted. But its core aim of bringing empirical rigor to the study of discursive politics continues to be vital.

APPENDIX B

Appendix for "LegisGraph: A Graph-Based Method for Analyzing Legislative Debates Using Language Models"

B.1 Graph Generation Prompt Overview

This appendix section provides a more detailed overview of the graph generation prompt used in the LegisGraph approach. The prompt is designed to guide the large language model (LLM) in analyzing legislative debates and creating structured graph representations that capture the relationships between speakers, arguments, topics, and procedural elements. The prompt includes high-level instructions, synthetic examples, an output schema, and a chain-of-thought technique to enhance the accuracy and coherence of the generated graphs.

B.1.1 Complete Annotated Prompt

The complete annotated prompt is presented below, providing detailed instructions for the LLM on how to analyze legislative debates and generate Mermaid graph representations. The prompt includes high-level instructions, synthetic examples, an output schema, and guidance on using Mermaid syntax for graph visualization.

B.1.1.1 High Level Instructions

The high-level instructions provide an overview of the task and set expectations for the LLM. They outline the key elements to be extracted from the debate, including speakers, parties, topics, arguments, and procedural motions, and specify the desired output format. In this case, the instructions specify two types of outputs: a structured list of key elements and a Mermaid graph representation. The structured list of key elements is intended to elicit a comprehensive overview of the debate structure and dynamic (including speakers, topics,

subjects of conversation, arguments, and procedural actions) as well as the relationships between these elements, and a paragraph summarizing the key points of the debate. The Mermaid graph representation follows, and is intended to translate the previously extracted elements into a visual graph format that captures the relationships between speakers, arguments, topics, and procedural elements, as well as allowing for a straightforward visualization of the debate structure and aggregation across multiple debates.

```
# Debate Analysis and Graph Representation Instructions

You are an AI assistant specialized in analyzing legislative debates and
    creating structured graph representations. Your task is to analyze the
    provided raw text of a legislative debate, extract key information, and
    generate a Mermaid graph that represents the debate structure, including
    speakers, parties, topics, subjects, arguments, and procedural motions (
    including their outcomes).

## Analysis Instructions:
1. Parties: Identify all political parties mentioned in the debate.
  - Example: "The debate mentions the Liberal party and the Conservative party
      ."

2. Speakers: Identify all speakers and their party affiliations.
  - Example: "John Doe (Liberal), Jane Smith (Conservative), and Mark Johnson
      (Liberal) are the speakers in the debate."

3. Topics and Subjects:
  - Identify 1-3 broad Topics that the debate focuses on, selected from the
      following list:
    - Environmental Policy
    - Economic Policy
    - Healthcare
    - Education
    - Foreign Affairs
    - Defense
    - Social Policy
    - Justice and Law
    - Infrastructure
    - Technology and Innovation
    - Procedure
    - Agriculture and Rural Affairs
    - Trade and Commerce
    - Energy Policy
    - Cultural and Heritage Affairs
```

    - Urban Development
    - Transportation
    - Labor and Employment
  - Determine 2-4 more specific Subjects within each Topic that are discussed in the debate.
  - Example: "The debate focuses on two main Topics: 1) Environmental Policy and 2) Economic Policy. The specific Subjects discussed are: 1) Emissions regulations, 2) Carbon taxation, 3) Job market impact, and 4) Industrial competitiveness."

4. Debate Node: Create a list representing the entire debate, including key metadata such as date, session, and bill number (if applicable).

5. Procedural Motions:
  - Identify all formal procedural actions, including motions, amendments, and their outcomes.
  - For each motion, note the type (e.g., amendment, referral, extension of debate), who proposed it, the outcome (passed/failed), and which Subject(s) it affects.
  - Example: "An amendment was proposed by Mr. Lennox to extend the implementation timeline. The amendment passed, affecting the Subject of Renewable Energy Transition."

6. Arguments:
  - Identify the key arguments being made in the debate, and which speakers are making them.
  - Ensure that the argument labels are assertive statements.
  - Indicate which Subject(s) each argument addresses.
  - Example: "John Doe argues that 'Increasing carbon taxes will accelerate the transition to clean energy' (addresses Subjects: Carbon taxation, Renewable energy investments)."

7. Relationships:
  - Identify clear support or contradiction between arguments, and between arguments and subjects.
  - Inter-Argument Relationships:
    - Identify and describe the relationships between different arguments presented in the debate.
    - Consider how arguments support, contradict, modify, or address each other.
    - Provide specific examples of these relationships, referencing the speakers and speech IDs.

```
    - Example: "Argument A (Speaker 1, Speech ID: 2) directly contradicts
      Argument B (Speaker 3, Speech ID: 5) on the issue of tax increases."
  - Argument-Subject Relationships:
    - For each argument identified, specify which subject(s) it addresses.
    - Explain how the argument relates to or impacts the subject(s).
    - Provide examples of these relationships, referencing the speakers and
      speech IDs.
    - Example: "Argument C (Speaker 2, Speech ID: 4) addresses the subject of
      Renewable Energy Transition by proposing specific timelines for
      implementation."

8. Summary:
  - Generate a concise paragraph summarizing the key points of the debate,
    including the main topics discussed, the primary arguments presented,
    and any significant outcomes or decisions reached.
  - Example: "This debate centered on the proposed Renewable Energy Transition
     Act, focusing on the balance between environmental protection and
    economic stability. The main arguments revolved around the pace of
    implementation, with the Green Party advocating for rapid change and the
     Conservative Party expressing concerns about economic impacts. A
    compromise was reached through amendments extending the implementation
    timeline and introducing regular review periods. The debate highlighted
    the complex interplay between environmental urgency and economic
    considerations in crafting effective climate policy."
```

B.1.1.2 Mermaid Graph Instructions

```
## Mermaid Graph Creation Instructions:
1. Create nodes for the debate itself, each identified Topic, each party,
   argument, and procedural motion. Include speakers speaker and subject ONLY
    IF they are associated with an argument. If a speaker or subject is not
   associated with at least one argument, exclude it from the graph.

2. Use the provided class definitions for color-coding, including classes for
   procedural motions based on their outcome (passed or failed).

3. Use the following edge types:
  - Between Debate and Topics: "debate about" (solid line)
  - Between Subject and Debate: "discussed during" (solid line)
  - Between Party and Speaker: "party of" (solid line)
  - Between Speaker and Argument: "makes -- Speech ID "(solid line, annotated
    with the speech ID)
```

- Between Speaker and Procedural Motion: "moves -- Speech ID" (solid line, annotated with the speech ID)
- Between Argument and Subject: "addresses" (solid line)
- Between Procedural Motion and Subject: "affects" (solid line)
- Between Arguments: "supports" or "contradicts" (dotted line)

4. For every procedural motion:
   - Include an edge between the speaker who moved the motion and the motion node. This edge must be labeled with "moves -- Speech ID" where Speech ID is the corresponding speech identifier.
   - Use the class "motion_passed" or "motion_failed" to color-code the motion nodes based on their outcome.
   - Connect the motion node to the relevant Subject(s) with an edge labeled "affects".

5. Represent both inter-argument and argument-subject relationships in the graph:
   - Use solid lines with the label "addresses" to connect arguments to the subjects they relate to.
   - Use dotted lines with labels such as "supports", "contradicts", or "modifies" to show relationships between arguments.
   - Ensure that all identified relationships from the analysis stage are represented in the graph.

5. Ensure that every speaker included as a speaker node in the graph has at least one outgoing edge. This edge should connect to either an argument ("makes") or a motion ("moves").

6. Use clear, concise labels for all nodes. For the Topics, use the exact standardized terms from the provided list.

7. Organize the graph layout for readability: arrange speaker nodes and their connected argument nodes in rough chronological order based on the speech IDs.

8. Focus on key interactions.

9. Include metadata for the debate as a whole (date, session, bill number if applicable) within the Debate node label.

B.1.1.3 Synthetic Examples

As discussed above, synthetic examples are an essential component of the prompt generation process. They provide the LLM with a diverse set of cases to ensure that the model can handle a wide range of debate structures and complexities. The synthetic examples cover various scenarios, including simple debates with clear structures and complex debates with multiple interrelated topics and procedural elements. These examples serve as templates for the LLM, demonstrating the level of detail and structure expected in the output. In this case, there are three examples, each with a corresponding structured list and Mermaid graph representation. The first example is a straightforward debate with a single topic and clear arguments, while the second example involves multiple topics and procedural motions. The third example is a more complex debate with interrelated arguments and procedural actions, and is intended to provide a template to follow for more contentious debates.

````
## Example Debate:
```
Debate: Renewable Energy Transition Act
Date: 2023-06-15
Session: Summer Session
Bill: S.123

Speaker 1 (Green Party): Madam Speaker, I rise today to argue for the urgent
    passage of the Renewable Energy Transition Act. Climate change poses an
    existential threat to our planet, and we must act decisively. This bill
    proposes a rapid shift to renewable energy sources, coupled with
    significant investments in green technology. We cannot afford to delay any
     longer. (Speech ID: 1)

Speaker 2 (Conservative Party): While I appreciate the concern for our
    environment, I must express reservations about the pace and scale of the
    proposed changes. The immediate transition this bill demands could lead to
     job losses in traditional energy sectors and potentially destabilize our
    energy grid. We need a more measured approach that balances environmental
    concerns with economic stability. (Speech ID: 2)

Speaker 3 (Labor Party): I support the general direction of this bill but
    share some of the concerns raised by my conservative colleague. Perhaps we
     can find a middle ground? I propose we amend the timeline for
    implementation, allowing for a more gradual transition that protects
    workers in affected industries. This could include provisions for re-
    training programs and phased implementation of new standards. (Speech ID:
    3)
````
````

Speaker 1 (Green Party): I understand the concerns about job losses, but I must emphasize that the renewable energy sector will create numerous new jobs. Studies show that investments in green technology stimulate economic growth. The cost of inaction far outweighs the short-term economic adjustments we might face. (Speech ID: 4)

Speaker 4 (Conservative Party): I move to refer this bill to the Economic Affairs Committee for further study of its economic implications before we proceed with a vote. (Speech ID: 5)

[The motion to refer the bill to committee is voted on and fails.]

Speaker 3 (Labor Party): Given the failure of the motion to refer, I'd like to formally propose an amendment to extend the implementation timeline from 5 years to 8 years, with mandatory review points every two years. This would allow for a more gradual transition while still maintaining our commitment to renewable energy. (Speech ID: 6)

Speaker 1 (Green Party): While I believe we need to act faster, I'm willing to accept this amendment in the spirit of compromise and to ensure we make progress. However, I propose we include more aggressive targets for the first four years to ensure we don't lose momentum. (Speech ID: 7)

Speaker 2 (Conservative Party): I appreciate the attempt at compromise, but I still have significant reservations about the economic impact. Instead of mandating a transition, why don't we focus on incentivizing renewable energy adoption through tax breaks and research grants? This would encourage innovation without risking economic stability. (Speech ID: 8)

Speaker 5 (Green Party): I'd like to address the economic concerns raised by my colleagues. The renewable energy sector is one of the fastest-growing job markets globally. By leading in this transition, we're not just protecting our environment, we're positioning our nation at the forefront of a booming industry. This bill isn't just about environmental protection, it's about economic opportunity. (Speech ID: 9)

Speaker 6 (Labor Party): I move to extend the debate by an additional hour to allow for further discussion of the proposed amendment and its implications. (Speech ID: 10)

[The motion to extend the debate passes.]

```
Speaker 3 (Labor Party): I call for a vote on my proposed amendment to extend
   the implementation timeline. (Speech ID: 11)

[The amendment to extend the implementation timeline passes.]
```

Analysis:

1. Parties: The debate involves three parties: Green Party, Conservative Party
   , and Labor Party.

2. Speakers:
  - Speaker 1 (Green Party)
  - Speaker 2 (Conservative Party)
  - Speaker 3 (Labor Party)
  - Speaker 4 (Conservative Party)
  - Speaker 5 (Green Party)
  - Speaker 6 (Labor Party)

3. Topics and Subjects:
  - Topics:
    1. Environmental Policy
    2. Economic Policy
  - Subjects:
    1. Renewable Energy Transition
    2. Economic Impact
    3. Job Market Changes
    4. Green Technology Investment

4. Debate Node: Renewable Energy Transition Act, 2023-06-15, Summer Session,
   Bill S.123

5. Procedural Motions:
  - Motion to refer the bill to the Economic Affairs Committee (failed, Speech
      ID: 5)
  - Amendment to extend implementation timeline from 5 to 8 years (passed,
     Speech ID: 6)
  - Motion to extend debate by one hour (passed, Speech ID: 10)

6. Arguments:
  a. Urgent need for rapid shift to renewable energy (Speaker 1, Green Party,
     Speech ID: 1)

Quote: "Climate change poses an existential threat to our planet, and we must act decisively."

b. Concerns about pace and economic stability (Speaker 2, Conservative Party, Speech ID: 2)
Quote: "The immediate transition this bill demands could lead to job losses in traditional energy sectors and potentially destabilize our energy grid."

c. Proposal for gradual transition and worker protection (Speaker 3, Labor Party, Speech ID: 3)
Quote: "I propose we amend the timeline for implementation, allowing for a more gradual transition that protects workers in affected industries."

d. Renewable sector job creation and economic growth (Speaker 1, Green Party, Speech ID: 4)
Quote: "Studies show that investments in green technology stimulate economic growth."

e. Proposal for incentives instead of mandates (Speaker 2, Conservative Party, Speech ID: 8)
Quote: "Instead of mandating a transition, why don't we focus on incentivizing renewable energy adoption through tax breaks and research grants?"

f. Renewable energy as economic opportunity (Speaker 5, Green Party, Speech ID: 9)
Quote: "By leading in this transition, we're not just protecting our environment, we're positioning our nation at the forefront of a booming industry."

7. Relationships:
Inter-Argument Relationships:
a. Contradictions:
- Argument A "Urgent need for rapid shift to renewable energy" (Speaker 1, Green Party, Speech ID: 1) contradicts Argument B "Concerns about pace and economic stability" (Speaker 2, Conservative Party, Speech ID: 2).
- Argument E "Proposal for incentives instead of mandates" (Speaker 2, Conservative Party, Speech ID: 8) contradicts Argument A's call for urgent action.
b. Support:

- Argument D "Renewable sector job creation and economic growth" (Speaker 1, Green Party, Speech ID: 4) supports Argument A by providing economic justification for rapid transition.
- Argument F "Renewable energy as economic opportunity" (Speaker 5, Green Party, Speech ID: 9) supports Arguments A and D by further emphasizing economic benefits.

Argument-Subject Relationships:
- Arguments A and F address the Renewable Energy Transition subject by emphasizing the need for rapid change and the opportunities it presents.
- Arguments B and E address the Economic Impact subject by raising concerns about the potential negative effects of rapid transition.
- Arguments C and D address the Job Market Changes subject by discussing the potential for job losses and new job creation in the renewable sector.
- Argument D also addresses the Green Technology Investment subject by linking it to economic growth.

8. Summary:
This debate on the Renewable Energy Transition Act showcases the tension between environmental urgency and economic concerns. The Green Party advocates for rapid transition to renewable energy, citing climate change threats and economic opportunities. The Conservative Party expresses concerns about the pace of change and its potential impact on jobs and economic stability. The Labor Party attempts to bridge this divide by proposing a more gradual transition that protects workers. Key procedural elements include a failed attempt to refer the bill to committee, a successful amendment to extend the implementation timeline, and an extension of the debate itself. The discussion evolves from stark opposition to attempts at compromise, reflecting the complex nature of balancing environmental and economic priorities in energy policy.

Example Mermaid Graph:

```mermaid
graph TD
   %% Debate and Topics
   D[Debate: Renewable Energy Transition Act<br>2023-06-15, Summer Session, Bill S.123]:::debate
   T1[Topic: Environmental Policy]:::topic
   T2[Topic: Economic Policy]:::topic
   D -->|debate about| T1
   D -->|debate about| T2
```

```
%% Subjects
S1[Renewable Energy Transition]:::subject
S2[Economic Impact]:::subject
S3[Job Market Changes]:::subject
S4[Green Technology Investment]:::subject
S1 -->|discussed during| D
S2 -->|discussed during| D
S3 -->|discussed during| D
S4 -->|discussed during| D

%% Parties
P1[Green Party]:::party
P2[Conservative Party]:::party
P3[Labor Party]:::party

%% Speakers
Spk1[Speaker 1]:::speaker
Spk2[Speaker 2]:::speaker
Spk3[Speaker 3]:::speaker
Spk4[Speaker 4]:::speaker
Spk5[Speaker 5]:::speaker
Spk6[Speaker 6]:::speaker

%% Arguments
A1[Urgent need for rapid shift to renewable energy]:::argument
A2[Concerns about pace and economic stability]:::argument
A3[Proposal for gradual transition and worker protection]:::argument
A4[Renewable sector job creation and economic growth]:::argument
A5[Proposal for incentives instead of mandates]:::argument
A6[Renewable energy as economic opportunity]:::argument

%% Procedural Motions
M1[Refer bill to Economic Affairs Committee]:::motion_failed
M2[Extend implementation timeline]:::motion_passed
M3[Extend debate by one hour]:::motion_passed

%% Party-Speaker Relationships
P1 -->|party of| Spk1
P1 -->|party of| Spk5
P2 -->|party of| Spk2
P2 -->|party of| Spk4
P3 -->|party of| Spk3
```

```
P3 -->|party of| Spk6

%% Speaker-Argument Relationships
Spk1 -->|makes -- 1| A1
Spk2 -->|makes -- 2| A2
Spk3 -->|makes -- 3| A3
Spk1 -->|makes -- 4| A4
Spk2 -->|makes -- 8| A5
Spk5 -->|makes -- 9| A6

%% Speaker-Motion Relationships
Spk4 -->|moves -- 5| M1
Spk3 -->|moves -- 6, 11| M2
Spk6 -->|moves -- 10| M3

%% Argument-Subject Relationships
A1 -->|addresses| S1
A2 -->|addresses| S2
A3 -->|addresses| S3
A4 -->|addresses| S3
A4 -->|addresses| S4
A5 -->|addresses| S2
A6 -->|addresses| S2
A6 -->|addresses| S3

%% Motion-Subject Relationships
M1 -->|affects| S2
M2 -->|affects| S1
M3 -->|affects| S1

%% Inter-Argument Relationships
A2 -.->|contradicts| A1
A3 -.->|modifies| A1
A4 -.->|supports| A1
A4 -.->|contradicts| A2
A5 -.->|contradicts| A1
A6 -.->|supports| A1
A6 -.->|contradicts| A2

%% Styling
classDef debate fill:#C7CEEA,stroke:#4B0082,stroke-width:2px;
classDef topic fill:#FF9999,stroke:#FF0000,stroke-width:2px;
classDef subject fill:#A9DFBF,stroke:#27AE60,stroke-width:2px;
```

```
   classDef party fill:#8E44AD,stroke:#8E44AD,stroke-width:2px;
   classDef speaker fill:#AED6F1,stroke:#3498DB,stroke-width:2px;
   classDef argument fill:#FAD7A0,stroke:#F39C12,stroke-width:2px;
   classDef motion_passed fill:#4CAF50,stroke:#4CAF50,stroke-width:2px;
   classDef motion_failed fill:#F44336,stroke:#F44336,stroke-width:2px;
```

This example demonstrates:
1. Multiple Topics (Environmental Policy and Economic Policy) linked to the Debate node with "debate about" edges.
2. Multiple Subjects linked to the Debate node with "discussed during" edges.
3. Multiple parties and speakers with clear affiliations.
4. A variety of arguments addressing different aspects of the debate.
5. Procedural motions, including both passed and failed motions, with their outcomes represented by color-coding.
6. Consistent representation of how both arguments and motions affect subjects, using "addresses" and "affects" edges respectively.
7. Complex relationships between arguments, including support, contradiction, and modification.
8. Clear connections between arguments/motions and the subjects they address/affect.
9. Temporal order of speeches represented by Speech IDs.

Example Debate: Education Reform Act

```
Debate: Education Reform Act
Date: 2023-09-10
Session: Fall Session
Bill: H.R. 456

Speaker 1 (Progressive Party): Mr. Speaker, I rise today to introduce the Education Reform Act. This comprehensive bill aims to address the growing inequalities in our education system, modernize our curriculum, and better prepare our students for the challenges of the 21st century. Key provisions include increased funding for underprivileged schools, integration of technology in classrooms, and a revised standardized testing approach. (Speech ID: 1)

Speaker 2 (Conservative Party): While I appreciate my colleague's intent, I have serious concerns about the scope and cost of this bill. The proposed increase in federal oversight could undermine local control of education. Furthermore, the financial burden on taxpayers would be substantial. We need to focus on improving efficiency within the current system rather
```

than expanding it. (Speech ID: 2)

Speaker 3 (Centrist Party): I believe there's merit to both arguments we've heard. Perhaps we can find a middle ground. I propose an amendment to phase in the funding increases over a longer period, allowing for a more gradual implementation and assessment of the program's effectiveness. (Speech ID: 3)

Speaker 4 (Progressive Party): I second the motion for the amendment. While I believe urgent action is necessary, a phased approach could help us gain broader support and allow for necessary adjustments as we proceed. (Speech ID: 4)

[The amendment for phased implementation is voted on and passes.]

Speaker 5 (Conservative Party): Even with this amendment, I'm not convinced. The bill still doesn't address the root issues of parental involvement and teacher quality. I move to refer this bill to the Education Committee for further study and revision. (Speech ID: 5)

Speaker 1 (Progressive Party): I must object to this motion. We've debated these issues extensively, and further delay only hurts our students. The amendment already provides for a measured approach. Let's not use committee referral as a tactic to stall much-needed reform. (Speech ID: 6)

[The motion to refer to committee is voted on and fails.]

Speaker 6 (Centrist Party): Given the complexity of this issue, I move to extend the debate by two hours to allow for a more thorough discussion of the bill's provisions, particularly in light of the recently passed amendment. (Speech ID: 7)

[The motion to extend debate passes.]

Speaker 2 (Conservative Party): Thank you for extending the debate. I'd like to address the issue of standardized testing. While I agree our current approach needs revision, completely overhauling it could lead to a lack of accountability. Instead, why don't we consider a hybrid system that maintains some standardized elements while allowing for more flexible, localized assessments? (Speech ID: 8)

Speaker 1 (Progressive Party): I appreciate the constructive suggestion. Perhaps we can find common ground here. I propose we amend the

standardized testing section to include this hybrid approach, maintaining national standards while allowing for local flexibility. (Speech ID: 9)

Speaker 3 (Centrist Party): I second this motion. This balanced approach to testing could address concerns from both sides while still moving us forward. (Speech ID: 10)

[The amendment for a hybrid testing approach is voted on and passes.]

Speaker 5 (Conservative Party): While I still have reservations about the overall bill, I must admit that these amendments have addressed some of my key concerns. I'm particularly pleased with the hybrid testing approach. However, I remain worried about the potential for federal overreach in local education matters. (Speech ID: 11)

Speaker 1 (Progressive Party): I understand those concerns. Let me clarify that this bill isn't about federal control, but about providing resources and guidelines to ensure quality education across the nation. Local districts will still have significant autonomy in implementing these changes. (Speech ID: 12)

Speaker 4 (Progressive Party): Colleagues, as we near the end of this extended debate, I urge you to consider the transformative potential of this bill. With the amendments we've passed, we've created a more balanced approach that addresses concerns from all sides. This is our opportunity to make a real difference in the lives of students across the country. (Speech ID: 13)

[The final vote on the amended Education Reform Act is called.]
```

## Analysis:

1. Parties: The debate involves three parties: Progressive Party, Conservative Party, and Centrist Party.

2. Speakers:
   - Speaker 1 (Progressive Party)
   - Speaker 2 (Conservative Party)
   - Speaker 3 (Centrist Party)
   - Speaker 4 (Progressive Party)
   - Speaker 5 (Conservative Party)
   - Speaker 6 (Centrist Party)
```

3. Topics and Subjects:
  - Topics:
    1. Education
    2. Social Policy
  - Subjects:
    1. Education Funding
    2. Curriculum Modernization
    3. Standardized Testing
    4. Federal vs Local Control

4. Debate Node: Education Reform Act, 2023-09-10, Fall Session, Bill H.R. 456

5. Procedural Motions:
  - Amendment for phased implementation (passed, Speech ID: 3)
  - Motion to refer to Education Committee (failed, Speech ID: 5)
  - Motion to extend debate by two hours (passed, Speech ID: 7)
  - Amendment for hybrid testing approach (passed, Speech ID: 9)

6. Arguments:
  a. Comprehensive reform needed to address inequalities (Speaker 1, Progressive Party, Speech ID: 1)
    Quote: "This comprehensive bill aims to address the growing inequalities in our education system, modernize our curriculum, and better prepare our students for the challenges of the 21st century."

  b. Concerns about scope, cost, and federal oversight (Speaker 2, Conservative Party, Speech ID: 2)
    Quote: "The proposed increase in federal oversight could undermine local control of education. Furthermore, the financial burden on taxpayers would be substantial."

  c. Phased implementation for gradual change (Speaker 3, Centrist Party, Speech ID: 3)
    Quote: "I propose an amendment to phase in the funding increases over a longer period, allowing for a more gradual implementation and assessment of the program's effectiveness."

  d. Focus on parental involvement and teacher quality (Speaker 5, Conservative Party, Speech ID: 5)
    Quote: "The bill still doesn't address the root issues of parental involvement and teacher quality."

e. Hybrid approach to standardized testing (Speaker 2, Conservative Party, Speech ID: 8)
Quote: "Why don't we consider a hybrid system that maintains some standardized elements while allowing for more flexible, localized assessments?"

f. Bill provides resources without overreaching (Speaker 1, Progressive Party, Speech ID: 12)
Quote: "This bill isn't about federal control, but about providing resources and guidelines to ensure quality education across the nation. Local districts will still have significant autonomy in implementing these changes."

7. Relationships:
Inter-Argument Relationships:
a. Contradictions:
- Argument A "Comprehensive reform needed to address inequalities" (Speaker 1, Progressive Party, Speech ID: 1) contradicts Argument B "Concerns about scope, cost, and federal oversight" (Speaker 2, Conservative Party, Speech ID: 2).
- Argument D "Focus on parental involvement and teacher quality" (Speaker 5, Conservative Party, Speech ID: 5) contradicts Argument A's comprehensive approach by suggesting a more focused reform.

b. Support:
- Argument F "Bill provides resources without overreaching" (Speaker 1, Progressive Party, Speech ID: 12) supports Argument A by clarifying the intent and scope of the proposed reforms.
- Argument E "Hybrid approach to standardized testing" (Speaker 2, Conservative Party, Speech ID: 8) partially supports Argument B by addressing concerns about federal oversight in a specific area.

Argument-Subject Relationships:
- Argument A addresses Education Funding and Curriculum Modernization by proposing comprehensive reforms.
- Argument B addresses Education Funding and Federal vs Local Control by raising concerns about costs and oversight.
- Argument C addresses Education Funding by proposing a phased implementation approach.
- Argument D addresses Curriculum Modernization by focusing on teacher quality and parental involvement.
- Argument E addresses Standardized Testing by proposing a hybrid approach.
- Argument F addresses Federal vs Local Control by clarifying the bill's intent regarding local autonomy.

8. Summary:
   This debate on the Education Reform Act highlights the complex interplay between the desire for comprehensive education reform and concerns about federal overreach and cost. The Progressive Party advocates for wide-ranging changes to address inequalities and modernize the curriculum, while the Conservative Party expresses reservations about the scope, cost, and potential loss of local control. The Centrist Party attempts to bridge these divides by proposing compromises such as phased implementation. Key outcomes include the passage of amendments for phased implementation and a hybrid testing approach, demonstrating a move towards a more balanced bill. The debate underscores the challenges in crafting education policy that addresses systemic issues while respecting local autonomy and managing costs.

Mermaid Graph Representation:

```mermaid
graph TD
   %% Debate and Topics
   D[Debate: Education Reform Act<br>2023-09-10, Fall Session, Bill H.R. 456]:::debate
   T1[Topic: Education]:::topic
   T2[Topic: Social Policy]:::topic
   D -->|debate about| T1
   D -->|debate about| T2

   %% Subjects
   S1[Education Funding]:::subject
   S2[Curriculum Modernization]:::subject
   S3[Standardized Testing]:::subject
   S4[Federal vs Local Control]:::subject
   S1 -->|discussed during| D
   S2 -->|discussed during| D
   S3 -->|discussed during| D
   S4 -->|discussed during| D

   %% Parties
   P1[Progressive Party]:::party
   P2[Conservative Party]:::party
   P3[Centrist Party]:::party

   %% Speakers
```

```
Spk1[Speaker 1]:::speaker
Spk2[Speaker 2]:::speaker
Spk3[Speaker 3]:::speaker
Spk4[Speaker 4]:::speaker
Spk5[Speaker 5]:::speaker
Spk6[Speaker 6]:::speaker

%% Arguments
A1[Comprehensive reform needed to address inequalities]:::argument
A2[Concerns about scope, cost, and federal oversight]:::argument
A3[Phased implementation for gradual change]:::argument
A4[Focus on parental involvement and teacher quality]:::argument
A5[Hybrid approach to standardized testing]:::argument
A6[Bill provides resources without overreaching]:::argument

%% Procedural Motions
M1[Phased implementation amendment]:::motion_passed
M2[Refer to Education Committee]:::motion_failed
M3[Extend debate by two hours]:::motion_passed
M4[Hybrid testing approach amendment]:::motion_passed

%% Party-Speaker Relationships
P1 -->|party of| Spk1
P1 -->|party of| Spk4
P2 -->|party of| Spk2
P2 -->|party of| Spk5
P3 -->|party of| Spk3
P3 -->|party of| Spk6

%% Speaker-Argument Relationships
Spk1 -->|makes -- 1| A1
Spk2 -->|makes -- 2| A2
Spk3 -->|makes -- 3| A3
Spk5 -->|makes -- 5| A4
Spk2 -->|makes -- 8| A5
Spk1 -->|makes -- 12| A6

%% Speaker-Motion Relationships
Spk3 -->|moves -- 3| M1
Spk5 -->|moves -- 5| M2
Spk6 -->|moves -- 7| M3
Spk1 -->|moves -- 9| M4
```

```
   %% Argument-Subject Relationships
   A1 -->|addresses| S1
   A1 -->|addresses| S2
   A2 -->|addresses| S1
   A2 -->|addresses| S4
   A3 -->|addresses| S1
   A4 -->|addresses| S2
   A5 -->|addresses| S3
   A6 -->|addresses| S4

   %% Motion-Subject Relationships
   M1 -->|affects| S1
   M2 -->|affects| S4
   M3 -->|affects| S1
   M3 -->|affects| S2
   M3 -->|affects| S3
   M3 -->|affects| S4
   M4 -->|affects| S3

   %% Inter-Argument Relationships
   A2 -.->|contradicts| A1
   A3 -.->|modifies| A1
   A3 -.->|addresses| A2
   A4 -.->|contradicts| A1
   A5 -.->|modifies| A1
   A5 -.->|addresses| A2
   A6 -.->|addresses| A2

   %% Styling
   classDef debate fill:#C7CEEA,stroke:#4B0082,stroke-width:2px;
   classDef topic fill:#FF9999,stroke:#FF0000,stroke-width:2px;
   classDef subject fill:#A9DFBF,stroke:#27AE60,stroke-width:2px;
   classDef party fill:#8E44AD,stroke:#8E44AD,stroke-width:2px;
   classDef speaker fill:#AED6F1,stroke:#3498DB,stroke-width:2px;
   classDef argument fill:#FAD7A0,stroke:#F39C12,stroke-width:2px;
   classDef motion_passed fill:#4CAF50,stroke:#4CAF50,stroke-width:2px;
   classDef motion_failed fill:#F44336,stroke:#F44336,stroke-width:2px;
```

This example demonstrates:

1. A different topic (Education Reform) with its own set of subjects and arguments.

2. More complex procedural elements, including multiple amendments and a failed motion to refer to committee.
3. A wider range of party involvement, showcasing how centrist positions can bridge opposing views.
4. Evolution of arguments throughout the debate, with initial positions being modified in response to others' concerns.
5. How a single motion (extending the debate) can affect multiple subjects.
6. The interplay between arguments and motions, showing how procedural actions can shape the substance of the debate.
7. A more nuanced set of inter-argument relationships, including modifications and addressing concerns, not just support or contradiction.

This example, contrasted with the previous one, helps illustrate the variety of ways debates can unfold and be represented in the knowledge graph format. It shows how the system can adapt to different topics, argument structures, and procedural complexities while maintaining a consistent representation approach.

Example Debate: National Security Enhancement Act

```
Debate: National Security Enhancement Act
Date: 2023-11-15
Session: Winter Emergency Session
Bill: S.789

Speaker 1 (Nationalist Party): Madam Speaker, I rise today to introduce the National Security Enhancement Act. In these perilous times, with threats lurking around every corner, we must take decisive action to protect our great nation. This bill will strengthen our borders, expand surveillance capabilities, and give our law enforcement the tools they need to keep us safe. Those who oppose this bill are nothing short of traitors to our country. (Speech ID: 1)

Speaker 2 (Liberty Party): The gentleman from the Nationalist Party speaks of traitors? That's rich coming from someone who seems intent on dismantling the very freedoms this nation was built upon! This bill isn't about security, it's about control. It's a thinly veiled attempt to turn our country into a police state. I wonder, did my esteemed colleague even read the Constitution before drafting this monstrosity? (Speech ID: 2)

Speaker 1 (Nationalist Party): How dare you question my patriotism! I've served this country while you were probably still learning your ABCs. This bill is about protecting the very freedoms you claim to cherish. Without
```

security, there is no liberty. But I wouldn't expect someone from the "Liberty" Party to understand such nuances. You're too busy coddling criminals and terrorists! (Speech ID: 3)

Speaker 3 (Centrist Alliance): Colleagues, please! Can we bring this debate back to the actual content of the bill? This kind of personal attacks does nothing to address the real issues at hand. (Speech ID: 4)

Speaker 4 (Liberty Party): Oh, I'm sorry, is the truth too uncomfortable for some? Fine, let's talk about the "content" of this bill. Section 7, subsection C - it allows for warrantless searches based on the flimsiest of pretexts. How's that for "enhancing security"? It's more like enhancing totalitarianism! (Speech ID: 5)

Speaker 1 (Nationalist Party): If you have nothing to hide, you have nothing to fear. Only guilty people worry about searches. But let me remind you all of the very real threats we face. Just last week, in a small town much like the ones many of us represent, a tragedy was narrowly averted... [Proceeds to give a 30-minute account of various crimes and potential terrorist activities around the country, many only tangentially related to the bill at hand] ...and that's why we need this bill. The blood of innocent citizens will be on the hands of anyone who votes against it. (Speech ID: 6)

Speaker 5 (Liberty Party): Point of order, Madam Speaker! The gentleman has been speaking for over half an hour, and most of what he's said has nothing to do with the bill. This is a blatant attempt to waste time and prevent meaningful debate. (Speech ID: 7)

Speaker 6 (Nationalist Party): Madam Speaker, I move to extend the debate by six hours. Clearly, my colleagues need more time to understand the critical importance of this bill. (Speech ID: 8)

[The motion to extend debate passes narrowly]

Speaker 2 (Liberty Party): Six more hours of this fearmongering? Fine. Let me tell you about the real threat to our nation. It's not some bogeyman terrorist, it's the erosion of our civil liberties by opportunistic politicians. [Launches into a lengthy historical lecture about various authoritarian regimes and their rise to power] ...and that's where this bill is taking us. Wake up, America! (Speech ID: 9)

Speaker 7 (Centrist Alliance): I move to amend the bill to include stronger

oversight provisions for any expanded surveillance activities. (Speech ID: 10)

Speaker 1 (Nationalist Party): Oversight? You want to tie the hands of our brave law enforcement officers with more red tape? Absolutely not! I object to this amendment in the strongest possible terms. (Speech ID: 11)

Speaker 8 (Liberty Party): Oh, now the gentleman objects to constitutional protections. Why am I not surprised? Your idea of security would make our Founding Fathers roll in their graves. (Speech ID: 12)

[The amendment for stronger oversight fails]

Speaker 9 (Nationalist Party): I move that we invoke cloture and bring this bill to an immediate vote. We've heard enough obstruction from the opposition. It's time to act! (Speech ID: 13)

Speaker 2 (Liberty Party): Cloture? Ha! Now who's trying to stifle debate? But you know what, go ahead. Force this vote. Every citizen will know exactly who stood for freedom and who cowered in fear. (Speech ID: 14)

[The motion for cloture fails]

Speaker 3 (Centrist Alliance): Colleagues, we're getting nowhere. I propose we table this bill and refer it back to committee for substantial revisions. Clearly, there are deep disagreements that need to be addressed before we can move forward productively. (Speech ID: 15)

Speaker 1 (Nationalist Party): Table it? Never! Every moment we delay puts lives at risk. I won't have that blood on my hands, even if you're all willing to. We'll stay here all night if we have to, but we're voting on this bill! (Speech ID: 16)

[Debate continues with similar tone...]
```
## Analysis:

1. Parties: The debate involves three parties: Nationalist Party, Liberty Party, and Centrist Alliance.

2. Speakers:
  - Speaker 1 (Nationalist Party)
  - Speaker 2 (Liberty Party)
```

- Speaker 3 (Centrist Alliance)
- Speaker 4 (Liberty Party)
- Speaker 5 (Liberty Party)
- Speaker 6 (Nationalist Party)
- Speaker 7 (Centrist Alliance)
- Speaker 8 (Liberty Party)
- Speaker 9 (Nationalist Party)

3. Topics and Subjects:
   - Topics:
     1. Defense
     2. Justice and Law
   - Subjects:
     1. Border Security
     2. Surveillance Powers
     3. Civil Liberties
     4. Law Enforcement Authority

4. Debate Node: National Security Enhancement Act, 2023-11-15, Winter Emergency Session, Bill S.789

5. Procedural Motions:
   - Motion to extend debate by six hours (passed, Speech ID: 8)
   - Amendment for stronger oversight on surveillance activities (failed, Speech ID: 10)
   - Motion to invoke cloture for immediate vote (failed, Speech ID: 13)
   - Motion to table bill and refer back to committee (failed, Speech ID: 15)

6. Arguments:
   a. Enhanced security measures are crucial for national safety (Speaker 1, Nationalist Party, Speech ID: 1)
      Quote: "This bill will strengthen our borders, expand surveillance capabilities, and give our law enforcement the tools they need to keep us safe."

   b. Bill threatens civil liberties and constitutional rights (Speaker 2, Liberty Party, Speech ID: 2)
      Quote: "This bill isn't about security, it's about control. It's a thinly veiled attempt to turn our country into a police state."

   c. Warrantless searches are necessary for effective security (Speaker 1, Nationalist Party, Speech ID: 6)
      Quote: "If you have nothing to hide, you have nothing to fear. Only

guilty people worry about searches."

d. Historical examples show dangers of authoritarian overreach (Speaker 2, Liberty Party, Speech ID: 9)
Quote: "Let me tell you about the real threat to our nation. It's not some bogeyman terrorist, it's the erosion of our civil liberties by opportunistic politicians."

e. Need for stronger oversight on surveillance activities (Speaker 7, Centrist Alliance, Speech ID: 10)
Quote: "I move to amend the bill to include stronger oversight provisions for any expanded surveillance activities."

f. Delay in passing bill puts lives at risk (Speaker 1, Nationalist Party, Speech ID: 16)
Quote: "Every moment we delay puts lives at risk. I won't have that blood on my hands, even if you're all willing to."

7. Relationships:
Inter-Argument Relationships:
a. Contradictions:
- Argument A "Enhanced security measures are crucial for national safety" (Speaker 1, Nationalist Party, Speech ID: 1) contradicts Argument B "Bill threatens civil liberties and constitutional rights" (Speaker 2, Liberty Party, Speech ID: 2).
- Argument D "Historical examples show dangers of authoritarian overreach" (Speaker 2, Liberty Party, Speech ID: 9) contradicts Argument C "Warrantless searches are necessary for effective security" (Speaker 1, Nationalist Party, Speech ID: 6).

b. Support:
- Argument C "Warrantless searches are necessary for effective security" (Speaker 1, Nationalist Party, Speech ID: 6) supports Argument A by providing a specific example of enhanced security measures.
- Argument F "Delay in passing bill puts lives at risk" (Speaker 1, Nationalist Party, Speech ID: 16) supports Argument A by emphasizing the urgency of implementing security measures.

Argument-Subject Relationships:
- Arguments A and F address Border Security and Law Enforcement Authority by emphasizing the need for enhanced security measures.
- Argument B addresses Civil Liberties by highlighting concerns about constitutional rights.
- Argument C addresses Surveillance Powers and Law Enforcement Authority by

justifying warrantless searches.
  - Argument D addresses Civil Liberties by drawing parallels to historical authoritarian regimes.
  - Argument E addresses Surveillance Powers by proposing stronger oversight mechanisms.

8. Summary:
  This debate on the National Security Enhancement Act reveals a deeply polarized discussion about the balance between national security and civil liberties. The Nationalist Party strongly advocates for expanded security measures, including increased border security and surveillance powers, arguing that these are crucial for public safety. The Liberty Party vehemently opposes the bill, viewing it as a threat to constitutional rights and a step towards authoritarianism. The Centrist Alliance attempts to bridge the gap by proposing stronger oversight provisions, but their efforts are largely unsuccessful. The debate is marked by emotional arguments, personal attacks, and procedural maneuvering, including attempts to extend the debate, force an immediate vote, and refer the bill back to committee. Despite these efforts, no clear resolution is reached, highlighting the challenges in finding common ground on issues of national security and civil liberties.

Mermaid Graph Representation:

```mermaid
graph TD
   %% Debate and Topics
   D[Debate: National Security Enhancement Act<br>2023-11-15, Winter Emergency Session, Bill S.789]:::debate
   T1[Topic: Defense]:::topic
   T2[Topic: Justice and Law]:::topic
   D -->|debate about| T1
   D -->|debate about| T2

   %% Subjects
   S1[Border Security]:::subject
   S2[Surveillance Powers]:::subject
   S3[Civil Liberties]:::subject
   S4[Law Enforcement Authority]:::subject
   S1 -->|discussed during| D
   S2 -->|discussed during| D
   S3 -->|discussed during| D
   S4 -->|discussed during| D
```

```
%% Parties
P1[Nationalist Party]:::party
P2[Liberty Party]:::party
P3[Centrist Alliance]:::party

%% Speakers
Spk1[Speaker 1]:::speaker
Spk2[Speaker 2]:::speaker
Spk3[Speaker 3]:::speaker
Spk4[Speaker 4]:::speaker
Spk5[Speaker 5]:::speaker
Spk6[Speaker 6]:::speaker
Spk7[Speaker 7]:::speaker
Spk8[Speaker 8]:::speaker
Spk9[Speaker 9]:::speaker

%% Arguments
A1[Enhanced security measures are crucial for national safety]:::argument
A2[Bill threatens civil liberties and constitutional rights]:::argument
A3[Warrantless searches are necessary for effective security]:::argument
A4[Historical examples show dangers of authoritarian overreach]:::argument
A5[Need for stronger oversight on surveillance activities]:::argument
A6[Delay in passing bill puts lives at risk]:::argument

%% Procedural Motions
M1[Extend debate by six hours]:::motion_passed
M2[Amend bill for stronger oversight]:::motion_failed
M3[Invoke cloture for immediate vote]:::motion_failed
M4[Table bill and refer back to committee]:::motion_failed

%% Party-Speaker Relationships
P1 -->|party of| Spk1
P1 -->|party of| Spk6
P1 -->|party of| Spk9
P2 -->|party of| Spk2
P2 -->|party of| Spk4
P2 -->|party of| Spk5
P2 -->|party of| Spk8
P3 -->|party of| Spk3
P3 -->|party of| Spk7

%% Speaker-Argument Relationships
```

```
Spk1 -->|makes -- 1, 3, 6| A1
Spk2 -->|makes -- 2, 9| A2
Spk1 -->|makes -- 6| A3
Spk2 -->|makes -- 9| A4
Spk7 -->|makes -- 10| A5
Spk1 -->|makes -- 16| A6

%% Speaker-Motion Relationships
Spk6 -->|moves -- 8| M1
Spk7 -->|moves -- 10| M2
Spk9 -->|moves -- 13| M3
Spk3 -->|moves -- 15| M4

%% Argument-Subject Relationships
A1 -->|addresses| S1
A1 -->|addresses| S2
A1 -->|addresses| S4
A2 -->|addresses| S3
A3 -->|addresses| S2
A3 -->|addresses| S4
A4 -->|addresses| S3
A5 -->|addresses| S2
A6 -->|addresses| S1
A6 -->|addresses| S4

%% Motion-Subject Relationships
M1 -->|affects| S1
M1 -->|affects| S2
M1 -->|affects| S3
M1 -->|affects| S4
M2 -->|affects| S2
M2 -->|affects| S3
M3 -->|affects| S1
M3 -->|affects| S2
M3 -->|affects| S3
M3 -->|affects| S4
M4 -->|affects| S1
M4 -->|affects| S2
M4 -->|affects| S3
M4 -->|affects| S4

%% Inter-Argument Relationships
A2 -.->|contradicts| A1
```

```
   A3 -.->|supports| A1
   A3 -.->|contradicts| A2
   A4 -.->|contradicts| A1
   A4 -.->|supports| A2
   A5 -.->|modifies| A1
   A5 -.->|addresses| A2
   A6 -.->|supports| A1
   A6 -.->|contradicts| A2

   %% Styling
   classDef debate fill:#C7CEEA,stroke:#4B0082,stroke-width:2px;
   classDef topic fill:#FF9999,stroke:#FF0000,stroke-width:2px;
   classDef subject fill:#A9DFBF,stroke:#27AE60,stroke-width:2px;
   classDef party fill:#8E44AD,stroke:#8E44AD,stroke-width:2px;
   classDef speaker fill:#AED6F1,stroke:#3498DB,stroke-width:2px;
   classDef argument fill:#FAD7A0,stroke:#F39C12,stroke-width:2px;
   classDef motion_passed fill:#4CAF50,stroke:#4CAF50,stroke-width:2px;
   classDef motion_failed fill:#F44336,stroke:#F44336,stroke-width:2px;
```

This example demonstrates:

1. A highly contentious debate on a sensitive topic (national security vs. civil liberties).
2. Aggressive and sarcastic exchanges between speakers, particularly from the Nationalist and Liberty parties.
3. Long speeches that sometimes stray from the main topic (e.g., Speaker 1's 30-minute account of various crimes).
4. Attempts at obstruction through extended speeches and procedural motions.
5. Multiple failed motions, showing the back-and-forth nature of the debate.
6. The challenge of maintaining focus on the bill's content amid personal attacks and emotional arguments.
7. How the graph can still represent the core arguments and procedural elements even in a chaotic debate.
8. The way different arguments address multiple subjects, reflecting the complex nature of the issue.
9. How motions like extending the debate affect all subjects, representing their broad impact on the discussion.

B.1.1.4 Closing Instructions

The closing instructions provide the annotator with a summary of the task and the steps to follow. It includes a reminder to analyze the debate transcript, generate the corresponding

mermaid graph, and adhere to the guidelines for choosing topics. It also instructs the annotator on how to respond if the input is not a debate transcript.

```
Here's the debate transcript to analyze:
{debate_transcript}

As instructed, first generate an analysis of the debate, then generate the
    corresponding mermaid graph. Remember that when choosing the topics of the
     debate, you should draw only from the list of validated topics referred
    to above, and not from the topic label from the debate itself. For example
    , if the debate has "Main Topic: THE IMPERIAL CONFERENCE", you should
    choose the appropriate topic(s) from the provided list (in this case, it
    might include Foreign Affairs, Defense, and potentially Trade and Commerce
    ).

If you are provided with something else instead a debate transcript (i.e. a
    list of expenditures or an index), simply respond with "This is not a
    debate, and I cannot annotate it as requested".
```

B.1.1.5 Example of a Debate to be Annotated

This section provides an example of a real debate transcript that the annotator would receive for analysis. The debate includes information about the date, parliament, session, main topic, subtopics, number of speeches, and total number of words. It also lists the speakers, their positions, parties, speech IDs, and the text of their speeches.

```
Debate Date: 1939-04-04 00:00:00
Parliament: Thirty-Sixth Parliament
Session: 1st
Main Topic: ABOLITION OF PENSION APPEAL BOARD AND EXTENSION OF TIME FOR
    PENSION APPLICATIONS
Subtopic: none
Subsubtopic: none
Number of Speeches: 7
Total Number of Words: 409

-----------------------

Speaker: Charles Gavan Power
Position: Minister of Pensions and National Health
Party: Liberal
Speech ID: 1183987
Speech Text: Section 16 stands.
On section 17-Sittings of appeal boards.
```

Speaker: Howard Charles Green
Position: none
Party: Conservative (1867-1942)
Speech ID: 1183988
Speech Text: As I read the new section
55 of the act, it would be possible for the members of the appeal boards to have sat as members of the commission when a claim was being dealt with on the first or second hearing. Would it not be well to have some provision prohibiting a pension commissioner who sat on the original application for pension, or on the second hearing, from sitting on an appeal board hearing that same case? I am certain that if the act allows the same man to sit both on the original court and on an appeal board, there will be great dissatisfaction among applicants for pension. That should be guarded against.

Speaker: Charles Gavan Power
Position: Minister of Pensions and National Health
Party: Liberal
Speech ID: 1183989
Speech Text: I think the point raised by my hon. friend is well taken. I should imagine, however, that the chairman of the commission would see to it that the same man who sat on the original claim would not sit on an appeal board hearing the same case. That is the common sense of it; I do not know that it is necessary to have special provision in the legislation . I am however, willing to let the section stand and go into it further.

Speaker: Howard Charles Green
Position: none
Party: Conservative (1867-1942)
Speech ID: 1183990
Speech Text: I think there is a similar provision in the act at the present time, is there not? I think it is important.

Speaker: Charles Grant MacNeil

```
Position: none
Party: Co-operative Commonwealth Federation (C.C.F.)
Speech ID: 1183994
Speech Text: It says that a case can be opened up only by leave of the appeal
   board of the commission. Subsection 4 states:
An application based upon any error in such decision or in any decision of the
    court, by reason of evidence not having been presented or otherwise, may
   be entertained by the commission with the leave of an appeal board of the
   commission . . . and such appeal board shall have jurisdiction to grant
   leave in any case in which it appears proper to grant it.
Will that mean a special appeal board? What has the minister in mind in regard
    to regulations governing such appeals?

----------------------

Speaker: Charles Grant MacNeil
Position: none
Party: Co-operative Commonwealth Federation (C.C.F.)
Speech ID: 1183992
Speech Text: Would the minister explain the conditions under which cases may
   be re-opened, and, where the applicant or appellant has evidence of new
   and substantial character that should be considered by the appeal board,
   what exactly are the conditions under which that may be brought before the
    appeal board?

----------------------

Speaker: Charles Gavan Power
Position: Minister of Pensions and National Health
Party: Liberal
Speech ID: 1183993
Speech Text: I take it the conditions are exactly as they stand to-day. There
   is no change in the act as far as that is concerned.
```

B.1.1.6 Example Output

This section shows the complete output generated by the annotator based on the provided debate transcript. It includes the analysis of the debate, the mermaid graph representation, and the summary of the debate. The output is structured to provide a clear and detailed overview of the debate's key elements, including parties, speakers, topics, subjects, arguments, relationships, and procedural motions.

```
## Analysis:
```

1. **Parties**: The debate involves three parties: Liberal, Conservative (1867-1942), and Co-operative Commonwealth Federation (C.C.F.).

2. **Speakers**:
   - Charles Gavan Power (Liberal)
   - Howard Charles Green (Conservative)
   - Charles Grant MacNeil (C.C.F.)

3. **Topics and Subjects**:
   - **Topics**:
     1. Justice and Law
     2. Social Policy
   - **Subjects**:
     1. Pension Application Process
     2. Appeal Board Procedures
     3. Regulations on Pension Claims

4. **Debate Node**: Abolition of Pension Appeal Board and Extension of Time for Pension Applications, 1939-04-04, Thirty-Sixth Parliament, 1st Session .

5. **Procedural Motions**: There are no formal procedural motions identified in the debate transcript.

6. **Arguments**:
   - **Argument A**: Charles Gavan Power asserts that there is no need for specific provisions regarding the appeal board as common sense dictates that the same individual should not sit on both the original claim and appeal board (Speaker 1, Speech ID: 1183989).
   - **Argument B**: Howard Charles Green raises concerns about potential dissatisfaction among pension applicants if the same commissioner sits on both the original application and the appeal board (Speaker 2, Speech ID: 1183988).
   - **Argument C**: Charles Grant MacNeil questions the conditions under which cases may be reopened, particularly regarding new and substantial evidence (Speaker 3, Speech ID: 1183992).
   - **Argument D**: Charles Grant MacNeil seeks clarification on the regulations governing appeals based on errors or new evidence (Speaker 4, Speech ID: 1183994).
   - **Argument E**: Charles Gavan Power confirms that the conditions for reopening cases remain unchanged from the current act (Speaker 5, Speech ID: 1183993).

7. **Relationships**:
   - **Inter-Argument Relationships**:
     - Argument A (Speaker 1, Speech ID: 1183989) supports Argument B (Speaker 2, Speech ID: 1183988) by acknowledging the concern raised about the appeal process.
     - Argument C (Speaker 3, Speech ID: 1183992) and Argument D (Speaker 4, Speech ID: 1183994) build on the concern about the appeal process by asking for clarity on the specific regulations.
   - **Argument-Subject Relationships**:
     - Argument A addresses the subject of Appeal Board Procedures by discussing the composition of the appeal board.
     - Argument B addresses Pension Application Process by raising concerns about the fairness of the appeal process.
     - Argument C and D address Regulations on Pension Claims by asking for clarification on how appeals can be reopened.

8. **Summary**:
   This debate focused on the abolition of the Pension Appeal Board and the extension of time for pension applications. Charles Gavan Power, the Minister of Pensions and National Health, argued that there is no need for specific provisions regarding the appeal board, as common sense should prevail. Howard Charles Green expressed concerns over potential dissatisfaction among pension applicants if the same commissioners handle both original applications and appeals. Charles Grant MacNeil further questioned the reopening of cases based on new evidence and sought clarity on the regulations that would govern such appeals. The discussion highlighted the complexities of the pension application process and the need for fairness and clarity in handling appeals.

Mermaid Graph Representation:

```mermaid
graph TD
   %% Debate and Topics
   D[Debate: Abolition of Pension Appeal Board and Extension of Time for Pension Applications<br>1939-04-04, Thirty-Sixth Parliament, 1st Session]:::debate
   T1[Topic: Justice and Law]:::topic
   T2[Topic: Social Policy]:::topic
   D -->|debate about| T1
   D -->|debate about| T2
```

```
%% Subjects
S1[Pension Application Process]:::subject
S2[Appeal Board Procedures]:::subject
S3[Regulations on Pension Claims]:::subject
S1 -->|discussed during| D
S2 -->|discussed during| D
S3 -->|discussed during| D

%% Parties
P1[Liberal]:::party
P2[Conservative (1867-1942)]:::party
P3[Co-operative Commonwealth Federation (C.C.F.)]:::party

%% Speakers
Spk1[Charles Gavan Power]:::speaker
Spk2[Howard Charles Green]:::speaker
Spk3[Charles Grant MacNeil]:::speaker

%% Arguments
A1[No need for specific provisions on appeal board composition]:::argument
A2[Concerns over dissatisfaction if same commissioner sits on both original
    claim and appeal]:::argument
A3[Clarification sought on conditions for reopening cases]:::argument
A4[Questions about regulations governing appeals based on new evidence]:::
   argument
A5[Conditions for reopening cases remain unchanged]:::argument

%% Party-Speaker Relationships
P1 -->|party of| Spk1
P2 -->|party of| Spk2
P3 -->|party of| Spk3

%% Speaker-Argument Relationships
Spk1 -->|makes -- 1183989| A1
Spk2 -->|makes -- 1183988| A2
Spk3 -->|makes -- 1183992| A3
Spk3 -->|makes -- 1183994| A4
Spk1 -->|makes -- 1183993| A5

%% Argument-Subject Relationships
A1 -->|addresses| S2
A2 -->|addresses| S1
A3 -->|addresses| S3
```

```
   A4 -->|addresses| S3
   A5 -->|addresses| S1

   %% Styling
   classDef debate fill:#C7CEEA,stroke:#4B0082,stroke-width:2px;
   classDef topic fill:#FF9999,stroke:#FF0000,stroke-width:2px;
   classDef subject fill:#A9DFBF,stroke:#27AE60,stroke-width:2px;
   classDef party fill:#8E44AD,stroke:#8E44AD,stroke-width:2px;
   classDef speaker fill:#AED6F1,stroke:#3498DB,stroke-width:2px;
   classDef argument fill:#FAD7A0,stroke:#F39C12,stroke-width:2px;
```

```
This graph reflects the structure of the debate, highlighting key arguments,
   subjects addressed, and the affiliations of the speakers involved.
```

B.1.2 Mermaid Graph Schema

The LegisGraph approach uses Mermaid, a markdown-like syntax for creating graphs, to represent the structure of legislative debates. This schema is designed to capture the complex relationships between various elements of a debate while maintaining readability and ease of interpretation.

B.1.2.1 Node Types

The Mermaid graphs in LegisGraph include the following node types:

- **Debate**: Represents the overall debate, including metadata such as date, session, and bill number.
- **Topic**: Broad categories of discussion, selected from a predefined list (e.g., "Environmental Policy", "Economic Policy").
- **Subject**: More specific areas of discussion within a topic.
- **Party**: Political parties involved in the debate.
- **Speaker**: Individual participants in the debate.
- **Argument**: Key points or assertions made by speakers.
- **Procedural Motion**: Formal actions taken during the debate (e.g., amendments, referrals).

B.1.2.2 Edge Types

Relationships between nodes are represented by edges, with the following types:

- **debate about**: Connects Debate to Topic nodes (solid line).
- **discussed during**: Connects Subject to Debate nodes (solid line).
- **party of**: Connects Party to Speaker nodes (solid line).
- **makes**: Connects Speaker to Argument nodes, annotated with Speech ID (solid line).
- **moves**: Connects Speaker to Procedural Motion nodes, annotated with Speech ID (solid line).
- **addresses**: Connects Argument to Subject nodes (solid line).
- **affects**: Connects Procedural Motion to Subject nodes (solid line).
- **supports/contradicts**: Connects Argument to Argument nodes (dotted line).

Styling Conventions To enhance readability and visual distinction, nodes are styled using CSS classes:

```
classDef debate fill:#C7CEEA,stroke:#4B0082,stroke-width:2px;
classDef topic fill:#FF9999,stroke:#FF0000,stroke-width:2px;
classDef subject fill:#A9DFBF,stroke:#27AE60,stroke-width:2px;
classDef party fill:#8E44AD,stroke:#8E44AD,stroke-width:2px;
classDef speaker fill:#AED6F1,stroke:#3498DB,stroke-width:2px;
classDef argument fill:#FAD7A0,stroke:#F39C12,stroke-width:2px;
classDef motion_passed fill:#4CAF50,stroke:#4CAF50,stroke-width:2px;
classDef motion_failed fill:#F44336,stroke:#F44336,stroke-width:2px;
```

B.1.2.3 Graph Structure

The overall structure of the Mermaid graph is designed to represent the flow and relationships of a legislative debate:

- Subject nodes are linked to the Debate node.
- Party nodes are connected to their respective Speaker nodes.
- Speaker nodes are linked to their Arguments and Procedural Motions.

- Arguments are connected to the Subjects they address and to other Arguments they support or contradict.
- Procedural Motions are linked to the Subjects they affect.

This schema allows for a comprehensive representation of a debate's structure, capturing not only the content of arguments and motions but also the complex interrelationships between different elements of the debate. The visual distinction between node types and edge styles helps in quickly interpreting the graph and understanding the flow of the debate.

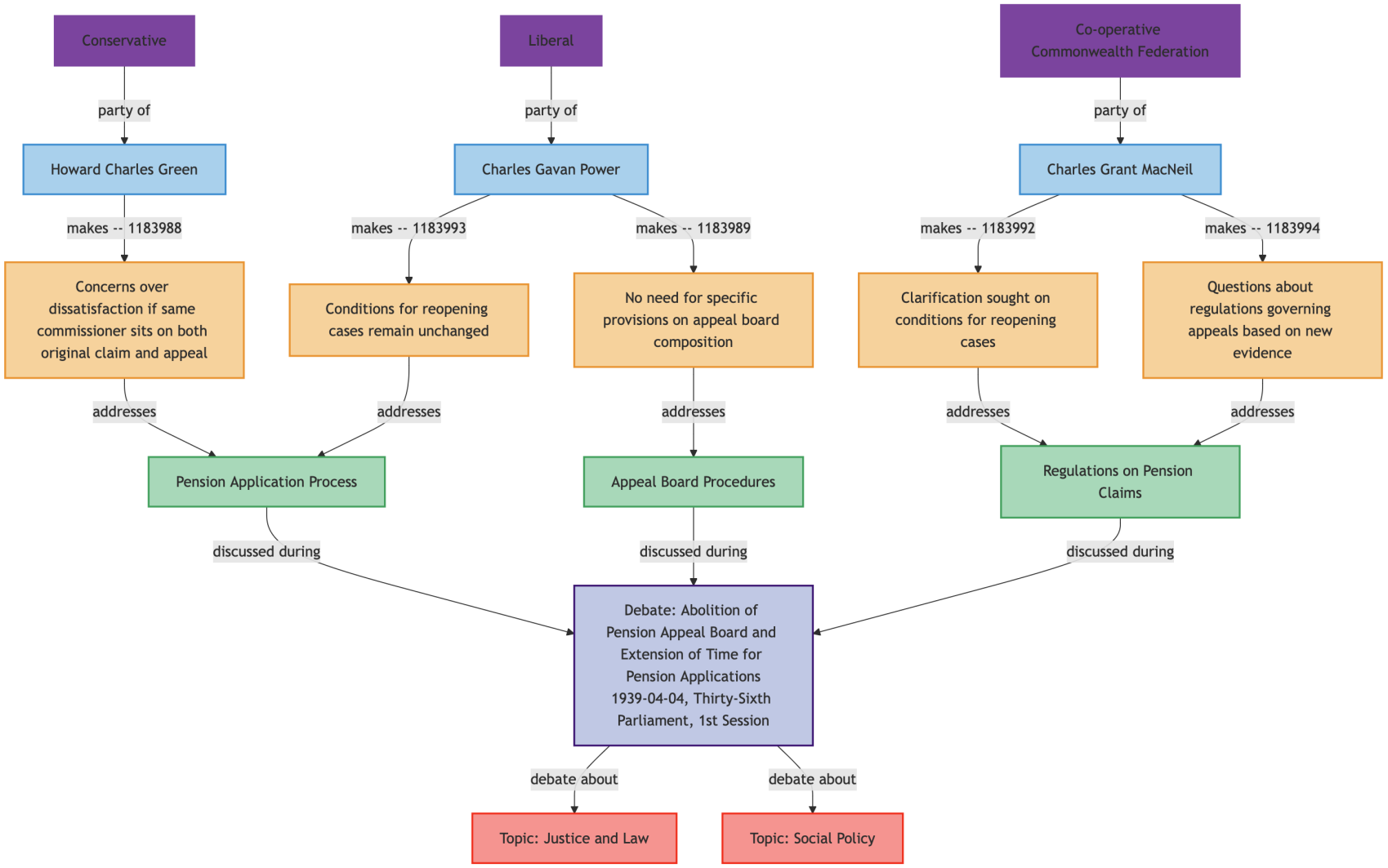


Figure B.1: Example Mermaid Graph Representation of a Legislative Debate. The graph shows the structure of the debate, including topics, subjects, parties, speakers, arguments, and procedural motions. Purple nodes represent parties, blue nodes speakers, yellow nodes arguments, green nodes subjects, red nodes topics, and orange nodes procedural motions. Solid lines represent relationships between nodes, while dotted lines indicate support or contradiction between arguments.

B.1.3 Graph Generation Prompt Refinement Process

The development of the graph generation prompt was an extensive, iterative process involving multiple rounds of testing, analysis, and refinement. This appendix provides a comprehensive overview of how the prompt evolved from its initial conception to its final form, detailing the challenges encountered, solutions implemented, and the rationale behind key decisions. The refinement process followed the Prompt Refinement Procedure outlined in Algorithm 2.

B.1.3.1 Initial Prompt Design

Following step 1 of Algorithm 2, I designed the initial version of the prompt based on the findings from Bosley (2024). This version was intended to capture the basic elements of a legislative debate without imposing too much structure, allowing for flexibility in analyzing different types of debates. The initial prompt was as follows:

```
Analyze the provided legislative debate transcript and create a
   structured representation of the debate, including:
1. Speakers and their party affiliations
2. Main topics discussed
3. Key arguments made by each speaker
4. Any procedural motions or votes

Generate a Mermaid graph to visualize these elements and their
   relationships.
```

This initial version had several limitations. It lacked specificity in identifying topics and arguments, provided no clear structure for representing complex relationships, gave insufficient guidance on how to handle procedural elements, and offered no standardization of node or edge types. These limitations became apparent as I began testing the prompt with various debate transcripts.

B.1.3.2 Synthetic Example Generation

In accordance with step 2 of Algorithm 2, I collaborated with the LLM model to create diverse synthetic examples. These examples were designed to cover various debate structures and complexities, ensuring that the prompt could handle a wide range of legislative discussions. The synthetic examples were then added to the initial prompt to provide concrete guidance on expected outputs.

B.1.3.3 First Iteration: Adding Structure and Specificity

The first round of refinements focused on adding more structure to the prompt and providing clearer guidelines for analysis. Following step 3 of Algorithm 2, I applied the prompt to sample debates, evaluated the outputs, and identified areas for improvement. Based on this analysis, I updated the prompt as follows:

```
Analyze the legislative debate transcript and extract the following
   information:

1. Parties: List all political parties mentioned.
2. Speakers: Identify all speakers and their party affiliations.
3. Topics: Determine 2-4 main topics of the debate.
4. Arguments: For each speaker, list their key arguments as assertive
   statements.
5. Procedural Motions: Note any formal motions, amendments, or votes.

Generate a Mermaid graph with the following elements:
- Nodes for parties, speakers, topics, arguments, and motions
- Edges connecting speakers to their arguments (label: "argues")
- Edges connecting arguments to topics (label: "addresses")
- Edges connecting speakers to motions (label: "proposes" or "votes")

Use different node shapes for different element types (e.g., circles
   for speakers, rectangles for arguments).
```

While these changes brought significant improvements, several issues remained, including a lack of standardization in topic identification and limited guidance on handling complex procedural elements.

B.1.3.4 Second Iteration: Introducing Standardization and Hierarchy

The second major refinement introduced standardized topic lists and a hierarchical structure for debate elements. This iteration addressed the issues identified in the previous round, as per step 3.3 of Algorithm 2. Key updates included:

```
3. Topics and Subjects:
  - Identify 1-3 broad Topics that the debate focuses on, selected from
      the following list:
    - Environmental Policy
    - Economic Policy
```

```
    - Healthcare
    - Education
    - Foreign Affairs
    [... other topics ...]
  - Determine 2-4 more specific Subjects within each Topic that are
    discussed in the debate.

4. Debate Node: Create a node representing the entire debate, including
   key metadata (date, session, bill number).

5. Procedural Motions:
  - Identify all formal procedural actions, including motions,
    amendments, and their outcomes.
  - For each motion, note the type, who proposed it, the outcome (
    passed/failed), and which Subject(s) it affects.
```

These improvements addressed several key challenges, including inconsistency in topic identification across different analyses and difficulty in representing the overall structure of the debate.

B.1.3.5 Third Iteration: Enhancing Relationship Representation

The third major refinement focused on improving the representation of complex relationships within the debate. This iteration was a result of the manual review process described in step 3.2 of Algorithm 2. The updated prompt included:

```
7. Relationships:
  - Identify clear support or contradiction between arguments, and
    between arguments and subjects.
  - Inter-Argument Relationships:
    - Identify and describe the relationships between different
      arguments presented in the debate.
    - Consider how arguments support, contradict, modify, or address
      each other.
    - Provide specific examples of these relationships, referencing the
      speakers and speech IDs.
  - Argument-Subject Relationships:
    - For each argument identified, specify which subject(s) it
      addresses.
    - Explain how the argument relates to or impacts the subject(s).
```

These improvements addressed the challenges of capturing the complex interplay between arguments and representing how arguments impact the debate's subjects.

B.1.3.6 Final Version: Refining Details and Enhancing Usability

The final version of the prompt incorporated further refinements based on extensive testing with a wide range of debate transcripts, as per the iterative process described in Algorithm 2. Key additions included:

```
8. Focus on key interactions: Limit the number of inter-argument
   relationships to maintain clarity (prioritize representing the most
   significant and direct interactions).
9. Include metadata for the debate as a whole (date, session, bill
   number if applicable) within the Debate node label.
10. If the platform allows, consider making the graph interactive for
   easier exploration:
   - Add click interactions to nodes or edges to display additional
      details or highlight connected elements.
   - Implement a timeline slider or other temporal navigation to focus
      on specific portions of the debate.
```

These final refinements significantly improved the prompt's effectiveness and the resulting graph's utility, addressing the ongoing challenge of balancing comprehensiveness with clarity in complex debates.

B.1.3.7 Refinement Process Methodology

Throughout the refinement process, I employed a systematic approach to improve the prompt, closely following the steps outlined in Algorithm 2. This methodology included diverse sample testing, where each version was tested against a variety of debate transcripts from different legislative bodies, on various policy areas, and with varying levels of contentiousness and procedural complexity.

Throughout the process, I maintained detailed documentation of changes, including the rationale behind each modification and its observed impact. This rigorous, multi-stage refinement process resulted in a robust and flexible prompt capable of generating informative and accurate graph representations of a wide range of legislative debates. The final version strikes a balance between capturing the complexity of legislative discussions and producing clear, interpretable visualizations.

B.2 DQI+ Annotation Prompt Overview

B.2.1 DQI+ Criteria

Table B.1: Discourse Quality Index (DQI) Criteria

Criterion	Description
1. Participation	**N/A:** Speaker did not have an opportunity to participate **0:** Participation was impaired **1:** Normal participation was possible
2.1 Level of Justification	**N/A:** No argument or claim made that requires justification **0:** No justification given **1:** Justification given but without clear linkage to claim **2:** Single reason given with clear linkage to claim **3:** Multiple reasons given, each with clear linkage to claim **4:** Multiple reasons given with clear linkages, plus at least one reason elaborated in depth
2.2 Content of Justification	**N/A:** No justification provided or applicable **0:** Explicitly self-interested or constituency-specific justification **1:** Neutral justification without reference to broader impacts **2:** Justification referring to broad common good or societal benefit **3:** Justification explicitly considering impacts on disadvantaged or underrepresented groups
3.1 Respect toward Groups	**N/A:** No mention of affected groups **0:** Explicitly negative statement about group to be helped

	1: Neither negative nor positive statement about group **2:** Explicitly positive statement about group
3.2 Respect toward main Demand	**N/A:** No mention of demands **0:** Explicitly negative statement about demand **1:** Neither negative nor positive statement about demand **2:** Explicitly positive statement about demand, but without full endorsement **3:** Fully agreeing with and positively valuing demand, expressing clear support
3.3 Respect toward main Counterarguments	**N/A:** No counterarguments presented or addressed **0:** Counterarguments degraded or dismissed without consideration **1:** Counterarguments ignored **2:** Counterarguments acknowledged but quickly dismissed without substantial engagement **3:** Counterarguments explicitly valued and engaged with, even if ultimately disagreed with **4:** Counterarguments agreed with, either in part or in full
3.4 Respect toward Same-Party Members	**N/A:** No interaction with or mention of same-party members **0:** Explicit accusation of bad faith or obstruction **1:** Strong implication of bad faith or obstruction **2:** Mild criticism or implication of misconduct **3:** Neutral or procedural interaction **4:** Explicitly respectful or collaborative interaction
3.5 Respect toward Other-Party Members	**N/A:** No interaction with or mention of other-party members **0:** Explicit accusation of bad faith or obstruction **1:** Strong implication of bad faith or obstruction **2:** Mild criticism or implication of misconduct **3:** Neutral or procedural interaction

	4: Explicitly respectful or collaborative interaction
4. Constructive Politics	**N/A:** No opportunity for constructive politics **0:** Purely oppositional or obstructive **1:** Restatement of position without new ideas **2:** New ideas or alternatives proposed, but not directly addressing others' concerns **3:** Attempt at finding common ground by partially addressing others' concerns or modifying own position **4:** Specific, actionable compromise or synthesis proposed that substantially addresses concerns from multiple perspectives
5. Relevance to Topic	**0:** Completely off-topic **1:** Tangentially related to the topic **2:** Somewhat relevant to the topic, but focuses more on peripheral issues **3:** Mostly relevant to the topic, addressing key aspects with some digressions **4:** Directly addressing the main topic of debate, staying focused on central issues
6. Tone	**0:** Highly confrontational or aggressive **1:** Sarcastic or subtly antagonistic **2:** Neutral or matter-of-fact **3:** Cordial and constructive **4:** Highly collaborative and solution-oriented, actively seeking to build consensus

B.2.2 Complete Annotated DQI Annotation Prompt

B.2.2.1 High Level Instructions

```
You are an AI assistant tasked with analyzing legislative debates using the
   Discourse Quality Index (DQI). Your goal is to provide a comprehensive
   analysis of the debate's discourse quality, focusing on specific speeches
   identified in the debate graph.

## INSTRUCTIONS
```

Your Task:
1. Review the entire debate text to understand the context and flow of the discussion.
2. Examine the provided mermaid graph, focusing on the "Speaker-Argument Relationships" section.
3. Identify the specific speeches mentioned in the Speaker-Argument Relationships and locate them in the debate text.
4. For each identified speech:
   a. Apply the DQI criteria, providing a score for each category.
   b. Offer detailed reasoning for each score, including relevant quotes from the speech.
   c. Ensure your analysis considers the speech's context within the broader debate.
5. After analyzing all identified speeches, provide a summary analysis of the debate's overall discourse quality.

Output Format:
- Present your analysis in a structured markdown format.
- Include sections for Speakers, Topics and Subjects, and DQI Annotations.
- For each speech, list all DQI criteria with scores and reasoning.
- Conclude with an overall analysis summarizing key trends and observations.

Important Considerations:
- Focus only on the speeches explicitly mentioned in the Speaker-Argument Relationships of the mermaid graph.
- Provide a chronological analysis that captures how the quality of discourse evolves throughout the debate.
- Pay attention to how speakers influence each other and how the overall tone or approach to the debate shifts over time.
- Use specific quotes from the speeches to support your scoring and reasoning.
- Ensure your analysis is objective and based solely on the content and quality of the arguments presented, not on personal opinions about the topic being debated.

Note on Scoring: When a speech falls between two scoring levels, you may use half-point scores (e.g., 1.5, 2.5, 3.5). These should be used when there is genuine uncertainty or when the speech shows characteristics of both adjacent levels. Always provide a brief explanation for why you chose a half-point score.

DISCOURSE QUALITY INDEX (DQI) CRITERIA

1. Participation
  N/A: Speaker did not have an opportunity to participate
  0: Participation was impaired
  1: Normal participation was possible

Notes to annotator: Look for instances where a speaker is interrupted, spoken over, or not given their allotted time to speak. If there are no clear impediments to the speaker's ability to participate, score this as 1. Only use N/A if the speaker was not supposed to speak at this point in the debate.

2. Justification
  2.1 Level of Justification
  N/A: No argument or claim made that requires justification
  0: No justification given
  1: Justification given but without clear linkage to claim (e.g., "because that's the way it is")
  2: Single reason given with clear linkage to claim
  3: Multiple reasons given, each with clear linkage to claim
  4: Multiple reasons given with clear linkages, plus at least one reason elaborated in depth (with specific examples, data, or extended logical reasoning)

Notes to annotator: Focus on the structure and complexity of the justification. Look for clear connections between reasons and claims. For level 4, ensure that at least one reason is explored in significant depth, not just mentioned. "In-depth" elaboration should include specific examples, relevant data, or extended logical reasoning that goes beyond a simple statement.

  2.2 Content of Justification
  N/A: No justification provided or applicable
  0: Explicitly self-interested or constituency-specific justification
  1: Neutral justification without reference to broader impacts
  2: Justification referring to broad common good or societal benefit
  3: Justification explicitly considering impacts on disadvantaged or underrepresented groups

Notes to annotator: Consider the scope of the justification. Is the speaker only considering their own interests or those of their specific constituents (0), or are they considering broader societal impacts (2-3)? Look for explicit mentions of societal benefit or disadvantaged groups for higher scores.

3. Respect

3.1 Respect toward Groups

N/A: No mention of affected groups

0: Explicitly negative statement about group to be helped

1: Neither negative nor positive statement about group

2: Explicitly positive statement about group

Notes to annotator: Focus on how the speaker talks about groups that would be affected by the policy under discussion, not political parties or other legislators. Look for clearly positive or negative characterizations of these groups.

3.2 Respect toward main Demand

N/A: No mention of demands

0: Explicitly negative statement about demand

1: Neither negative nor positive statement about demand

2: Explicitly positive statement about demand, but without full endorsement

3: Fully agreeing with and positively valuing demand, expressing clear support

Notes to annotator: The "demand" here refers to the main policy proposal or change being debated. Pay attention to how the speaker characterizes this demand, from outright rejection (0) to full agreement and support (3). For score 2, look for positive statements that fall short of full endorsement, while score 3 should indicate clear and unequivocal support.

3.3 Respect toward main Counterarguments

N/A: No counterarguments presented or addressed

0: Counterarguments degraded or dismissed without consideration

1: Counterarguments ignored

2: Counterarguments acknowledged but quickly dismissed without substantial engagement

3: Counterarguments explicitly valued and engaged with, even if ultimately disagreed with

4: Counterarguments agreed with, either in part or in full

Notes to annotator: Look at how the speaker handles arguments contrary to their position. Do they dismiss or mock these arguments (0), acknowledge but quickly dismiss them (2), show respect for them by engaging substantively (3), or even agree with some points (4)?

3.4 Respect toward Same-Party Members

N/A: No interaction with or mention of same-party members
0: Explicit accusation of bad faith or obstruction
1: Strong implication of bad faith or obstruction
2: Mild criticism or implication of misconduct
3: Neutral or procedural interaction
4: Explicitly respectful or collaborative interaction

Notes to annotator: Focus on how the speaker interacts with or refers to members of their own party. Look for signs of internal party conflict (0-2) or unity and collaboration (3-4).

3.5 Respect toward Other-Party Members
N/A: No interaction with or mention of other-party members
0: Explicit accusation of bad faith or obstruction
1: Strong implication of bad faith or obstruction
2: Mild criticism or implication of misconduct
3: Neutral or procedural interaction
4: Explicitly respectful or collaborative interaction

Notes to annotator: Similar to 3.4, but focusing on interactions with members of other parties. Pay attention to accusations of obstruction or bad faith (0-1) versus more respectful or collaborative interactions (3-4).

4. Constructive Politics
N/A: No opportunity for constructive politics
0: Purely oppositional or obstructive
1: Restatement of position without new ideas
2: New ideas or alternatives proposed, but not directly addressing others' concerns
3: Attempt at finding common ground by partially addressing others' concerns or modifying own position
4: Specific, actionable compromise or synthesis proposed that substantially addresses concerns from multiple perspectives

Notes to annotator: Evaluate how the speaker contributes to moving the debate forward. Are they simply restating positions (1), offering new ideas without considering others' viewpoints (2), actively trying to find common ground by partially addressing concerns or showing flexibility (3), or proposing specific compromises that bridge multiple perspectives (4)?

5. Relevance to Topic
0: Completely off-topic
1: Tangentially related to the topic

```
   2: Somewhat relevant to the topic, but focuses more on peripheral issues
   3: Mostly relevant to the topic, addressing key aspects with some
      digressions
   4: Directly addressing the main topic of debate, staying focused on central
      issues

Notes to annotator: Assess how closely the speaker's comments align with the
    main topic of debate. Be aware of attempts to introduce unrelated issues
    or steer the conversation away from the central topic. For scores 2 and 3,
     consider the balance between relevant points and digressions or
    peripheral issues.

6. Tone
   0: Highly confrontational or aggressive
   1: Sarcastic or subtly antagonistic
   2: Neutral or matter-of-fact
   3: Cordial and constructive
   4: Highly collaborative and solution-oriented, actively seeking to build
      consensus

Notes to annotator: Focus on the overall manner and tone of the speech. Look
    beyond the content to how it's delivered. Pay attention to language
    choices, emphasis, and any non-verbal cues that might be noted in the
    transcript. For score 4, look for explicit attempts to build consensus and
     bridge divides, going beyond mere cordiality.

When using half-point scores, be sure to explain your reasoning clearly.
    Highlight which aspects of the speech align with the lower score and which
     align with the higher score. This will help maintain consistency and
    transparency in the scoring process.
```

B.2.2.2 Example Annotations

```
## EXAMPLES OF ANNOTATIONS

## Example Debate 1: Renewable Energy Transition Act
```
Debate: Renewable Energy Transition Act
Date: 2023-06-15
Session: Summer Session
Bill: S.123

Speaker 1 (Green Party): Madam Speaker, I rise today to argue for the urgent
```
```

passage of the Renewable Energy Transition Act. Climate change poses an existential threat to our planet, and we must act decisively. This bill proposes a rapid shift to renewable energy sources, coupled with significant investments in green technology. We cannot afford to delay any longer. (Speech ID: 1)

Speaker 2 (Conservative Party): While I appreciate the concern for our environment, I must express reservations about the pace and scale of the proposed changes. The immediate transition this bill demands could lead to job losses in traditional energy sectors and potentially destabilize our energy grid. We need a more measured approach that balances environmental concerns with economic stability. (Speech ID: 2)

[...]

Analysis:

1. Parties: The debate involves three parties: Green Party, Conservative Party, and Labor Party.

2. Speakers:
   - Speaker 1 (Green Party)
   - Speaker 2 (Conservative Party)
   - Speaker 3 (Labor Party)
   - Speaker 4 (Conservative Party)
   - Speaker 5 (Green Party)
   - Speaker 6 (Labor Party)

3. Topics and Subjects:
   - Topics:
     1. Environmental Policy
     2. Economic Policy
   - Subjects:
     1. Renewable Energy Transition
     2. Economic Impact
     3. Job Market Changes
     4. Green Technology Investment

4. Debate Node: Renewable Energy Transition Act, 2023-06-15, Summer Session, Bill S.123

5. Procedural Motions:
   - Motion to refer the bill to the Economic Affairs Committee (failed, Speech

```
      ID: 5)
  - Amendment to extend implementation timeline from 5 to 8 years (passed,
      Speech ID: 6)
  - Motion to extend debate by one hour (passed, Speech ID: 10)

[...]

## Example Mermaid Graph:

```mermaid
graph TD
   %% Debate and Topics
   D[Debate: Renewable Energy Transition Act<br>2023-06-15, Summer Session,
      Bill S.123]:::debate
   T1[Topic: Environmental Policy]:::topic
   T2[Topic: Economic Policy]:::topic
   D -->|debate about| T1
   D -->|debate about| T2

   %% Subjects
   S1[Renewable Energy Transition]:::subject
   S2[Economic Impact]:::subject
   S3[Job Market Changes]:::subject
   S4[Green Technology Investment]:::subject
   S1 -->|discussed during| D
   S2 -->|discussed during| D
   S3 -->|discussed during| D
   S4 -->|discussed during| D

[...]
```

Example Debate: Renewable Energy Transition Act

Speakers:
- Speaker 1 (Green Party)
- Speaker 2 (Conservative Party)
- Speaker 3 (Labor Party)
- Speaker 5 (Green Party)

Topics and Subjects:
- Topics:
  1. Environmental Policy
```

  2. Economic Policy
- Subjects:
  1. Renewable Energy Transition
  2. Economic Impact
  3. Job Market Changes
  4. Green Technology Investment

DQI Annotations:

1. Speech ID: 1 (Speaker 1, Green Party)
  - Participation: 1
    Reasoning: The speaker presents their argument without interruption. "Madam Speaker, I rise today to argue for the urgent passage of the Renewable Energy Transition Act."
  - Justification:
    - Level: 2
      Reasoning: A single clear reason is provided: the existential threat of climate change. "Climate change poses an existential threat to our planet, and we must act decisively."
    - Content: 2
      Reasoning: The justification refers to a broad common good (addressing climate change), which affects the entire planet.
  - Respect:
    - Toward Groups: N/A
      Reasoning: No specific groups are mentioned in the speech.
    - Toward main Demand: 3
      Reasoning: The speaker fully supports and positively values the bill, emphasizing its urgency and necessity.
    - Toward main Counterarguments: N/A
      Reasoning: No counterarguments are presented or addressed in this opening statement.
    - Toward Same-Party Members: N/A
      Reasoning: No interaction with or mention of same-party members.
    - Toward Other-Party Members: N/A
      Reasoning: No interaction with or mention of other-party members.
  - Constructive Politics: 2
    Reasoning: The speaker proposes new ideas (rapid shift to renewable energy, investments in green technology) but doesn't address potential concerns or opposition.
  - Relevance to Topic: 4
    Reasoning: The speech directly addresses the main topic of the Renewable Energy Transition Act.
  - Tone: 3

```
    Reasoning: The tone is cordial and constructive, presenting the bill as a
       necessary solution to a critical problem.

[...]
```

B.2.2.3 Closing Instructions

```
## TOTAL DEBATE TEXT
{debate_text}

## DEBATE ANALYSIS
{debate_analysis}

Generate the annotations below. Ensure that the markdown lists are formatted
   as instructed. Here is an illustrative example.
1. Speech ID: [Speech ID Here] (Speaker 1, Party A)
  - Participation: [Participation Score]
    Reasoning: [Explain the reasoning for the score given.]
  - Justification:
    - Level: [Justification Level Score]
      Reasoning: [Explain the reasoning for the score given.]
    - Content: [Content Score]
      Reasoning: [Explain the reasoning for the score given.]
  - Respect:
    - Toward Groups: N/A
      Reasoning: [Explain the reasoning for the score given.]
    - Toward main Demand: [Respect Score]
      Reasoning: [Explain the reasoning for the score given.]]
    - Toward main Counterarguments: [Respect Score]
      Reasoning: [Explain the reasoning for the score given.]
    - Toward Same-Party Members: N/A
      Reasoning: [Explain the reasoning for the score given.]
    - Toward Other-Party Members: [Respect Score]
      Reasoning: [Explain the reasoning for the score given.]
  - Constructive Politics: [Constructive Politics Score]
    Reasoning: [Explain the reasoning for the score given.]
  - Relevance to Topic: [Relevance Score]
    Reasoning: [Explain the reasoning for the score given.]
  - Tone: [Tone Score]
    Reasoning: [Explain the reasoning for the score given.]
```

B.3 Extracting Information from the Neo4j database

This appendix provides a detailed explanation of the key database queries used in the analysis of legislative debates. Each query is presented in the Cypher query language, followed by a description of its purpose, function, and the insights it provides.

B.3.1 Discourse Quality Index (DQI) Trends

```
MATCH (d:Debate)<-[:DISCUSSED_DURING]-(s:Subject)<-[:ADDRESSES]-(a:
   Argument)<-[:MAKES]-(sp:Speech)
WITH d.parliament AS parliament, sp
RETURN
parliament,
avg(sp.participation) AS avg_participation,
avg(sp.justification_level) AS avg_justification,
avg(sp.respect_groups) AS avg_respect,
avg(sp.constructive_politics) AS avg_constructive_politics,
avg(sp.relevance) AS avg_relevance,
avg(sp.tone) AS avg_tone
ORDER BY parliament
```

This query tracks changes in the Discourse Quality Index across different parliaments. It matches the path from Debates through Subjects and Arguments to Speeches, grouping the data by parliament. For each parliament, it computes the average scores for each DQI component: participation, justification level, respect for groups, constructive politics, relevance, and tone. The average DQI score for a component c in parliament p is calculated as:

$$\overline{DQI}c, p = \frac{1}{n_p} \sum i = 1^{n_p} DQI_{c,i} \tag{B.1}$$

where n_p is the number of speeches in parliament p, and $DQI_{c,i}$ is the score for component c in speech i. The results are ordered by parliament, allowing for the observation of trends in discourse quality across successive parliaments. This query is crucial for understanding how the quality of legislative debate has evolved, potentially revealing parliaments of particularly high or low discourse quality and long-term trends in parliamentary communication.

B.3.2 Inter-Party Argument Dynamics

```
MATCH (d:Debate)<-[:DISCUSSED_DURING]-(s:Subject)<-[:ADDRESSES]-(a1:
   Argument)<-[:MAKES]-(sp1:Speech)<-[:GIVES]-(s1:Speaker)<-[:PARTY_OF
   ]-(p1:Party)
MATCH (a2:Argument)<-[:MAKES]-(sp2:Speech)<-[:GIVES]-(s2:Speaker)<-[:
   PARTY_OF]-(p2:Party)
WHERE p1 <> p2 AND (a1)-[:SUPPORTS|CONTRADICTS]->(a2)
WITH d.parliament AS parliament, p1.name AS party1, p2.name AS party2,
type((a1)-[:SUPPORTS|CONTRADICTS]->(a2)) AS relation_type,
count(*) AS count
RETURN
parliament,
party1,
party2,
sum(CASE WHEN relation_type = 'SUPPORTS' THEN count ELSE 0 END) AS
   support_count,
sum(CASE WHEN relation_type = 'CONTRADICTS' THEN count ELSE 0 END) AS
   contradict_count,
toFloat(sum(CASE WHEN relation_type = 'SUPPORTS' THEN count ELSE 0 END)
   ) /
(sum(CASE WHEN relation_type = 'SUPPORTS' THEN count ELSE 0 END) +
sum(CASE WHEN relation_type = 'CONTRADICTS' THEN count ELSE 0 END)) AS
   support_ratio
ORDER BY parliament, party1, party2
```

This query analyzes the dynamics of inter-party interactions in debates for each parliament. It matches pairs of arguments made by speakers from different parties, counting instances where one argument supports or contradicts another. The query returns a summary for each pair of parties within each parliament, showing the number of supporting and contradicting interactions, as well as a support ratio. The support ratio for parties i and j in parliament p is calculated as:

$$R_{i,j,p} = \frac{S_{i,j,p}}{S_{i,j,p} + C_{i,j,p}} \tag{B.2}$$

where $S_{i,j,p}$ is the number of supporting interactions and $C_{i,j,p}$ is the number of contradicting interactions between parties i and j in parliament p. This ratio provides insight into the level of agreement or disagreement between parties within each parliament, helping to quantify party relationships and identify patterns of cooperation or opposition across different parliamentary terms.

```
MATCH (d:Debate)-[:DEBATE_ABOUT]->(t:Topic)
```

```
WITH d.parliament AS parliament, t.name AS topic, count(DISTINCT d) AS
   debate_count
MATCH (d:Debate)
WHERE d.parliament = parliament
WITH parliament, topic, debate_count, count(DISTINCT d) AS
   total_debates
RETURN
parliament,
topic,
debate_count,
toFloat(debate_count) / total_debates AS topic_percentage
ORDER BY parliament, topic_percentage DESC
```

This query tracks the evolution of debate topics across parliaments. It matches debates to their topics and groups them by parliament. For each parliament and topic, it counts the number of debates and calculates this as a percentage of total debates in that parliament. The percentage $P_{t,p}$ for topic t in parliament p is calculated as:

$$P_{t,p} = \frac{N_{t,p}}{N_p} \times 100 \tag{B.3}$$

where $N_{t,p}$ is the number of debates about topic t in parliament p, and N_p is the total number of debates in parliament p. The results are ordered by parliament and then by topic prevalence, allowing for the identification of which topics dominated parliamentary discussion in each parliament and how this changed over time. This query is valuable for identifying shifts in political priorities and public concerns as reflected in legislative debates across different parliamentary terms.

B.3.3 Argument Complexity and Interconnectedness

```
MATCH (d:Debate)<-[:DISCUSSED_DURING]-(s:Subject)<-[:ADDRESSES]-(a:
   Argument)
WITH d, a, count(DISTINCT s) AS subject_count
MATCH (a)-[:SUPPORTS|CONTRADICTS]-(other:Argument)
WITH d, a, subject_count, count(DISTINCT other) AS connection_count
WITH d, avg(subject_count) AS avg_subjects, avg(connection_count) AS
   avg_connections
RETURN
d.parliament AS parliament,
```

```
avg(avg_subjects) AS avg_subjects_per_argument,
avg(avg_connections) AS avg_connections_per_argument
ORDER BY parliament
```

This query measures the complexity and interconnectedness of arguments for each parliament. For each argument, it counts the number of subjects it addresses and the number of connections (support or contradiction relationships) it has with other arguments. These counts are then averaged for each debate and finally for each parliament. The average number of subjects per argument $\overline{S_p}$ and average number of connections per argument $\overline{C_p}$ for parliament p are calculated as:

$$\overline{S_p} = \frac{1}{N_p} \sum_{i=1}^{N_p} \frac{1}{n_i} \sum_{j=1}^{n_i} S_{i,j} \tag{B.4}$$

$$\overline{C_p} = \frac{1}{N_p} \sum_{i=1}^{N_p} \frac{1}{n_i} \sum_{j=1}^{n_i} C_{i,j} \tag{B.5}$$

where N_p is the number of debates in parliament p, n_i is the number of arguments in debate i, $S_{i,j}$ is the number of subjects addressed by argument j in debate i, and $C_{i,j}$ is the number of connections for argument j in debate i. The result shows how the complexity of arguments (measured by the number of subjects they address) and their interconnectedness (measured by their connections to other arguments) have changed across parliaments. This provides insight into whether parliamentary debates have become more or less complex and interconnected over different parliamentary terms.

B.3.4 Polarization Index

```
MATCH (d:Debate)<-[:DISCUSSED_DURING]-(s:Subject)<-[:ADDRESSES]-(a1:
   Argument)<-[:MAKES]-(sp1:Speech)<-[:GIVES]-(s1:Speaker)<-[:PARTY_OF
   ]-(p1:Party)
MATCH (a2:Argument)<-[:MAKES]-(sp2:Speech)<-[:GIVES]-(s2:Speaker)<-[:
   PARTY_OF]-(p2:Party)
WHERE p1 <> p2 AND (a1)-[:SUPPORTS|CONTRADICTS]->(a2)
WITH d, type((a1)-[:SUPPORTS|CONTRADICTS]->(a2)) AS relation_type,
   count(*) AS count
WITH d, sum(CASE WHEN relation_type = 'CONTRADICTS' THEN count ELSE 0
   END) AS contradict_count,
sum(count) AS total_count
WITH d, toFloat(contradict_count) / total_count AS polarization_index
```

```
RETURN
d.parliament AS parliament,
avg(polarization_index) AS avg_polarization
ORDER BY parliament
```

This query calculates a polarization index based on inter-party argument dynamics for each parliament. It looks at relationships between arguments made by speakers from different parties, counting the number of contradicting relationships as a proportion of total relationships (both supporting and contradicting). This proportion is averaged for each debate and then for each parliament. The polarization index P_p for parliament p is calculated as:

$$P_p = \frac{1}{N_p} \sum_{i=1}^{N_p} \frac{C_i}{T_i} \tag{B.6}$$

where N_p is the number of debates in parliament p, C_i is the number of contradicting relationships in debate i, and T_i is the total number of relationships (supporting and contradicting) in debate i. The resulting polarization index provides a measure of the level of disagreement between parties for each parliament. A higher index indicates more contradictions between parties, suggesting greater polarization. This query allows for the tracking of changes in political polarization across different parliaments, providing valuable insights into the evolving nature of political discourse and party relations in the legislature over successive parliamentary terms.

APPENDIX C

Appendix for "Improving Probabilistic Models in Text Classification via Active Learning"

C.1 Using Machine Learning for Text Classification

In this section we discuss the process of encoding text data into matrix form for the purpose of classifying e.g., whether a document refers to politics or not. We describe Supervised and Unsupervised Learning. In addition, this discussion extends to the selection between discriminative and generative models, shedding light on the advantages and trade-offs of each. Finally, we discuss the issue of model evaluation in text classification tasks, with a focus on using a validation dataset (test data) and evaluation metrics such as accuracy, precision, recall, and the F1 score.

C.1.1 Encoding Text in Matrix Form

Suppose that a researcher has a collection of social media text data, called a corpus, and wishes to classify whether each text in a corpus is political (e.g., refers to political protest, human rights violations, unfavorable views of a given candidate, targeted political repression, etc.) or not solely based on the words used in a given observation. Critically, the researcher does not yet know which of the texts are political or not at this point.

The researcher must first choose how to represent text as a series of *tokens*, and decide which tokens to include in their analysis. This involves a series of sub-choices, such as whether each token represents an individual word (such as "political") or a combination of words (such as "political party"), whether words should be stemmed or not (e.g., reducing both "political" and "politics" their common stem "politic"), and whether to remove stop-words (such as "in", "and", "on", etc.) that are collectively referred to as *pre-processing.*[1]

[1]For a survey of pre-processing techniques and their implications for political science research, see Denny and Spirling (2018).

The researcher must then choose how to encode information about these tokens in matrix form. The most straightforward way to accomplish this is using a *bag-of-words* approach, where the corpus is transformed into a document-feature matrix (DFM) $\mathbf{D}$ with n rows and m columns, where n is the number of documents and m is the number of tokens, which are more generally referred to as features.[2] Each element of the DFM encodes the frequency that a token occurs in a given document.[3] Once the researcher chooses how to encode their corpus as a matrix, she is left with a set of features corresponding to each document $\mathbf{D}$ and an unknown vector of true labels $\mathbf{Z}$, where each element of $\mathbf{Z}$ indicates whether a given document is political or not. Then, we can repose the classification question as follows: given $\mathbf{D}$, how might we best learn $\mathbf{Z}$, that is, whether each document is political or not?

C.1.2 Supervised, Unsupervised, and Semi-supervised Learning

Supervised and unsupervised learning are two fundamental approaches in machine learning for text-as-data (Grimmer et al., 2022). In the supervised approach, the researcher follows these steps: (1) acquiring accurate labels for a subset of the documents through human coding, where e.g., the researcher determines that a news headline such as "Biden and Trump clinch nominations, heading to a general election rematch" refers to politics due to its focus on a political figures (Presidents Biden and Trump) and their role in the upcoming general election; (2) establishing a connection between the textual features of each document in the corpus as represented by a matrix $\mathbf{D}$ and the true labels represented by a vector $\mathbf{Z}$ for the labeled documents. This involves understanding how terms like "Biden," "Trump," "clinch," "nominations," "general," "election," and "rematch" are important in determining the political nature of the headline; and (3) utilizing the acquired understanding of the relationship between the text data and the known labels to later predict whether the remaining unlabeled documents in the corpus are political or not (Hastie et al., 2009).

On the other hand, an unsupervised approach does not require the use of labeled data. Instead, the researcher employing an unsupervised approach would select a model that groups documents in the corpus based on shared patterns in the features as represented by the matrix

[2]Note that in the machine learning literature, the concept typically described by the term "variable" is communicated using the term "feature."

[3]An alternative to the bag-of-words approach is to encode tokens as *word embeddings*, where in addition to the matrix summarizing the incidences of words in each document, neural network models are used to create vector representations of each token. In this framework, each token is represented by a vector of some arbitrary length, and tokens that are used in similar contexts in the corpus (such as "minister" and "cabinet") will have similar vectors. While this approach is more complicated, it yields considerably more information about the use of words in the corpus than the simple count that the bag-of-words approach does. For an accessible introduction to the construction and use of word embeddings in political science research, see Rodriguez and Spirling (2022). For a more technical treatment, see Pennington et al. (2014).

D. After assigning documents to clusters, the researcher would determine which cluster corresponds to the desired outcome of interest, namely whether a document is political or not. However, it is important to note that there is no guarantee that there will be a direct connection between clusters and the outcomes of interest (Knox et al., 2022).

Semi-supervised learning combines supervised and unsupervised approaches (Miller and Uyar, 1996; Nigam et al., 2000), making it particularly useful when there is a large amount of unlabeled data and labeling is expensive. In a semi-supervised model, the relationship between the text data matrix **D** and the classification outcome **Z** is learned using both labeled and unlabeled data. Although **Z** is not known for unlabeled data, it still provides information about the joint distribution of the features in **D**. Thus, by incorporating patterns recovered from the unlabeled data and using the labeled data as a foundation for measurement, semi-supervised learning produces more accurate and robust predictions compared to purely supervised or unsupervised methods (Nigam et al., 2000).

C.1.3 Discriminative vs. Generative Models

In addition to choosing a supervised, unsupervised, or semi-supervised approach as described in Section "Machine Learning Approaches to Text Classification," a researcher must also choose whether to use a discriminative or generative model. As noted by Ng and Jordan (2001) and Bishop and Lassarre (2007), when using a discriminative model (e.g., logistic regression, SVM, etc.), the goal is to directly estimate the probability of the classification outcomes **Z** given the text data **D** i.e., directly estimate $p(\mathbf{Z}|\mathbf{D})$. In contrast, when using a generative model (e.g., Naive Bayes), learning the relationship between the **Z** and **D** is a two-step process. In the first step, the likelihood of the matrix of text data **D** and outcome labels **Z** is estimated given the data and a set of parameters θ that indicate structural assumptions about how the data is generated. In other words, $p(\mathbf{D}, \mathbf{Z}|\theta)$ is directly estimated. In the second step, the researcher uses Bayes' rule to calculate the probability of the outcome vector given the features and the learned distribution of the parameters i.e., $p(\mathbf{Z}|\mathbf{D}; \theta)$.

In addition to allowing for the use of unlabeled data (which reduces labeling costs), one of the main benefits of a generative rather than a discriminative model is that the researcher can include information they know about the data generating process by choosing appropriate functional forms.[4] This can help prevent overfitting when the amount of data in a corpus is small.[5] Conversely, because it is not necessary to model the data generating process directly,

[4]This is particularly true when e.g., the researcher knows that the data has a complicated hierarchical structure since the hierarchy can be incorporated directly into the generative model.

[5]Overfitting occurs when a model learns to predict classification outcomes based on patterns in the training set (i.e., the data used to fit the model) that does not generalize to the broader universe of cases to be classified. A model that is overfitted may predict the correct class with an extremely high degree of

the main benefit of a discriminative rather than generative model is simplicity (in general it involves estimating fewer parameters). Discriminative models are therefore appropriate in situations where the amount of data in a corpus is very large, and/or when the researcher is unsure about the data-generating process, which could lead to mis-specification (Bishop and Lassarre, 2007).[6]

C.1.4 Model Evaluation

A researcher must also decide when she is satisfied with the predictions generated by the model. In most circumstances, the best way to evaluate the performance of a classification algorithm is to reserve a subset of the corpus for validation, which is sometimes referred to as validation and/or test set. At the very beginning of the classification process, a researcher puts aside and label a set of randomly chosen documents that the active learning algorithm does not have access to.[7] Then, after training the model on the remainder of the documents (often called the training set), the researcher should generate predictions for the documents in the validation set using the trained model. By comparing the predicted labels generated by the model to the actual labels, the researcher can evaluate how well the model does at predicting the correct labels.

A common tool for comparing the predicted labels to the actual labels is a *confusion matrix*. In a binary classification setting, a confusion matrix will be a 2 by 2 matrix, with rows corresponding to the actual label, and the columns corresponding to the predicted label. Returning to our running example, imagine that the classification is to predict whether documents are political or not, Table C.1 shows the corresponding confusion matrix. In this scenario, True Positives (TP) are the number of documents that the model predicts to be about politics and that is in fact labeled as such. Correspondingly, True Negatives (TN), are the number of documents that the model predicts to be non-political and is labeled as such in the validation set. A False Negative (FN) occurs when the model classifies a document as non-political, but the document is about politics. Similarly, a False Positive (FP) occurs when the model classifies as political a document that is non-political.

Using the confusion matrix, the researcher can calculate a variety of evaluation statistics. Some of the most common of these are accuracy, precision, and recall. Accuracy is the

accuracy for items in the training set, but will perform poorly when used to predict the class for items that the model has not seen before.

[6]Another benefit of generative models is that they can yield better estimates of how certain we are about the relationship between the outcome and the features. This is the case when a researcher uses an inference algorithm like Markov Chain Monte Carlo (MCMC) that learns the entire distribution for each of the parameters, rather than only point estimates.

[7]It is important to use a set-aside validation set for testing model performance, rather than a subset of the documents used to train the model, to avoid *overfitting*.

		Predicted Label	
		Political	Non-political
Actual Label	Political	True Positive (TP)	False Negative (FN)
	Non-political	False Positive (FP)	True Negative (TN)

Table C.1: **Confusion Matrix: Comparison of the Predictions of a Classifier to Documents' True Labels**

proportion of documents that have been correctly classified. Precision is used to evaluate the false positivity rate and is the proportion of the model's positive classifications that are true positives. As the number of false positives increases (decreases), precision decreases (increases). Recall is used to evaluate the false negativity rate, and is the proportion of the actual positive documents that are true positives. As the number of false negatives increases, recall decreases, and *vice-versa*. Accuracy, precision, and recall can be formally calculated as:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}; \qquad \text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}; \qquad \text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

When the proportion of political and non-political documents in a corpus is balanced, accuracy is an adequate measure of model performance. However, it is often the case in text classification that the corpus is unbalanced, and the proportion of documents associated with one class is low. When this is the case, accuracy does a poor job at model evaluation. Consider the case when 99 percent of documents are non-political, and 1 percent are about politics. A model which simply predicts that all documents belong to the non-politics class would have an accuracy score of 0.99, but would be poorly suited to the actual classification task. In contrast, the precision and recall rates would be 0, which would signal to the researcher that the model does a poor job at classifying documents as political. Precision and recall are not perfect measures of model performance, however. There is a fundamental trade-off involved in controlling the false positivity and false negativity rates: you can have few false positives if you are content with an extremely high number of false negatives, and you can have few false negatives if you are content with an extremely high number of false positives.

Recognizing this trade-off, researchers often combine precision and recall scores to find a model that has the optimal balance of the two. One common way of combining the two is an F1 score, which is the harmonic mean of precision and recall. Formally, the F1 score is calculated as:

$$\text{F1} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

The F1 score assigns equal importance to precision and recall, and so a high F1 score would indicate that both the false negativity and false positivity rate are low. It is worth noting these evaluation measures (accuracy, precision, recall, and the F1 score) are computed using labeled data ("ground truth"), which in practice, are available only for a limited subset of the records.

C.2 Supplemental Information for "Validation Performance"

We describe the data, explain decisions regarding pre-processing steps, and present additional results supplementing those presented in the Section "Validation Performance."

C.2.1 Data Description

The following four datasets are used to validate the performance of *activeText*:

BBC News: The BBC News Dataset is a collection of 2,225 documents from 2004 to 2005 available at the BBC news website (Greene and Cunningham, 2006). This dataset is divided equally into five topics: business, entertainment, politics, sport, and technology. For example, the binary classification exercise is to correctly predict whether or not an article belongs to the 'politics' topic.

Wikipedia Toxic Comments: The Wikipedia Toxic Comments dataset is a dataset made up of conversations between Wikipedia editors in Wikipedia's internal forums. The dataset was made openly available as part of a Kaggle competition,[8] and was used as a principle dataset of investigation by Miller et al. (2020). The basic classification task is to label a given speech as toxic or not, where toxicity is defined as including harassment and/or abuse of other users.[9] The complete dataset is comprised of approximately 560,000 documents, where roughly 10 percent of which are labeled as toxic.

Supreme Court Cases: The Supreme Court Rulings dataset is a collection of the text of 2000 US Supreme Court rulings between 1946 and 2012. We use the majority opinion of each case and the text was obtained through Caselaw Access Project.[10] For the classification label, we use the categories created by the Supreme Court Database.[11] The classification exercise here is to correctly identify rulings that are categorized as 'criminal procedure', which is the largest category in the corpus (26% of all rulings).

Human Rights Allegation: Human Rights Allegation dataset contains more than 2 million sentences of human rights reports in 196 countries between 1996 and 2016, produced by

[8] See https://www.kaggle.com/c/jigsaw-toxic-comment-classification-challenge

[9] While the dataset also contains finer gradation of 'types' of toxicity, we like Miller et al. (2020) stick to the binary toxic-or-not classification task.

[10] https://case.law

[11] For a full list of categories, see http://www.supremecourtdatabase.org/documentation.php?var=issueArea.

Amnesty International, Human Rights Watch and the US State Department (Cordell et al., 2022). The classification goal is to identify sentences with physical integrity rights allegation (16% of all reports). Example violations of physical integrity rights include torture, extrajudicial killing, and arbitrary arrest and imprisonment.

C.2.2 Pre-processing Steps

We employ the same pre-processing step for each of the four datasets using the *R* package *Quanteda*.[12] For each dataset, we construct a *document-feature matrix* (DFM), where each row is a document and each column is a feature. Each feature is a stemmed unigram. We remove stopwords, features that occur extremely infrequently, as well as all features under 4 characters.

To generate datasets with a fixed proportion p (e.g. 5% or 50%) of the main class of interest, we randomly sample documents from the original dataset so that it achieves the proportion p. Let's use the BBC dataset as an example. Suppose the number of documents in the BBC dataset is N with N_1 and N_0 representing the total number of documents about politics and non-politics in the original data, respectively. We compute $M_1 = \text{floor}(Np)$ and $M_0 = N - M_1$ as the number of sampled politics and non-politics documents, respectively.[13] When $M_1 > N_1$ or when $M_0 > N_0$, we decrement M_1 and M_0 keeping the politics proportion (p) to its target value. When $M_1 < N_1$ and $M_0 < N_0$, we sample M_1 politics documents and M_0 non-politics documents from the original dataset. Finally, combine the sampled politics and non-politics documents to obtain the final dataset. This sampling scheme is applied to all the other datasets when generating the validation datasets with a fixed proportion p.

[12] See https://quanteda.io

[13] The floor function takes as input a real number x and returns the greatest integer larger or equal to x. For example, the floor(3.1415) is 3.

Note that in the main text, to facilitate exposition, we use the terms political and non-political to describe the problem of binary classification. Without loss of generality, in the sections below, we use the (more generic) positive vs. negative class dichotomy instead.

In addition, note that for a matrix $\mathbf{X}$, we denote its ath row by $\mathbf{X}_{a\cdot}$. In other words, we use the subscript $a\cdot$ to denote the ath row of $\mathbf{X}$. Similarly, we use the subscript $\cdot b$ to refer to the bth column of $\mathbf{X}$.

C.3 Binary Classification: EM Algorithm

We now describe how we estimate the parameters of the probabilistic model behind our approach. As mentioned in the main text, our approach is built upon the work of Nigam et al. (2000), which showed that probabilistic classifiers can be enhanced by integrating information from both labeled and unlabeled data.

Let's consider the task of classifying documents into one of two classes (e.g., positive vs. negative). However, as we discuss below, our approach can be extended to accommodate: 1) a multi-class classification setting where $K > 2$ and each document is assigned to one of K classes, such as classifying news articles into politics, business, and sports (refer to SI C.4), and 2) modeling more than two classes while maintaining a binary final classification output (refer to SI C.5).

Under the same model structure defined in Section "The Method" of the main text, let $\mathbf{D}^{lp}$, $\mathbf{D}^{ln}$ and $\mathbf{D}^{u}$ be the document feature matrices for documents with positive labels, documents with negative labels, and unlabeled documents, respectively. Also let N^{lp}, N^{ln}, and N^{u} be the number of documents with positive labels, negative labels, and documents without labels.[14] Likewise, $\mathbf{C}^{lp}$ and $\mathbf{C}^{ln}$ be the vectors of positive and negative labels. Then, the observed-data likelihood is:

[14]Consequently, the number of labeled documents N^{l} is equal to $N^{lp} + N^{ln}$.

$$
\begin{aligned}
&p(\pi, \boldsymbol{\eta}|\mathbf{D}, \mathbf{C}^{lp}, \mathbf{C}^{ln}) \\
&\propto p(\pi)p(\boldsymbol{\eta})p(\mathbf{D}^{lp}, \mathbf{C}^{lp}|\pi, \boldsymbol{\eta})p(\mathbf{D}^{ln}, \mathbf{C}^{ln}|\pi, \boldsymbol{\eta})\Big[p(\mathbf{D}^{u}|\pi, \boldsymbol{\eta})\Big]^{\lambda} \\
&= p(\pi)p(\boldsymbol{\eta}) \times \prod_{i=1}^{N^{lp}} p(\mathbf{D}^{lp}_{i\cdot}|Z_i = 1, \boldsymbol{\eta}_{\cdot 1})p(Z_i = 1|\pi) \times \prod_{i=1}^{N^{ln}} \Big\{p(\mathbf{D}^{ln}_{i\cdot}|Z_i = 0, \boldsymbol{\eta}_{\cdot 0})p(Z_i = 0|\pi)\Big\} \\
&\quad \times \left[\prod_{i=1}^{N^{u}} \Big\{p(\mathbf{D}^{u}_{i\cdot}|Z_i = 1, \boldsymbol{\eta}_{\cdot 1})p(Z_i = 1|\pi) + p(\mathbf{D}^{u}_{i\cdot}|Z_i = 0, \boldsymbol{\eta}_{\cdot 0})p(Z_i = 0|\pi)\Big\}\right]^{\lambda} \\
&\propto \underbrace{\Big\{(1-\pi)^{\alpha_0 - 1}\prod_{v=1}^{V}\eta_{v0}^{\beta_{v0}-1}\Big\} \times \Big\{\pi^{\alpha_1 - 1}\prod_{v=1}^{V}\eta_{v1}^{\beta_{v1}-1}\Big\}}_{\text{prior}} \times \underbrace{\prod_{i=1}^{N^{lp}}\Big\{\prod_{v=1}^{V}\eta_{v1}^{D_{iv}} \times \pi\Big\}}_{\text{positive labeled doc. likelihood}} \\
&\quad \times \underbrace{\prod_{i=1}^{N^{ln}}\Big\{\prod_{v=1}^{V}\eta_{v0}^{D_{iv}} \times (1-\pi)\Big\}}_{\text{negative labeled doc. likelihood}} \times \underbrace{\left[\prod_{i=1}^{N^{u}}\Big\{\prod_{v=1}^{V}\eta_{v0}^{D_{iv}} \times (1-\pi)\Big\} + \Big\{\prod_{v=1}^{V}\eta_{v1}^{D_{iv}} \times \pi\Big\}\right]^{\lambda}}_{\text{unlabeled doc. likelihood}}
\end{aligned}
\tag{C.1}
$$

As described in Section "The Method" of the main text, we weigh the part of the observed likelihood that refers to the unlabeled document with $\lambda \in [0, 1]$. This is done because we typically have many more unlabeled documents than labeled documents. By downweighting the information from the unlabeled document (i.e., setting λ to be small), we can use more reliable information from labeled documents than from unlabeled documents.

We estimate the parameters using EM algorithm Dempster et al. (1977b). Note that by taking the expectation of the complete-data log-likelihood function (Q function), we have that:

$$
\begin{aligned}
Q &\equiv \mathbb{E}_{\mathbf{Z}|\pi^{(t)}, \boldsymbol{\eta}^{(t)}, D, C}[\log p(\pi, \boldsymbol{\eta}, \mathbf{Z}|\mathbf{D}, \mathbf{C})] \\
&= (\alpha_0 - 1)\log(1 - \pi^{(t)}) + (\alpha_1 - 1)\log \pi^{(t)} + \sum_{v=1}^{V}\Big\{(\beta_{v0} - 1)\log \eta_{v0}^{(t)} + (\beta_{v1} - 1)\log \eta_{v1}^{(t)}\Big\} \\
&\quad + \sum_{i=1}^{N^{lp}}\Big\{\sum_{v=1}^{V} D_{iv}\log \eta_{v1}^{(t)} + \log \pi^{(t)}\Big\} + \sum_{i=1}^{N^{ln}}\Big\{\sum_{v=1}^{V} D_{iv}\log \eta_{v0}^{(t)} + \log(1 - \pi^{(t)})\Big\} \\
&\quad + \lambda\left[\sum_{i=1}^{N^{u}} p_{i0}\Big\{\sum_{v=1}^{V} D_{iv}\log \eta_{v0}^{(t)} + \log(1 - \pi^{(t)})\Big\} + p_{i1}\Big\{\sum_{v=1}^{V} D_{iv}\log \eta_{v1}^{(t)} + \log \pi^{(t)}\Big\}\right]
\end{aligned}
\tag{C.2}
$$

where p_{ik} is the probability of a document i being assigned to the kth class, $k = \{0, 1\}$, given data and the parameters at t th iteration. If a document has been hand-coded and has a positive label, we have that $p_{i1} = 1$. If a document has been hand-coded and has been assigned a negative label, we have that $p_{i0} = 0$. If a document has no label, then:

$$\begin{aligned} p_{i0} &= 1 - p_{i1} \\ p_{i1} &= \frac{\prod_{v=1}^{V} \eta_{v1}^{D_{iv}} \times \pi}{\prod_{v=1}^{V} \left\{\eta_{v0}^{D_{iv}} \times (1-\pi)\right\} + \prod_{v=1}^{V} \left\{\eta_{v1}^{D_{iv}} \times \pi\right\}} \end{aligned} \tag{C.3}$$

Equation C.3 also works as the prediction equation. The predicted class of a document i is k that maximizes this probability.

In the M-step, we maximize the Q function, and obtain the updating equations for π, η_{v1}, and η_{v0}. The updating equation for π is the following:

$$\hat{\pi}^{(t+1)} = \frac{\alpha_1 - 1 + N^{lp} + \lambda \sum_{i=1}^{N^u} p_{i1}}{\left(\alpha_1 - 1 + N^{lp} + \lambda \sum_{i=1}^{N^u} p_{i1}\right) + \left(\alpha_0 - 1 + N^{ln} + \lambda \sum_{i=1}^{N^u} p_{i0}\right)} \tag{C.4}$$

The updating equations for each η_{vk} for $k \in \{0, 1\}$ and $v \in \{1, \ldots, V\}$, are as follows:

$$\begin{aligned} \hat{\eta}_{v0}^{(t+1)} &\propto (\beta_{v0} - 1) + \sum_{i=1}^{N^{ln}} D_{iv} + \lambda \sum_{i=1}^{N^u} p_{i0} D_{iv}, \quad v = 1, \ldots, V \\ \hat{\eta}_{v1}^{(t+1)} &\propto (\beta_{v1} - 1) + \sum_{i=1}^{N^{lp}} D_{iv} + \lambda \sum_{i=1}^{N^u} p_{i1} D_{iv}, \quad v = 1, \ldots, V \end{aligned} \tag{C.5}$$

We continue cycling between the E-step and M-step until we reach convergence. In Figure C.1, we graphically present the mixture model at the core of our active learning algorithm. In this representation, hyperparameters are shown in shaded rectangles, random variables and parameters in unshaded circles, and observed data in shaded circles. The larger unshaded rectangular plates are used to indicate the scope of indices for various types of data and parameters.

In Section "An Active Learning Algorithm," we explain how we integrate this model into our algorithm. In each iteration, we follow these steps: first, we fit the model, then we learn its parameters, next, we label the most uncertain cases, and finally, we refit the model with the newly labeled data. We repeat this cycle until a stopping rule is met. We present the implementation of our active learning algorithm as pseudocode in Algorithm **??**.

1: **Input:** Labeled data $\mathbf{D}^l$, unlabeled data $\mathbf{D}^u$, parameters $\boldsymbol{\alpha}$, $\boldsymbol{\beta}$
2: **Output:** Maximize $p(\pi^{(t)}, \boldsymbol{\eta}^{(t)} \mid \mathbf{D}^l, \mathbf{Z}^l, \mathbf{D}^u, \boldsymbol{\alpha}, \boldsymbol{\beta})$

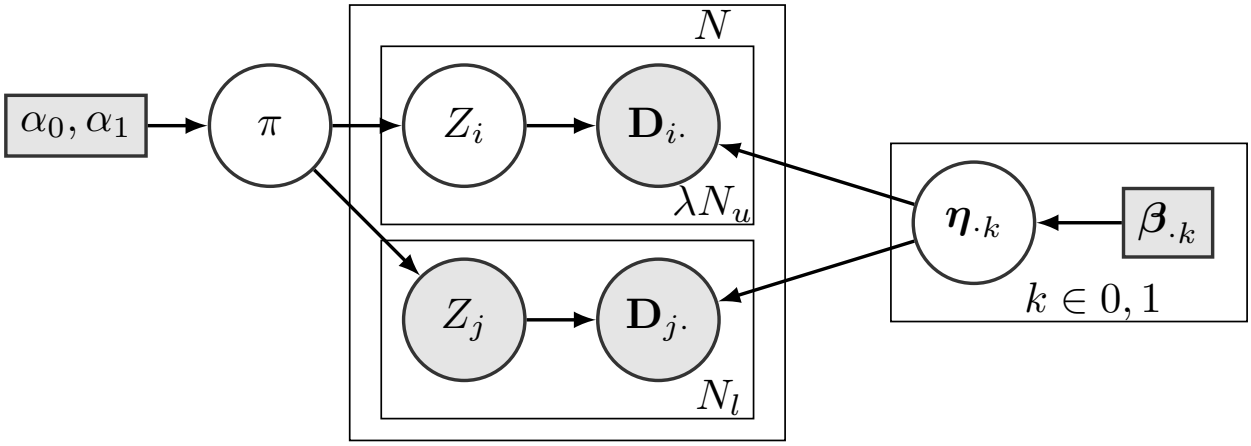


Figure C.1: **Graphical Representation of the Mixture Model with Two Classes.** Hyperparameters are represented by shaded rectangles, while unobserved random variables and parameters are depicted by unshaded circles. Observed data is displayed within shaded circles. Larger unshaded rectangular plates are used to denote the range of indices for different types of data and parameters.

if In the first iteration of Active learning **then**
    Initialize π and $\boldsymbol{\eta}$ by Naive Bayes
    $\pi^{(0)} \leftarrow \text{NB}(\mathbf{D}^l, \mathbf{Z}^l, \boldsymbol{\alpha})$
    $\boldsymbol{\eta}^{(0)} \leftarrow \text{NB}(\mathbf{D}^l, \mathbf{Z}^l, \boldsymbol{\beta})$
else
    Inherit $\pi^{(0)}$ and $\boldsymbol{\eta}^{(0)}$ from the previous iteration
end if
while $p(\pi^{(t)}, \boldsymbol{\eta}^{(t)} \mid \mathbf{D}^l, \mathbf{Z}^l, \mathbf{D}^u, \boldsymbol{\alpha}, \boldsymbol{\beta})$ does not converge **do**
    (1) E step: Obtain the class probability for unlabeled documents
    $p(\mathbf{Z}^u \mid \pi^{(t)}, \boldsymbol{\eta}^{(t)}, \mathbf{D}^l, \mathbf{Z}^l, \mathbf{D}^u) \leftarrow \text{E step}(\mathbf{D}^u, \pi^{(t)}, \boldsymbol{\eta}^{(t)})$
    (2) Combine estimated classes for unlabeled docs with known classes
    $p(\mathbf{Z} \mid \pi^{(t)}, \boldsymbol{\eta}^{(t)}, \mathbf{D}^l, \mathbf{Z}^l, \mathbf{D}^u) \leftarrow \text{combine}(\mathbf{D}^l, \mathbf{D}^u, \mathbf{Z}^l, p(\mathbf{Z}^u \mid \pi^{(t)}, \boldsymbol{\eta}^{(t)}, \mathbf{D}^l, \mathbf{Z}^l, \mathbf{D}^u))$
    (3) M step: Maximize $Q \equiv \mathbb{E}[p(\pi, \boldsymbol{\eta}, \mathbf{Z}^u \mid \mathbf{D}^l, \mathbf{Z}^l, \mathbf{D}^u, \boldsymbol{\alpha}, \boldsymbol{\beta})]$
    $\pi^{(t+1)} \leftarrow \text{argmax } Q$
    $\boldsymbol{\eta}^{(t+1)} \leftarrow \text{argmax } Q$
    (4) Check convergence
    Obtain the value of $p(\pi^{(t+1)}, \boldsymbol{\eta}^{(t+1)} \mid \mathbf{D}^l, \mathbf{Z}^l, \mathbf{D}^u, \boldsymbol{\alpha}, \boldsymbol{\beta})$
end while

C.4 Multiclass Classification

The model outlined in Section "The Method," assumes a binary outcome for the classification task. However, consider a scenario where researchers aim to categorize news articles into various classes such as political news, business news, entertainment news, sports news, and technology news. This section introduces a solution specifically designed for this multi-class classification task (where $K \geq 2$) and each document is assigned to one of K classes.

C.4.1 Model

Let K be the number of the classes and is equal to the number of classes to be classified, with $K \geq 2$. In other words, the model presented below is a generalization of the model presented in the main text. The multiclass classification model can be summarized by the following equations:

$$\begin{aligned}
\boldsymbol{\pi} &\sim \text{Dirichlet}(\boldsymbol{\alpha}) && \text{(C.6)}\\
Z_i &\overset{\text{i.i.d}}{\sim} \text{Categorical}(\boldsymbol{\pi}) && \text{(C.7)}\\
\boldsymbol{\eta}_{\cdot k} &\overset{\text{i.i.d}}{\sim} \text{Dirichlet}(\boldsymbol{\beta}_{\cdot k}), \quad k = \{1, \ldots, K\} && \text{(C.8)}\\
\mathbf{D}_{i\cdot} | Z_i = k &\overset{\text{i.i.d}}{\sim} \text{Multinomial}(n_i, \boldsymbol{\eta}_{\cdot k}) && \text{(C.9)}
\end{aligned}$$

In this model setup, we consider a probability vector denoted by $\boldsymbol{\pi} = (\pi_1, \pi_2, \ldots, \pi_K)$, where each π_k represents the proportion of documents in the corpus belonging to class k. This vector $\boldsymbol{\pi}$ is sampled from a Dirichlet distribution with parameters defined by the K-dimensional vector $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \ldots, \alpha_K)$. Then, for each document i, we generate its features from a multinomial distribution based on the total word count n_i and the vector $\boldsymbol{\eta}_{\cdot k}$, which represents the kth column of the $V \times K$ matrix $\boldsymbol{\eta}$. Here, each entry η_{vk} of $\boldsymbol{\eta}_{\cdot k}$ is the probability of observing word v given that the document belongs to class k. The prior distribution of $\boldsymbol{\eta}_{\cdot k}$ is specified by a Dirichlet distribution with hyperparameter vector $\boldsymbol{\beta}_{\cdot k}$ of length V. The word counts of document i are represented by the row vector $\mathbf{D}_{i\cdot}$ of length V. Given that document i belongs to class k (i.e., $Z_i = k$), $\mathbf{D}_{i\cdot}$ is drawn from a multinomial distribution with parameters n_i and $\boldsymbol{\eta}_{\cdot k}$.

In the case where document i is labeled, the sampling of Z_i is not based on a Bernoulli distribution as in the scenario with unlabeled data. Instead, its value is manually assigned. However, apart from this difference, the model structure remains the same for both labeled and unlabeled data. Notably, to give more weight to the information from labeled documents, we introduce a downweighting factor λ when incorporating information from unlabeled documents.

Figure C.2 graphically presents the multiclass mixture model. In this representation, hyperparameters are shown in shaded rectangles, random variables and parameters in unshaded circles, and observed data in shaded circles. The larger unshaded plates are used to indicate the scope of indices for various types of data and parameters.

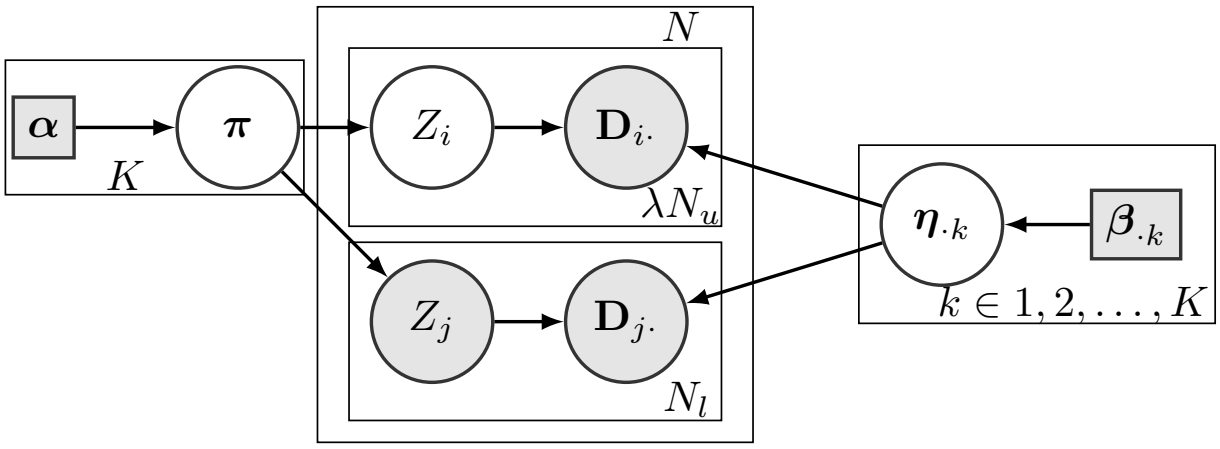


Figure C.2: **Graphical Representation of the Model for Multiclass Classification.** Hyperparameters are represented by shaded rectangles, while unobserved random variables and parameters are depicted by unshaded circles. Observed data is displayed within shaded circles. Larger unshaded rectangular plates are used to denote the range of indices for different types of data and parameters.

C.4.2 EM Algorithm

Equations (C.6) to (C.9) result in the following observed likelihood:

$$
\begin{aligned}
p(\boldsymbol{\pi}, \boldsymbol{\eta} | \mathbf{D}, \mathbf{C}^l) &\propto p(\boldsymbol{\pi}) p(\boldsymbol{\eta}) p(\mathbf{D}, \mathbf{C} | \boldsymbol{\pi}, \boldsymbol{\eta}) \Big[p(\mathbf{D}^u | \boldsymbol{\pi}, \boldsymbol{\eta}) \Big]^{\lambda} \\
&= p(\boldsymbol{\pi}) p(\boldsymbol{\eta}) \times \prod_{k=1}^{K} \prod_{i=1}^{N^k} p(\mathbf{D}^l_{i\cdot} | Z_i = k, \boldsymbol{\eta}_{\cdot k}) p(Z_i = k | \boldsymbol{\pi}) \\
&\quad \times \left[\prod_{i=1}^{N^u} \sum_{k=1}^{K} \Big\{ p(\mathbf{D}^u_{i\cdot} | Z_i = k, \boldsymbol{\eta}_{\cdot k}) p(Z_i = k | \boldsymbol{\pi}) \Big\} \right]^{\lambda} \\
&\propto \underbrace{\prod_{k=1}^{K} \left\{ \pi_k^{\alpha_k - 1} \prod_{v=1}^{V} \eta_{vk}^{\beta_{vk} - 1} \right\}}_{\text{prior}} \times \underbrace{\prod_{k=1}^{K} \prod_{i=1}^{N^k} \Big\{ \prod_{v=1}^{V} \eta_{vk}^{D_{iv}} \times \pi_k \Big\}}_{\text{labeled doc. likelihood}} \times \underbrace{\left[\prod_{i=1}^{N^u} \sum_{k=1}^{K} \Big\{ \prod_{v=1}^{V} \eta_{vk}^{D_{iv}} \times \pi_k \Big\} \right]^{\lambda}}_{\text{unlabeled doc. likelihood}}
\end{aligned}
\tag{C.10}
$$

The Q function (the expectation of the complete-data log-likelihood) is

$$
\begin{aligned}
Q &\equiv \mathbb{E}_{\mathbf{Z} | \boldsymbol{\pi}^{(t)}, \boldsymbol{\eta}^{(t)}, D, C} [\log p(\boldsymbol{\pi}, \boldsymbol{\eta}, \mathbf{Z} | \mathbf{D}, \mathbf{C})] \\
&= \sum_{k=1}^{K} \left[(\alpha_k - 1) \log \pi_k^{(t)} + \sum_{v=1}^{V} \Big\{ (\beta_{vk} - 1) \log \eta_{vk}^{(t)} \Big\} \right] \\
&\quad + \sum_{k=1}^{K} \sum_{i=1}^{N^k} \Big\{ \sum_{v=1}^{V} D_{iv} \log \eta_{vk}^{(t)} + \log \pi_k^{(t)} \Big\} \\
&\quad + \lambda \left[\sum_{i=1}^{N^u} \sum_{k=1}^{K} p_{ik} \Big\{ \sum_{v=1}^{V} D_{iv} \log \eta_{vk}^{(t)} + \log \pi_k^{(t)} \Big\} \right]
\end{aligned}
\tag{C.11}
$$

For the unlabeled documents, the probability of $Z_i = k$, p_{ik}, is given by

$$
p_{ik} = \frac{\prod_{v=1}^{V} \eta_{vk}^{D_{iv}} \times \pi_k}{\sum_{k=1}^{K} \Big[\prod_{v=1}^{V} \eta_{vk}^{D_{iv}} \times \pi_k \Big]} \tag{C.12}
$$

In the M step, the updating equation for π is the following:

$$
\hat{\pi}_k \propto \alpha_k - 1 + N^k + \lambda \sum_{i=1}^{N^u} p_{ik} \tag{C.13}
$$

The updating equation for each η_{vk} is the following:

$$\hat{\eta}_{vk} \propto (\beta_{vk} - 1) + \sum_{i=1}^{N^k} D_{iv} + \lambda \sum_{i=1}^{N^u} p_{ik} D_{iv} \tag{C.14}$$

C.4.3 Results

To evaluate how well *activeText* performs in classifying out-of-sample in a multiclass scenario, we utilize two datasets: BBC news articles and Supreme Court Cases (see SI C.2 for more details about these datasets). In the BBC dataset, the classes are "Politics," "Entertainment," "Business," "Sports," and "Technology." Here, "Politics" represents 5% of the entire dataset, while the remaining 95% is evenly distributed among the other four classes. In the Supreme Court Cases dataset, the classes are "Criminal Procedure" (32.4% of the corpus), "Civil Rights" (21.4%), "Economic Activity" (22.2%), "Judicial Power" (15.4%), and "First Amendment" (8.6%).

We compare the performance of four models: 1) *activeText*, 2) a passive learning variant of *activeText*(Random Mixture), 3) Active Learning with Support Vector Machines (Active SVM), and 4) Passive Learning with Support Vector Machines (Random SVM). Our analysis, illustrated in Figure C.3, demonstrates that *activeText* outperforms the other models across both datasets, even with just 100 labeled documents. Not only does *activeText* exhibit superior classification performance in our validated datasets, but this result is confirmed in the reanalysis of Gohdes (2020) and Park et al. (2020) – both empirical applications involve multi-class classification.

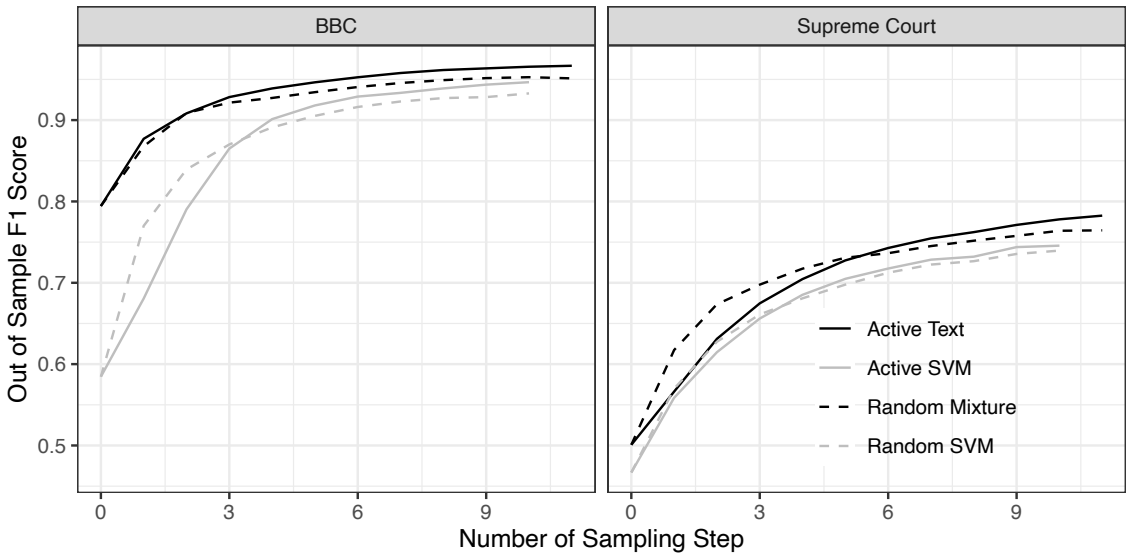


Figure C.3: **Multiclass Classification Results.**
The darker lines show the results with *activeText* and the lighter lines show the results with SVM. The solid lines use active sampling to decide the next set of documents to be labeled, and the dashed lines use random (passive) sampling. The y-axis indicates the out-of-sample F1 score and the x-axis show the number of sampling steps. The left column shows the results on BBC corpus, where the target classes are "Politics," "Entertainment," "Business," "Sports," and "Technology." "Politics" class has 5% of the total dataset, and the rest 95% is evenly split across the rest of classes. The right column shows the results on the Supreme Court corpus, where the target classes are "Criminal Procedure" (32.4% of the corpus), "Civil Rights" (21.4%), "Economic Activity" (22.2%), "Judicial Power" (15.4%), "First Amendment (8.6%)." In our model, we set the number of classes to be the same as the classification categories and linked each class to one classification category. *activeText* performs the best across the four specifications on both corpora.

C.5 Binary Classification With Multiple Classes

The model outlined in the main text (see Section "The Method") assumes that there are two classes (a positive and negative class). However, this assumption can be relaxed to link multiple classes to the negative class. In the world of mixture models, the simplest setup is to let $K = 2$ since the classification goal is binary, and we can link each class to the final classification categories. A more general setup is to use $K > 2$ even when a goal is a binary classification. If $K > 2$, but our focus is to uncover the identity of one class, we can choose one of the class to be linked to the "positive" class and let all other classes be linked to the "negative" class (see e.g., Larsen and Rubin 2001 for a similar idea in the realm of record linkage). In other words, we collapse the $K - 1$ classes into one class for the classification purpose. Using $K > 2$ makes sense if the "negative" class consists of multiple sub-categories.

For instance, suppose researchers are interested in classifying news articles into political news or not. Then, it is reasonable to assume that the non-political news category consists of multiple sub-categories, such as technology, entertainment, and sports news. Note that differently than in SI C.4, in this section, we impose hierarchies between classes and the end goal is binary classification of documents. However, as described below, the model structure is the same, with the collapsing of classes happens at the estimation stage.

C.5.1 Model

This section presents a model and inference algorithm when we use more than 2 classes in estimation but the final classification task is binary. In other words, we impose a hierarchy where many classes are collapsed into the negative class. In contrast, the positive class is made out of just one class. The model presented is as follows:

$$\begin{aligned}
\boldsymbol{\pi} &\sim \text{Dirichlet}(\boldsymbol{\alpha}) && \text{(C.15)}\\
Z_i &\overset{\text{i.i.d}}{\sim} \text{Categorical}(\boldsymbol{\pi}) && \text{(C.16)}\\
\boldsymbol{\eta}_{\cdot k} &\overset{\text{i.i.d}}{\sim} \text{Dirichlet}(\boldsymbol{\beta}_{\cdot k}), \quad k = \{1, \ldots, K\} && \text{(C.17)}\\
\mathbf{D}_{i\cdot} | Z_i = k &\overset{\text{i.i.d}}{\sim} \text{Multinomial}(n_i, \boldsymbol{\eta}_{\cdot k}) && \text{(C.18)}
\end{aligned}$$

It is important to note that if a document is labeled, then we have that the value assigned to Z_i is determined by hand-coding. The model structure is identical to the one presented in SI C.4. The difference, as described below is that the collapsing of the $K-1$ classes that make the negative class happens during the estimation stage. As before, note that we downweight the information from the unlabeled documents by λ, to utilize more information from labeled documents.

C.5.2 EM Algorithm

Let k^* be the class index linked to the positive class. Then, under the structure imposed by equations (C.15) to (C.18), the observed likelihood is the following:

$$
\begin{aligned}
&p(\boldsymbol{\pi}, \boldsymbol{\eta} | \mathbf{D}, \mathbf{C}^{lp}, \mathbf{C}^{ln}) \\
&\propto p(\boldsymbol{\pi}) p(\boldsymbol{\eta}) p(\mathbf{D}^{lp}, \mathbf{C}^{lp} | \boldsymbol{\pi}, \boldsymbol{\eta}) p(\mathbf{D}^{ln}, \mathbf{C}^{ln} | \boldsymbol{\pi}, \boldsymbol{\eta}) \Big[p(\mathbf{D}^{u} | \boldsymbol{\pi}, \boldsymbol{\eta}) \Big]^{\lambda} \\
&= p(\boldsymbol{\pi}) p(\boldsymbol{\eta}) \times \prod_{i=1}^{N^{lp}} p(\mathbf{D}_{i\cdot}^{lp} | Z_i = k^*, \boldsymbol{\eta}_{vk^*}) p(Z_i = k^* | \boldsymbol{\pi}) \\
&\quad \times \prod_{i=1}^{N^{ln}} \sum_{k \neq k^*} \Big\{ p(\mathbf{D}_{i\cdot}^{ln} | Z_i = k, \boldsymbol{\eta}_{v,k}) p(Z_i = k | \boldsymbol{\pi}) \Big\} \times \left[\prod_{i=1}^{N^{u}} \sum_{k=1}^{K} \Big\{ p(\mathbf{D}_{i\cdot}^{u} | Z_i = k, \boldsymbol{\eta}) p(Z_i = k | \boldsymbol{\pi}) \Big\} \right]^{\lambda} \\
&\propto \underbrace{\prod_{k=1}^{K} \left\{ \pi_k^{\alpha_k - 1} \prod_{v=1}^{V} \eta_{vk}^{\beta_{vk} - 1} \right\}}_{\text{prior}} \times \underbrace{\prod_{i=1}^{N^{lp}} \Big\{ \prod_{v=1}^{V} \eta_{vk^*}^{D_{iv}} \times \pi_k \Big\}}_{\text{positive labeled doc. likelihood}} \\
&\quad \times \underbrace{\prod_{i=1}^{N^{ln}} \sum_{k \neq k^*} \Big\{ \prod_{v=1}^{V} \eta_{vk}^{D_{iv}} \times \pi_k \Big\}}_{\text{negative labeled doc. likelihood}} \times \underbrace{\left[\prod_{i=1}^{N^{u}} \sum_{k=1}^{K} \Big\{ \prod_{v=1}^{V} \eta_{vk}^{D_{iv}} \times \pi_k \Big\} \right]^{\lambda}}_{\text{unlabeled doc. likelihood}}
\end{aligned}
\tag{C.19}
$$

The Q function (the expectation of the complete log likelihood) is:

$$
\begin{aligned}
Q &\equiv \mathbb{E}_{\mathbf{Z} | \boldsymbol{\pi}^{(t)}, \boldsymbol{\eta}^{(t)}, D, C} [\log p(\boldsymbol{\pi}, \boldsymbol{\eta}, \mathbf{Z} | \mathbf{D}, \mathbf{C})] \\
&= \sum_{k=1}^{K} \left[(\alpha_k - 1) \log \pi_k^{(t)} + \sum_{v=1}^{V} \Big\{ (\beta_{vk} - 1) \log \eta_{vk}^{(t)} \Big\} \right] \\
&\quad + \sum_{i=1}^{N^{lp}} \Big\{ \sum_{v=1}^{V} D_{iv} \log \eta_{vk^*}^{(t)} + \log \pi_{k^*}^{(t)} \Big\} + \sum_{i=1}^{N^{ln}} \sum_{k \neq k^*} p_{ik} \Big\{ \sum_{v=1}^{V} D_{iv} \log \eta_{vk}^{(t)} + \log \pi_k^{(t)} \Big\} \\
&\quad + \lambda \left[\sum_{i=1}^{N^{u}} \sum_{k=1}^{K} p_{ik} \Big\{ \sum_{v=1}^{V} D_{iv} \log \eta_{vk}^{(t)} + \log \pi_k^{(t)} \Big\} \right]
\end{aligned}
\tag{C.20}
$$

For the unlabeled documents, the probability of $Z_i = k$, p_{ik}, is:

$$
p_{ik} = \frac{\prod_{v=1}^{V} \eta_{vk}^{D_{iv}} \times \pi_k}{\sum_{k=1}^{K} \left[\prod_{v=1}^{V} \eta_{vk}^{D_{iv}} \times \pi_k \right]}
\tag{C.21}
$$

In the M step, the updating equation for π_k is the following:

$$\hat{\pi}_k \propto \begin{cases} \alpha_k - 1 + \sum_{i=1}^{N^{ln}} p_{ik} + \lambda \sum_{i=1}^{N^u} p_{ik} & \text{if } k \neq k^* \\ \alpha_k - 1 + N^{lp} + \lambda \sum_{i=1}^{N^u} p_{ik^*} & \text{if } k = k^* \end{cases} \tag{C.22}$$

The updating equation for η_{vk} is the following.

$$\hat{\eta}_{vk} \propto \begin{cases} (\beta_{vk} - 1) + \sum_{i=1}^{N^{ln}} p_{ik} D_{iv} + \lambda \sum_{i=1}^{N^u} p_{ik} D_{iv} & \text{if } k \neq k^* \\ (\beta_{vk} - 1) + \sum_{i=1}^{N^{lp}} D_{iv} + \lambda \sum_{i=1}^{N^u} p_{ik^*} D_{iv} & \text{if } k = k^* \end{cases} \tag{C.23}$$

C.5.3 Results

In this section, we first evaluate the out-of-sample performance of the model for binary classification, with multiple classes. Then, we evaluate the combined performance of the multiple class approach for binary classification and keyword upweighting.

Figure C.4 shows the results of a model with just two classes vs. a model with 5 classes. In both instances, the final output of the classification task are two classes (positive vs. negative). The darker lines show the results with 5 classes and the lighter lines show the results with 2 classes. Overall, the model with 5 classes performs better or as well as the model with 2 classes. The gain from using 5 classes is the highest when the proportion of positive labels is small and when the size of labeled data is small.

Figure C.5 shows the results when the multiple class approach for binary classification and keyword upweighting approaches are combined. We find that the model with 5 classes yields performance comparable to or slightly better than using 2 classes. The most significant performance improvement occurs with the BBC corpus, comprising 5 news topic categories. Thus, our findings suggest that the classification performance of our method is improved when employing multiple classes, particularly when the number of classes aligns with the data generating process. Finally, regardless of the number of classes, our mixture models incorporating keywords demonstrate superior performance compared to models without keywords. The most notable improvement is observed in the human rights corpus, characterized by the smallest number of words per document.

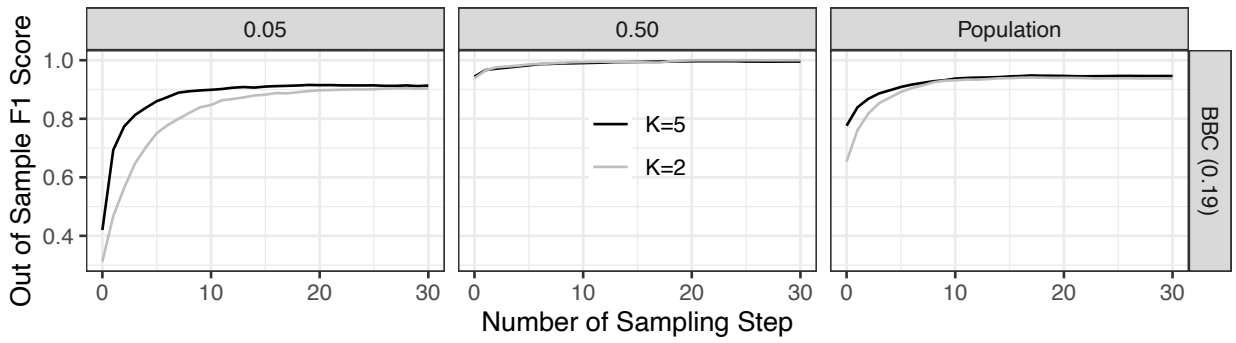


Figure C.4: **Binary Classification Results with 2 and 5 Classes.**
The darker lines show the results with 5 classes and the lighter lines show 2 classes. The columns correspond to various proportions of positive labels in the corpus. The y-axis indicates the out-of-sample F1 score and the x-axis show the number of sampling steps. For binary classification, using multiple classes improves the classification performance when the number of classes matches the data generating process.

C.6 A LURE Approach to Remove The Bias of Active Learning

As described in Section "The Bias of Active Learning" in the main text, active learning introduces statistical biases in training. This problem arises because active learning selects non-i.i.d. data by choosing the most "informative" data points. Farquhar et al. (2021) propose a framework to address this bias, especially in scenarios where each observation has a chance of being actively sampled. Farquhar et al. (2021) proposal, the Levelled Unbiased Risk Estimator (LURE), addresses this bias by adjusting the empirical risk computation using weighted averages based on sampling probabilities of actively selected documents. More specifically, the LURE estimator is defined as follows: $\widetilde{R}_{\text{lure}} = \frac{1}{M}\sum_{m=1}^{M}[v_m L(Z_m - \widehat{Z}_m)]$, where L is a loss function (e.g., the L2 norm), and v_m is calculated as $1 + \frac{N-M}{N-m}\left(\frac{1}{(N-m+1)p(\text{index}_m|\text{index}_{1:m-1},\mathbf{D}^u)} - 1\right)$. Here, $p(\text{index}_m|\text{index}_{1:m-1},\mathbf{D}^u)$ represents the likelihood that the index associated with the mth document is actively sampled, given that the indices of $m-1$ documents have already been actively sampled and $\mathbf{D}^u$ represents the pool of unlabeled documents. In essence, the LURE estimator for the empirical risk computes a weighted average, where the weights are adjusted to account for the sampling probabilities of all actively selected documents.

In this section, we first describe how to incorporate such weights in training the model and recover model parameters that are not subject to these issues. Then, we provide addition details the simulation setting used to test the in-sample and out-of-sample performance of

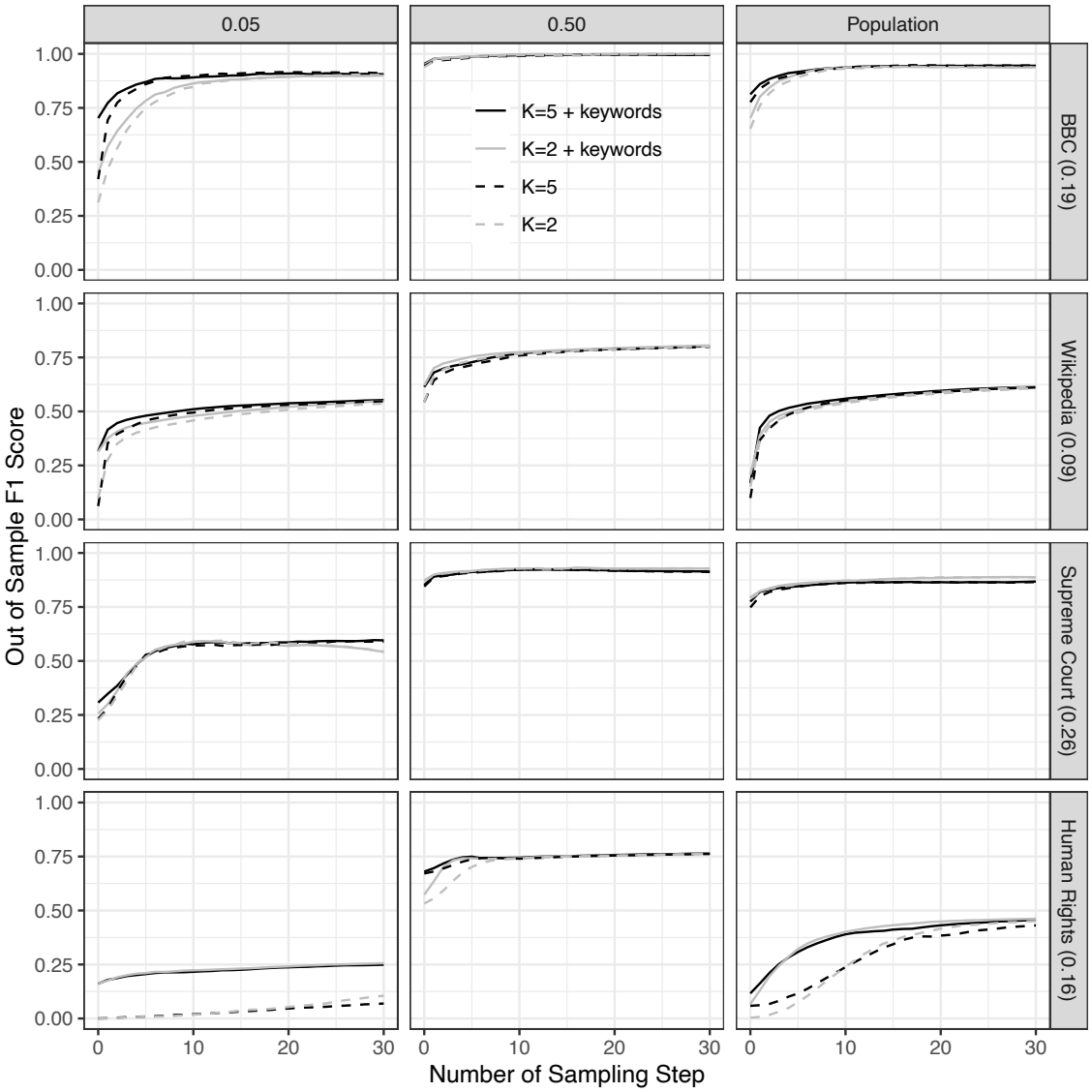


Figure C.5: **Binary Classification Results with Multiple Classes and Keywords.** The rows correspond to different datasets and the columns correspond to various proportions of positively labeled documents in the corpus. The y-axis indicates the out-of-sample F1 score and the x-axis show the number of sampling steps. The linetype show whether keywords are supplied: the solid lines show the results with keywords and the dashed lines without keywords. The colors show the number of classes in the mixture model: the darker lines show the results with 5 classes and the lighter lines with 2 classes. Using 5 classes leads to as good or slightly better performance than using 2 classes. The performance improvement is the largest with the BBC corpus, which consists of 5 news topic categories. Likewise, our mixture models with keywords leads to as good or better performance than the models without keywords. The improvement is the largest with the human rights corpus, where the number of words per document is the smallest.

*activeText*and *activeText*+LURE.

C.6.1 An *activeText* Model with (LURE) Weights

We introduce an extension to *activeText* that enables the use of weights. Let's revisit the binary classification problem, where the model structure aligns with the one described in Section "The Method" in the main text and in SI C.3. The key distinction here is that each labeled observation carries a predetermined weight w_i. Consequently, integrating the LURE weights entails expressing our observed-data quasi-likelihood as follows:

$$
\begin{aligned}
&p(\pi, \boldsymbol{\eta}|\mathbf{D}, \mathbf{C}^{lp}, \mathbf{C}^{ln}, w) \\
&\propto p(\pi)p(\boldsymbol{\eta})p(\mathbf{D}^{lp}, \mathbf{C}^{lp}|\pi, \boldsymbol{\eta}, w)p(\mathbf{D}^{ln}, \mathbf{C}^{ln}|\pi, \boldsymbol{\eta}, w)\Big[p(\mathbf{D}^{u}|\pi, \boldsymbol{\eta})\Big]^{\lambda} \\
&= p(\pi)p(\boldsymbol{\eta}) \times \prod_{i=1}^{N^{lp}} p(\mathbf{D}^{lp}_{i\cdot}|Z_i = 1, \boldsymbol{\eta}_{\cdot 1}, w_i)p(Z_i = 1|\pi, w_i) \times \prod_{i=1}^{N^{ln}} \Big\{p(\mathbf{D}^{ln}_{i\cdot}|Z_i = 0, \boldsymbol{\eta}_{\cdot 0}, w_i)p(Z_i = 0|\pi, w_i)\Big\} \\
&\quad \times \left[\prod_{i=1}^{N^{u}} \Big\{p(\mathbf{D}^{u}_{i\cdot}|Z_i = 1, \boldsymbol{\eta}_{\cdot 1})p(Z_i = 1|\pi) + p(\mathbf{D}^{u}_{i\cdot}|Z_i = 0, \boldsymbol{\eta}_{\cdot 0})p(Z_i = 0|\pi)\Big\}\right]^{\lambda} \\
&\propto \underbrace{\Big\{(1-\pi)^{\alpha_0 - 1}\prod_{v=1}^{V}\eta_{v0}^{\beta_{v0}-1}\Big\} \times \Big\{\pi^{\alpha_1 - 1}\prod_{v=1}^{V}\eta_{v1}^{\beta_{v1}-1}\Big\}}_{\text{prior}} \times \underbrace{\prod_{i=1}^{N^{lp}}\Big\{\prod_{v=1}^{V}\eta_{v1}^{D_{iv}w_i} \times \pi^{w_i}\Big\}}_{\text{positive labeled doc. quasi-likelihood}} \\
&\quad \times \underbrace{\prod_{i=1}^{N^{ln}}\Big\{\prod_{v=1}^{V}\eta_{v0}^{D_{iv}w_i} \times (1-\pi)^{w_i}\Big\}}_{\text{negative labeled doc. quasi-likelihood}} \times \underbrace{\left[\prod_{i=1}^{N^{u}}\Big\{\prod_{v=1}^{V}\eta_{v0}^{D_{iv}} \times (1-\pi)\Big\} + \Big\{\prod_{v=1}^{V}\eta_{v1}^{D_{iv}} \times \pi\Big\}\right]^{\lambda}}_{\text{unlabeled doc. likelihood}}
\end{aligned}
\tag{C.24}
$$

We estimate the parameters π and each η_{vk} using EM algorithm. In the E-step, we take

the expectation of the log complete likelihood function (Q function),

$$
\begin{aligned}
Q &\equiv \mathbb{E}_{\mathbf{Z}|\pi^{(t)},\boldsymbol{\eta}^{(t)},\mathbf{D},\mathbf{C},w}[\log p(\pi,\boldsymbol{\eta},\mathbf{Z}|\mathbf{D},\mathbf{C},w)] \\
&= (\alpha_0-1)\log(1-\pi^{(t)}) + (\alpha_1-1)\log\pi^{(t)} + \sum_{v=1}^{V}\Big\{(\beta_{v0}-1)\log\eta_{v0}^{(t)} + (\beta_{v1}-1)\log\eta_{v1}^{(t)}\Big\} \\
&\quad + \sum_{i=1}^{N^{lp}}\Big\{\sum_{v=1}^{V} w_i\big(D_{iv}\log\eta_{v1}^{(t)} + \log\pi^{(t)}\big)\Big\} + \sum_{i=1}^{N^{ln}}\Big\{\sum_{v=1}^{V} w_i\big(D_{iv}\log\eta_{v0}^{(t)} + \log(1-\pi^{(t)})\big)\Big\} \\
&\quad + \lambda\left[\sum_{i=1}^{N^u} p_{i0}\Big\{\sum_{v=1}^{V} D_{iv}\log\eta_{v0}^{(t)} + \log(1-\pi^{(t)})\Big\} + p_{i1}\Big\{\sum_{v=1}^{V} D_{iv}\log\eta_{v1}^{(t)} + \log\pi^{(t)}\Big\}\right]
\end{aligned}
\tag{C.25}
$$

Again, p_{ik} is the probability of a document i being assigned to the kth cluster, $k=\{0,1\}$, given data and the parameters at t th iteration. If a document has a positive label, $p_{i0}=0$ and $p_{i1}=1$. If a document has no label,

$$
\begin{aligned}
p_{i0} &= 1-p_{i1} \\
p_{i1} &= \frac{\prod_{v=1}^{V}\eta_{v1}^{D_{iv}}\times\pi}{\prod_{v=1}^{V}\left\{\eta_{v0}^{D_{iv}}\times(1-\pi)\right\} + \prod_{v=1}^{V}\left\{\eta_{v1}^{D_{iv}}\times\pi\right\}}
\end{aligned}
\tag{C.26}
$$

In the M-step, we maximize the Q function, and obtain the updating equations for π and each η_{vk} for $k\in\{0,1\}$. The updating equation for π is the following:

$$
\pi^{(t+1)} = \frac{\alpha_1 - 1 + \sum_{i=1}^{N^{lp}} w_i + \lambda\sum_{i=1}^{N^u} p_{i1}}{\left(\alpha_1 - 1 + \sum_{i=1}^{N^{lp}} w_i + \lambda\sum_{i=1}^{N^u} p_{i1}\right) + \left(\alpha_0 - 1 + \sum_{i=1}^{N^{ln}} w_i + \lambda\sum_{i=1}^{N^u} p_{i0}\right)} \tag{C.27}
$$

The updating equation for each η_{vk} are as follows:

$$
\hat{\eta}_{v0}^{(t+1)} \propto (\beta_{v0}-1) + \sum_{i=1}^{N^{ln}} w_i D_{iv} + \lambda\sum_{i=1}^{N^u} p_{i0}D_{iv}, \quad v=1,\ldots,V \tag{C.28}
$$

$$
\hat{\eta}_{v1}^{(t+1)} \propto (\beta_{v1}-1) + \sum_{i=1}^{N^{lp}} w_i D_{iv} + \lambda\sum_{i=1}^{N^u} p_{i1}D_{iv}, \quad v=1,\ldots,V \tag{C.29}
$$

As before, we downweight the information from unlabeled document by λ, to utilize more reliable information from labeled documents.

C.6.2 Simulation Setup to Evaluate the Bias of Active Learning

We now describe in more detail the simulation setting used to produce Figure 4.7 in the main text.

The process of generating simulated data is based on the mixture model that we have defined above (See Section "The Method"). This allows us to systematically vary many dimensions of datasets and examine the performance of our classifier under different settings. We vary the following parameters:.

1. **The number of documents in the corpus** (N): We consider 1000 documents. 800 documents are used for training and 200 documents are used for out-of-sample testing.

2. **The size of vocabulary** (V): We consider two setups, 500 and 5000 words.

3. **The number of words per document**: (n_i): We generate n_i for $i = 1, \ldots, N$ from Poisson distribution with three different means (10, 50, and 100). This means that the average number of words per document is 10, 50, and 100, respectively.

4. **Proportion of documents in the positive class** (π): We consider two setups, 5% and 10%.

5. **Difficulty of classification** ($\tilde{\boldsymbol{\beta}}$): To control the difficulty of classification, we change the way the vector containing the probabilities of observing words for each class ($\boldsymbol{\eta}_{\cdot 0}$ and $\boldsymbol{\eta}_{\cdot 1}$ for negative and positive class, respectively) are generated from the original model. Note that simulating real corpora poses a challenge due to the presence of words that occur in both positive and negative classes, as well as words that are more prevalent in one class compared to the other. For instance, words like "election" and "democracy" are expected to appear frequently in documents labeled as "political" compared to those labeled as "non-political."

   To mimic this scenario, we first draw a "base" parameter vector of length V (the size of the vocabulary), denoted as $\tilde{\boldsymbol{\eta}}$, from a Dirichlet distribution with parameter vector $\tilde{\boldsymbol{\beta}} = (\tilde{\beta}_1, \tilde{\beta}_2, \ldots, \tilde{\beta}_V)$. We initially set $\boldsymbol{\eta}_{\cdot 0} = \boldsymbol{\eta}_{\cdot 1} = \tilde{\boldsymbol{\eta}}$. Subsequently, we exchange the highest values with those associated with the lowest values in the vector $\boldsymbol{\eta}_{\cdot 1}$ while leaving the vector $\boldsymbol{\eta}_{\cdot 0}$ unchanged. This results in $\boldsymbol{\eta}_{\cdot 1}$ and $\boldsymbol{\eta}_{\cdot 0}$ being nearly identical, except for the swapped elements.

   Consequently, the swapped words act as keywords since their values in $\boldsymbol{\eta}_{\cdot 1}$ are entirely distinct from their values in $\boldsymbol{\eta}_{\cdot 0}$. This allows us to control the level of informativeness of the keywords when manipulating the values in $\tilde{\boldsymbol{\beta}}$. For instance, lower values in $\tilde{\boldsymbol{\beta}}$ concentrate the values in $\tilde{\boldsymbol{\eta}}$ towards the edges of the probability space, thereby

making the keywords more informative. Conversely, higher values in $\tilde{\boldsymbol{\beta}}$ distribute the values in $\tilde{\boldsymbol{\eta}}$ more evenly, resulting in less informative keywords and a more challenging classification task. We have two parameters to control for the difficulty of classification: The number of keywords and the hyperparameter vector $\tilde{\boldsymbol{\beta}}$. First, for the number of keywords, or the number of words swapped, we consider three setups: 1, 5, and 20. Second, for the hyperparameter vector $\tilde{\boldsymbol{\beta}}$, we investigate three different values: 0.1, 0.5, and 0.9. As $\tilde{\boldsymbol{\beta}}$ is a vector with a length of V, for each simulation setting we fix all elements of $\tilde{\boldsymbol{\beta}}$ to the same value e.g., $\forall v, \tilde{\beta}_v = 0.1$. Figure **??** illustrates how the values in the word-class matrix $\boldsymbol{\eta} = [\boldsymbol{\eta}_{\cdot 0}; \boldsymbol{\eta}_{\cdot 1}]$ are generated under the different values of $\tilde{\boldsymbol{\beta}}$. As the figure reveals, smaller (larger) values for $\tilde{\boldsymbol{\beta}}$ result in more (less) informative keywords, and, as a result, easier (more difficult) classification tasks.

In our study, we examine a total of 108 distinct simulation setups, obtained by combining different values for five parameters: $V = \{500, 5000\}$, $n_i = \{10, 50, 100\}$, $\pi = \{0.05, 0.1\}$, the number of swapped words $= \{1, 5, 20\}$, and $\tilde{\boldsymbol{\beta}} = \{0.1, 0.5, 0.9\}$. For each setup, we generate 100 datasets.[15] We then train the model using *activeText* and *activeText* with LURE weights. Because the LURE weights can only be computed when all documents have non-zero probability of being labeled, we slightly modified our sampling scheme using the epsilon-greedy algorithm as in Farquhar et al. (2021). In short, we select the document with the highest probability of being labeled with probability $1 - \epsilon$ and select a document at random with probability ϵ. We set $\epsilon = 0.1$. This ensures that the modified algorithm is almost identical to the entropy-based sampling scheme we used throughout the paper, but allows us to compute the LURE weights.

We report two quantities: the in-sample bias of the empirical risk and the out-of-sample average classification metrics, following the approach of Farquhar et al. (2021). For the in-sample bias, we first obtain a sequence of labeled document selected according to our active learning algorithm. After obtaining the sequence of labeled documents, we calculate the empirical risk at each active step both with and without LURE weights. The empirical risk is evaluated at a fixed parameter that do not depend on the training data. This ensures that the bias is only due to the fact that more difficult documents are labeled more likely. We then calculate the bias as the difference between the population risk and the empirical risk with and without LURE weights, respectively.[16] For the out-of-sample average classification metrics, we calculate the average F1 score for each simulation setup using the model trained with and without LURE weights.

Figure C.6 presents additional results from the simulation study aside from Figure 4.7 in

[15] Please refer to Supplementary Appendix for the Simulation Studies for the full set of results.

[16] This is the same approach as Figure 2 in Farquhar et al. (2021)

the main text. While Figure 4.7 in the main text presents the results for the easy classification setting, Figure C.6 presents the results for the difficult classification setting. Specifically, while the dataset used in Figure 4.7 in the main text swap 5 keywords, the dataset used in Figure C.6 swap only 1 keyword. Other simulation settings are the same as the ones used in Figure 4.7 in the main text. Similar to Figure 4.7 in the main text, the results show that the out-of-sample F1 score is higher without LURE weights.

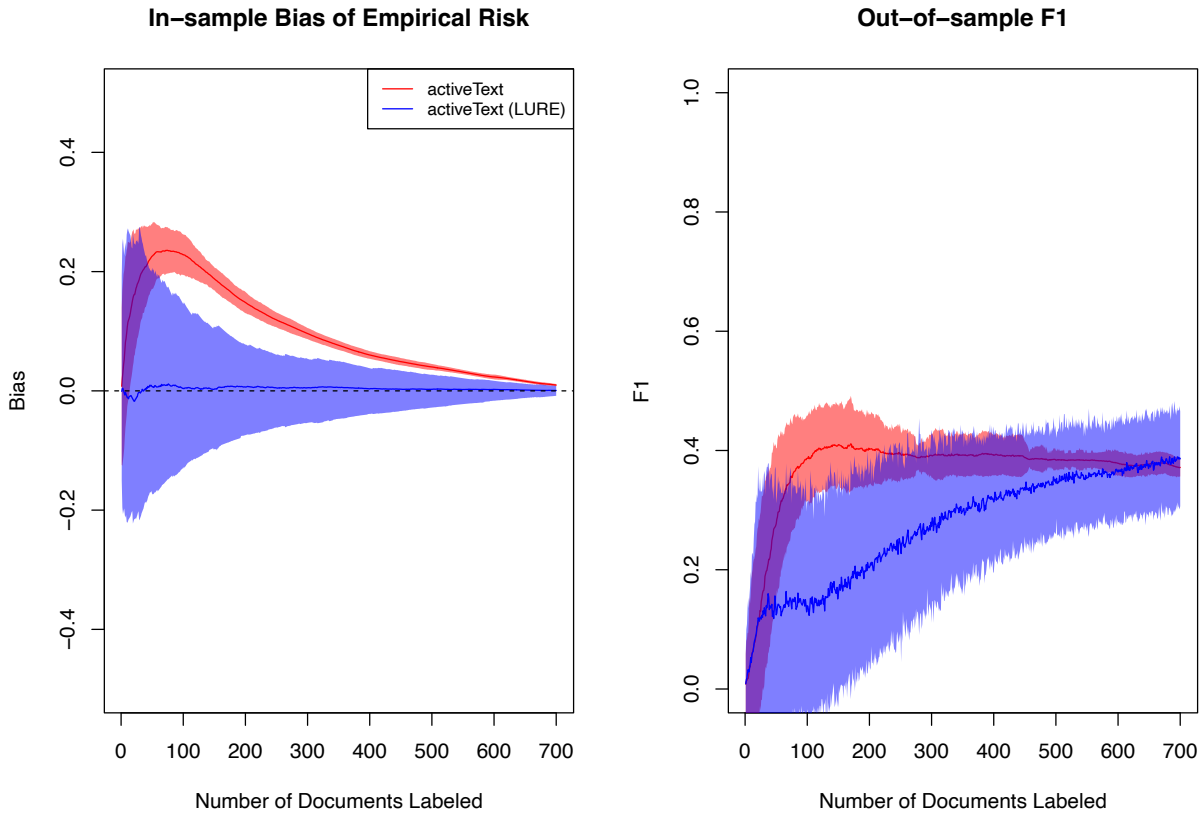


Figure C.6: Bias of the Empirical Risk for Labeled Data (left panel) and Out-of-sample Classification Performance (right panel) of *activeText*, *activeText*+LURE. For each panel, the x-axis represents the number of documents labeled, and the y-axis represents the average bias and average out-of-sample F1 score across 100 Monte Carlo simulations. Shaded areas represent the 95% confidence intervals across Monte Carlo simulations.

Finally, Figure C.7 illustrates the results of applying the LURE bias correction to *activeText* for editors involved in our validation dataset using toxic comment discussions on Wikipedia's internal forum. We divided the data into 80% for training and 20% for testing, conducting 100 Monte Carlo simulations. The left panel displays the in-sample bias of $\widetilde{R}$ and $\widetilde{R}_{lure}$, while the right panel shows their respective out-of-sample F1 scores. The figure confirms our simulation-based findings. Initially, $\widetilde{R}$ shows a positive bias during the labeling

process, which LURE effectively eliminates. However, the unadjusted *activeText* outperforms its bias-corrected version in out-of-sample classification, as depicted in the right panel of Figure C.7.Thus, our validation results suggest that although correcting for in-sample bias improves the performance within the sample, it does not necessarily translate to better out-of-sample classification for *activeText*.

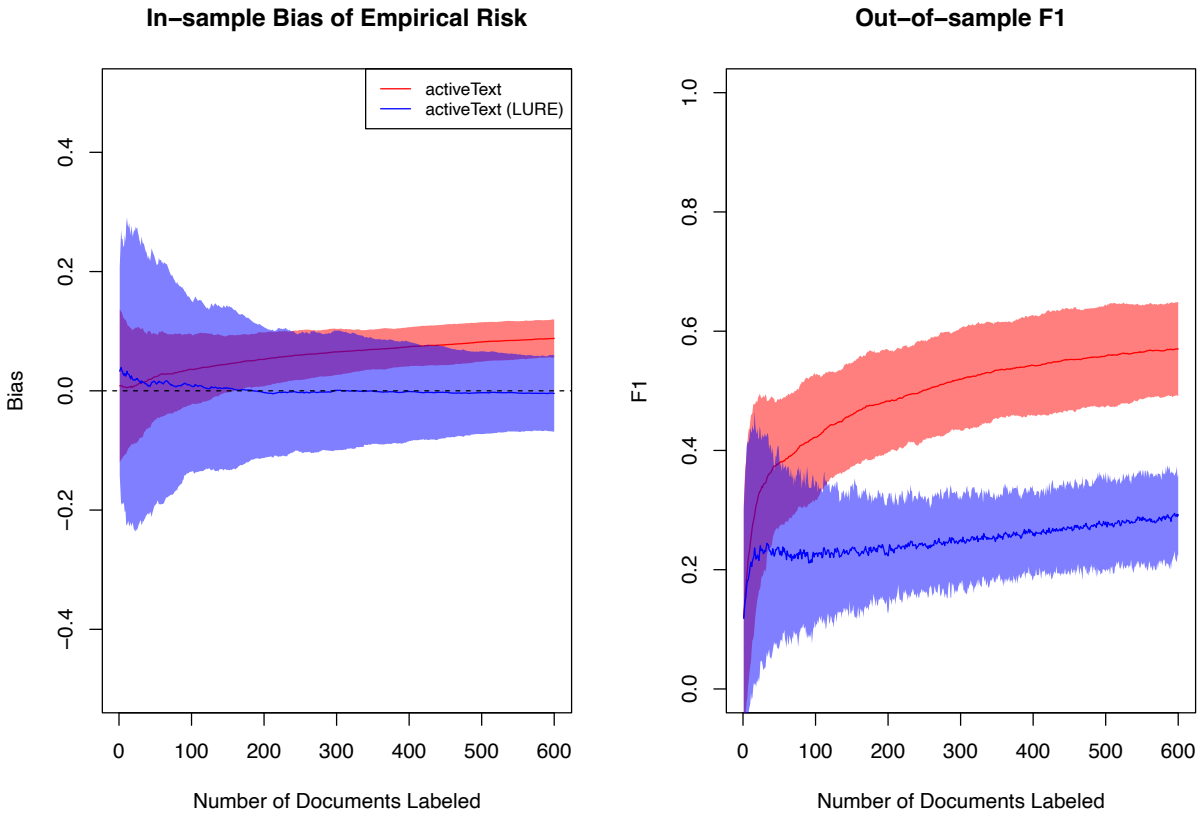


Figure C.7: Wikipedia Toxic Comments: Bias of the Empirical Risk for Labeled Data (left panel) and Out-of-sample Classification Performance (right panel) of *activeText*, *activeText*+LURE. The reference class is Toxic comment (9% of all documents). For each panel, the x-axis represents the number of documents labeled, and the y-axis represents the average bias and average out-of-sample F1 score across 100 Monte Carlo simulations. Shaded areas represent the 95% confidence intervals across Monte Carlo simulations.

C.7 Classification Performance with Mislabels

In this section, we assess the impact of errors introduced during 1) manual document labeling and 2) manual labeling of keywords. To ensure a realistic evaluation, we utilize data derived from our validation studies for binary classification (refer to Section "Validation Performance" in the main text and SI C.2 for data details). The datasets include: internal forum conversations of Wikipedia editors (class of interest: toxic comment), BBC News articles (political topic), the United States Supreme Court decisions (criminal procedure), and Human Rights allegations (physical integrity rights allegation). Again, we use 80% of each dataset for training and and 20% for evaluation. Documents to be labeled are sampled from the training dataset, and documents to test are not included in training. The out-of-sample F1 score is calculated to evaluate model performance. We use $\lambda = 0.001$ to downweight information from unlabeled documents.

C.7.1 Mislabeled Documents

To introduce label noise, we systematically vary the percentage of incorrectly labeled documents. Specifically, for each dataset and at every active iteration (20 labeled documents per iteration), we randomly select θ% of documents chosen by our algorithm for labeling and switch their labels. For example, if a document's true label is politics, we relabel it as non-political, and vice versa. We explore different levels of 'honest' (random) mistakes represented by θ, with values ranging from $\{0, 0.10, 0.20, 0.30, 0.40, 0.50\}$. Note that this process of adding noise uniformly at random, is often referred to as classical measurement error.

As shown in Figure C.8, the classification performance of *activeText* for binary classification, diminishes with an increasing proportion of mislabels. For example, considering the BBC News articles dataset, the out-of-sample F1 score remains high (approximately 0.90) after labeling around 200 documents with no measurement error (θ equal to 0). However, the F1 score decreases to 0.87 when introducing a small amount of measurement error with θ set to 0.10. Notably, the F1 score experiences a significant drop when θ exceeds 0.20. This observed trend holds consistently across all datasets.

What is the effect of labeling a document incorrectly on downstream analyses? For instance, let's say we want to predict how many documents fall into a specific category, like politics. In our validation studies, we already know the true proportions of documents in the classes we are interested in. For example, in the BBC dataset, we find that 19% of the articles are about politics, while in the Wikipedia corpus, 9% of the documents are toxic. Similarly, 26% of Supreme Court cases deal with criminal procedure, and 16% of Human

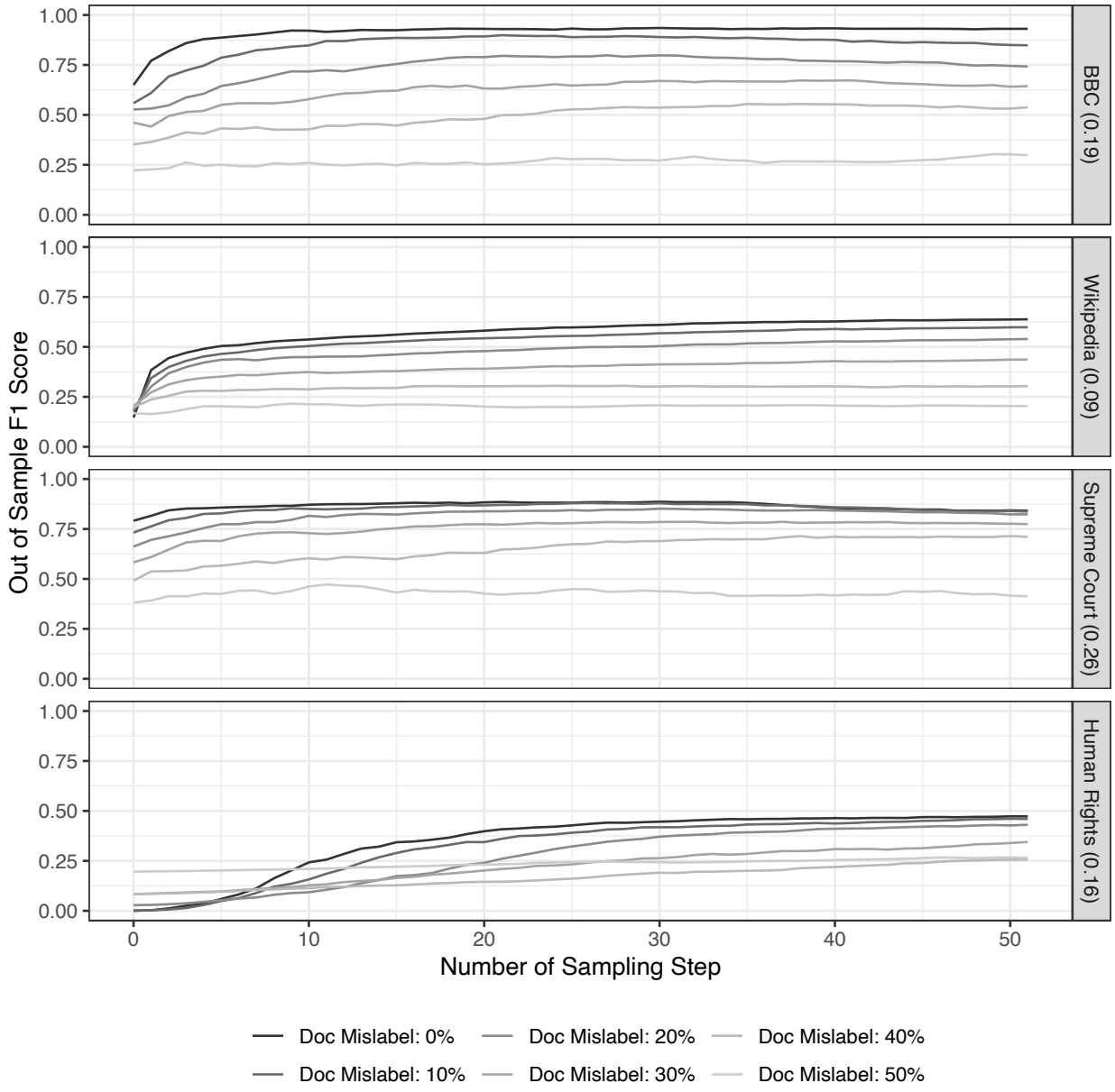


Figure C.8: **Classification Results with Mislabels in Active Document Labeling** The rows correspond to different datasets. The y-axis indicates the out-of-sample F1 score and the x-axis shows the number of sampling steps. 20 documents are labeled at each sampling step. The colors correspond to different levels of mislabels in the labeling of documents. We find that as the proportion of mislabels increases, the classification performance of *activeText* decreases.

Rights reports involve allegations of physical integrity rights violations. To understand the impact of mislabeling, we calculate the absolute value of the bias in the prediction for the proportion of documents in the reference class made by *activeText*. In other words, the

out-of-sample magnitude of the bias in the proportion of documents in the class of interest is calculate as: $\left|\pi - \frac{\sum_{i=1}^{N} p_{i1}}{N}\right|$.

As shown in Figure C.9, the magnitude of the bias increases as the proportion of mislabels rises. For example, in the Human Rights allegations dataset, the bias is small if we label around 200 documents accurately (with no measurement error, θ equal to 0). However, introducing classical measurement error with $\theta > 0.30$ leads to an increase in bias by 0.25 units. This trend holds consistently across all datasets, similar to what we observe with the F1 score. In other words, even when there are unintentional (or "random") errors present, the performance of *activeText* is reduced when calculating a simple summary statistic such as a sample mean.

If our predictions are not accurate, using them either as an outcome or as an explanatory variable in popular downstream tasks (such a regression analysis) could introduce bias in our estimates. The extent of this bias varies. In the best-case scenario, in a regression framework, random errors in explanatory variables result in attenuation bias. However, any departure from classical measurement error could lead to bias whose direction is difficult to determine. This underscores the importance of exercising caution when assessing the limitations of methods based on active learning. Nevertheless, refer to Egami et al. (2023) for an apporach on how to adjust quantities of interest in situations where prediction outcomes are used as outcomes in regression.

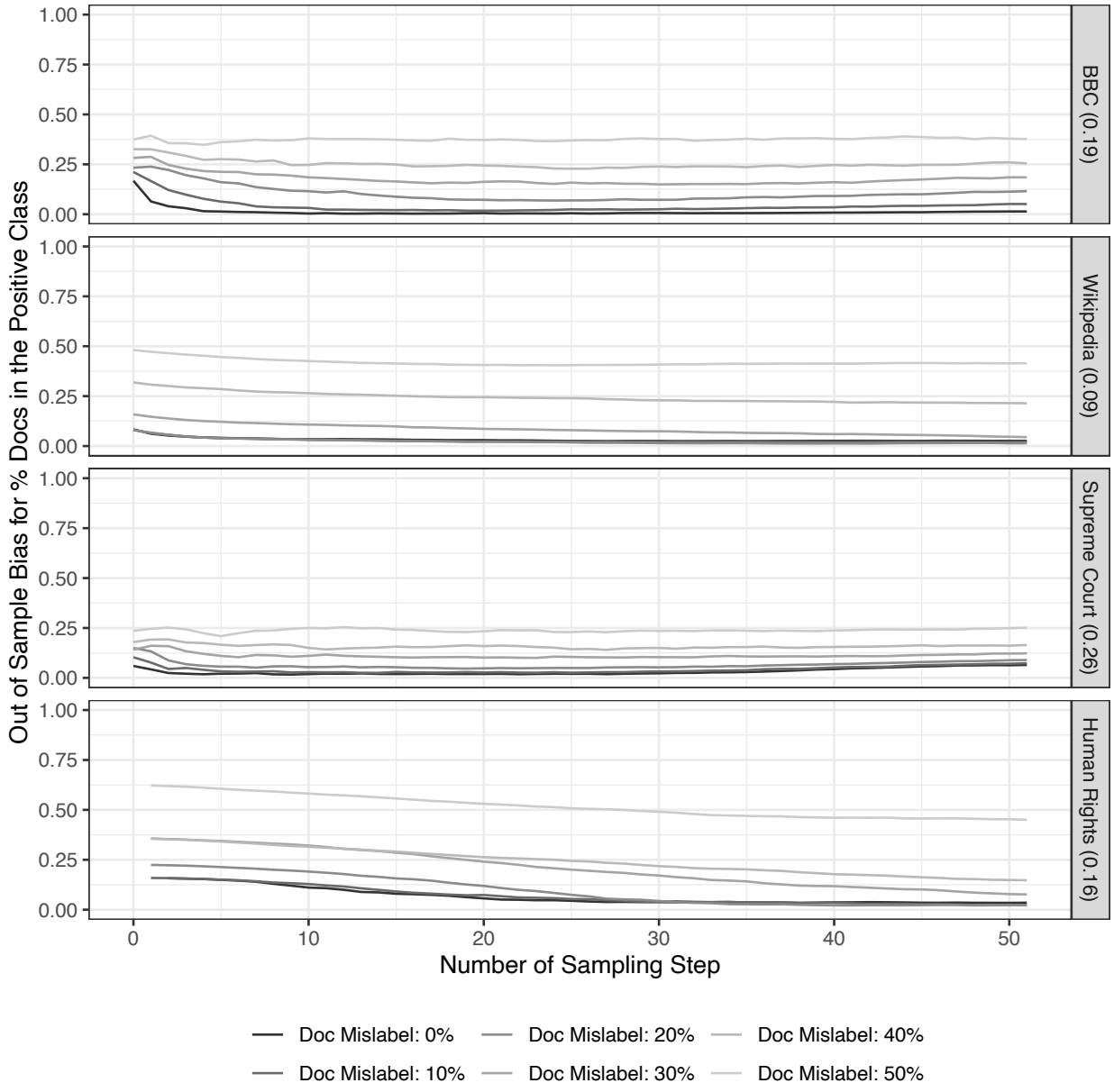


Figure C.9: **Bias in the Proportion of Documents in the Reference Class as a result of Mislabels in Active Document Labeling**
The rows correspond to different datasets. The y-axis indicates the absolute value of the out-of-sample bias and the x-axis shows the number of sampling steps. 20 documents are labeled at each sampling step. The colors correspond to different levels of mislabels in the labeling of documents. We find that as the proportion of mislabels increases, the magnitude of the bias increases as well.

C.7.2 Mislabeled Keywords

We now evaluate the impact of errors that occur during the manual labeling of keywords. First, we need a definition of a "true" keyword. To do so, as we discussed in Section "Benefits of Keyword Upweighting" in the main text, we use all documents and their corresponding labels to obtain the empirical distributions of observing word v in each class. Then, using this information, a keyword v is considered a true keyword for the positive class if the ratio $\eta_{v,1}/\eta_{v,0}$ exceeds the 90th percentile. Remember, for each η_{vk}, the positive class is denoted by $k = 1$ and the negative class by $k = 0$. Similarly, a true negative keyword is defined based on the ratio $\eta_{v,0}/\eta_{v,1}$ and based on the same percentile cutpoint.

Once we have defined the true keywords for each class, the model settings for training and testing are the same ones used are the same as those described in the previous subsection. At each iteration of *activeText*, 10 candidate keywords are suggested. Noise in labeling keywords is introduced by selecting uniformly at random $p\%$ of the candidate keywords to be mislabeled. In other words, the human introduces an honest (random) mistake in labeling with a probability of $p \in \{0, 0.10, 0.20, 0.30\}$. Specifically, if a candidate keyword v is a true keyword, the human will not label v as a keyword. Conversely, if a candidate keyword v is not a true keyword, the human will label v as a keyword. We also vary the value of γ (the parameter representing the upweight for the keywords) to take two values, 10 (small upweight) and 100 (large upweight).

Figure C.10 presents the results. In contrast to the case where noise is added to the labels of the actively sampled documents, the classification performance of *activeText* for binary classification slightly diminishes with an increasing proportion of mislabels for the keywords (30% or more). This is result is consistent across datasets and values of γ (the weight given to each keyword).

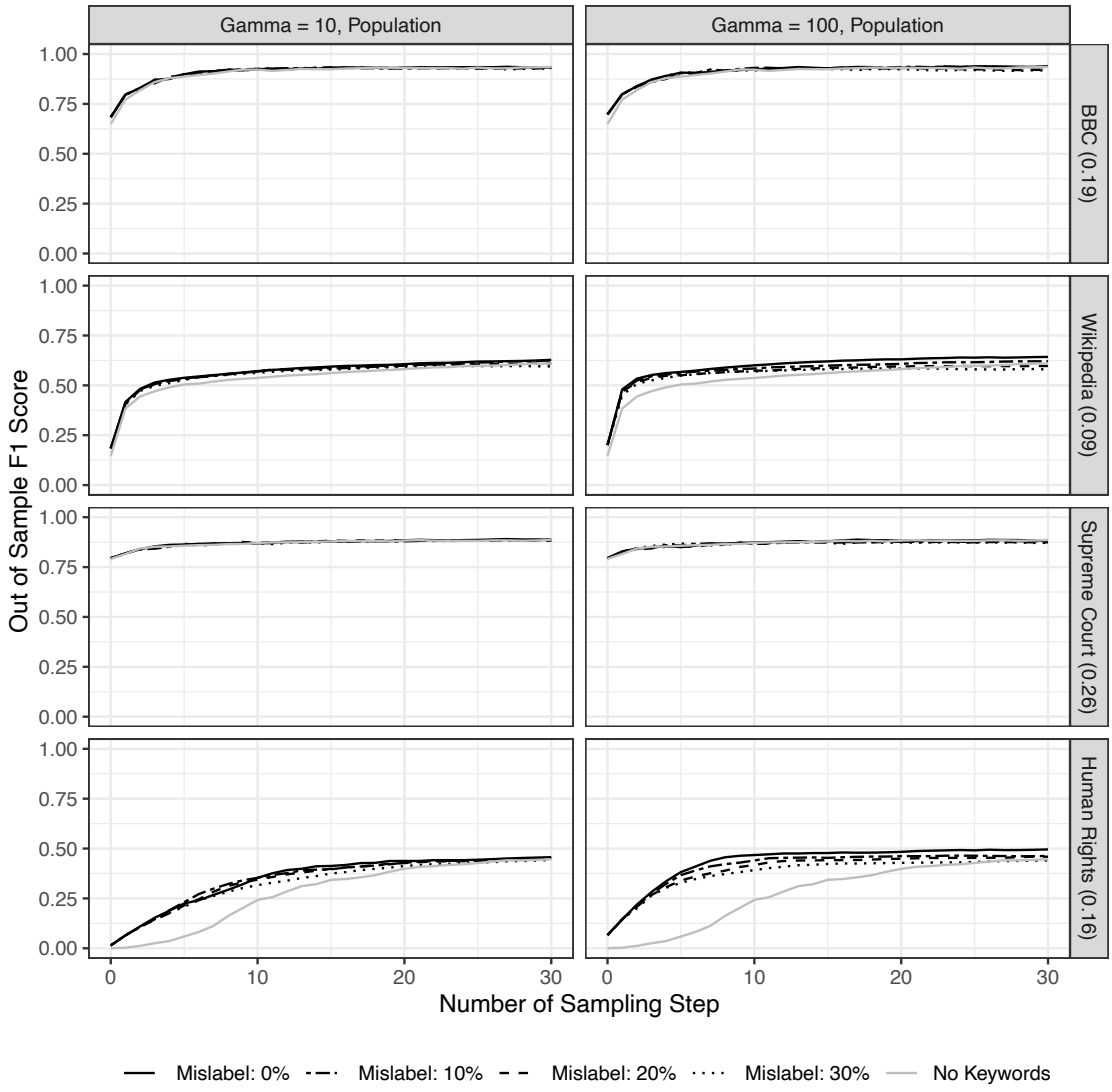


Figure C.10: **Classification Results with Mislabels in Active Keywords**
The rows correspond to different datasets and the columns correspond to two values of γ, which controls the degree of keyword upweighting. The y-axis indicates the out-of-sample F1 score and the x-axis shows the number of sampling steps. The lines correspond to different levels of mislabels at the keyword labeling.

BIBLIOGRAPHY

Airoldi, E., Fienberg, S., and Skinner, K. (2007). Whose ideas? whose words? authorship of ronald reagan's radio addresses. *PS: Political Science & Politics*, 40(3):501–506.

Altschuler, M. and Bloodgood, M. (2019). Stopping active learning based on predicted change of f measure for text classification. In *2019 IEEE 13th International Conference on Semantic Computing (ICSC)*. Newport Beach, CA: IEEE. 47–54.

Anthropic (2023). Introducing claude. *Anthropic Blog*.

Arnold, L. W., Deen, R. E., and Patterson, S. C. (2000). Friendship and votes: The impact of interpersonal ties on legislative decision making. *State and Local Government Review*, 32(2):142–147.

Bächtiger, A., Dryzek, J. S., Mansbridge, J., and Warren, M. (2018a). Studying deliberation empirically: Taking stock and looking forward. In *The Oxford Handbook of Deliberative Democracy*, pages 580–589. Oxford University Press, Oxford.

Bächtiger, A., Dryzek, J. S., Mansbridge, J., and Warren, M. E. (2018b). Taking the goals of deliberation seriously: A differentiated view on equality and equity in deliberative designs and processes. *The Oxford Handbook of Deliberative Democracy*, page 300.

Bächtiger, A., Niemeyer, S., Neblo, M., Steenbergen, M. R., and Steiner, J. (2010). Disentangling diversity in deliberative democracy: Competing theories, their blind spots and complementarities. *Journal of Political Philosophy*, 18(1):32–63.

Bächtiger, A. and Steiner, J. (2005). The deliberative dimensions of legislatures. *Acta Politica*, 40:225–238.

Baumgartner, F. R. and Jones, B. D. (2019). Comparative policy agendas: Theory, measurement, and application. *Oxford University Press*.

Bishop, C. M. and Lassarre, J. (2007). Generative or discriminative? getting the best of both worlds. *Bayesian Statistics*, 8:3–24.

Blei, D. M., Ng, A. Y., and Jordan, M. I. (2003). Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022.

Blitzstein, J. and Hwang, J. (2019). *Introduction to Probability*. Chapman and Hall/CRC, Boca Raton, FL, 2nd edition.

Bommasani, R., Hudson, D. A., Adeli, E., Altman, R., Arora, S., von Arx, S., Bernstein, M. S., Bohg, J., Bosselut, A., Brunskill, E., et al. (2021). On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258*.

Bonica, A. (2015). Mapping the ideological marketplace. *American Journal of Political Science*, 59(2):349–369.

Bosley, M. (2024). Towards qualitative measurement at scale: A prompt-engineering framework for large-scale analysis of deliberative quality in parliamentary debates. Working Paper.

Bosley, M., Kuzushima, S., Enamorado, T., and Shiraito, Y. (2024). Replication Data for: Improving Probabilistic Models in Text Classification via Active Learning.

Boydstun, A. (2013). *Making the News: Politics, the Media, and Agenda Setting*. University of Chicago Press, Chicago, IL.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al. (2020). Language models are few-shot learners. *Advances in Neural Information Processing Systems*, 33:1877–1901.

Bubeck, S., Chandrasekaran, V., Eldan, R., Gehrke, J., Horvitz, E., Kamar, E., Lee, P., Lee, Y. T., Li, Y., Lundberg, S., et al. (2023). Sparks of artificial general intelligence: Early experiments with gpt-4. *arXiv preprint arXiv:2303.12712*.

Buchanan, N. and et al. (2021). Deep learning for open-ended survey response coding. *Artificial Intelligence*, 295.

Caluwaerts, D. and Deschouwer, K. (2014). Deliberative democracy and the challenge of diversity. *Democratic Deliberation in Deeply Divided Societies: From Conflict to Common Ground*, pages 18–34.

Caluwaerts, D. and Reuchamps, M. (2014). Breeding intolerance? exposure to political diversity and the development of political tolerance. *Political Studies*, 62(S1):246–262.

Card, D., Boydstun, A. E., Gross, J. H., Resnik, P., and Smith, N. A. (2015). The media frames corpus: Annotations of frames across issues. *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, 2:438–444.

Catalinac, A. (2016). *Electoral Reform and National Security in Japan: From Pork to Foreign Policy*. Cambridge University Press, New York, NY.

Chamberlin, D. D. and Boyce, R. F. (1974). A sequel to the sequel: Sql. *ACM Transactions on Database Systems (TODS)*, 1(4):327–336.

Chaney, A. J.-B. and Blei, D. M. (2012). Visualizing topic models. *Proceedings of the International AAAI Conference on Web and Social Media*, 6(1):419–422.

Chowdhery, A., Narang, S., Devlin, J., Bosma, M., Mishra, G., Roberts, A., Barham, P., Chung, H. W., Sutton, C., Gehrmann, S., et al. (2022). Palm: Scaling language modeling with pathways. *arXiv preprint arXiv:2204.02311*.

Christiano, P. F., Leike, J., Brown, T., Martic, M., Legg, S., and Amodei, D. (2017). Deep reinforcement learning from human preferences. In *Advances in Neural Information Processing Systems*, pages 4299–4307.

Codd, E. F. (1970). A relational model of data for large shared data banks. *Communications of the ACM*, 13(6):377–387.

Colleoni, E., Rozza, A., and Arvidsson, A. (2014). Echo chamber or public sphere? predicting political orientation and measuring political homophily in twitter using big data. *Journal of Communication*, 64(2):317–332.

Cordell, R., Clay, C., Fariss, C., Wood, R., and Wright, T. (2022). Recording repression: Identifying physical integrity rights allegations in annual country human rights reports. *International Studies Quarterly*, 66(2):sqac016.

Cui, L. and Lee, D. (2020). Coaid: Covid-19 healthcare misinformation dataset. *arXiv preprint arXiv:2006.00885*.

Cui, L., Wu, Y., Liu, S., Zhang, Y., and Zhou, M. (2021). Template-based named entity recognition using bart. *arXiv preprint arXiv:2106.01760*.

Dasgupta, S. (2011). Two faces of active learning. *Theoretical Computer Science*, 412(19):1767–1781.

Dasgupta, S. and Hsu, D. (2008). Hierarchical sampling for active learning. In *ICML '08: Proceedings of the 25th International Conference on Machine Learning*. Helsinki, FI: ICML. 208–215.

Davoodi, R. and et al. (2020). Understanding legislative speeches with bert. *arXiv preprint arXiv:2004.07922*.

Dempster, A., Laird, N., and Rubin, D. (1977a). Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society, Series B*, 39(1):1–38.

Dempster, A., Laird, N., and Rubin, D. (1977b). Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society: Series B*, 39(1):1–22.

Denny, M. J. and Spirling, A. (2018). Text preprocessing for unsupervised learning: Why it matters, when it misleads, and what to do about it. *Political Analysis*, 26(2):168–189.

Devlin, J., Chang, M., Lee, K., and Toutanova, K. (2019a). Bert: Pre-training of deep bidirectional transformers for language understanding. In *Annual Conference of the North American Chapter of the Association for Computational Linguistics*. Minneapolis, MN: ACL. 4171–4186.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019b). Bert: Pre-training of deep bidirectional transformers for language understanding. *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 1:4171–4186.

Dubey, A., Jauhri, A., Pandey, A., Kadian, A., Al-Dahle, A., Letman, A., Mathur, A., Schelten, A., Yang, A., Fan, A., et al. (2024). The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Egami, N., Hinck, M., Stewart, B., and Wei, H. (2023). Using imperfect surrogates for downstream inference: Design-based supervised learning for social science applications of large language models. In *37th Conference on Advances in Neural Information Processing Systems (NeurIPS)*. New Orleans, LA: NEURIPS.

Elman, J. L. (1990). Finding structure in time. *Cognitive Science*, 14(2):179–211.

Eshima, S., Imai, K., and Sasaki, T. (2024). Keyword assisted topic models. *American Journal of Political Science*, 68(2):730–750.

Farquhar, S., Gal, Y., and Rainforth, T. (2021). On statistical bias in active learning: How and when to fix it. In *International Conference on Learning Representations*. Online: ICLR.

Fong, C. and Tyler, M. (2021). Machine learning predictions as regression covariates. *Political Analysis*, 29(4):467–484.

Fowler, J. H. (2006). Connecting the congress: A study of cosponsorship networks. *Political Analysis*, 14(4):456–487.

Friess, D. and Eilders, C. (2015). Analyzing political online communication and deliberation. *The Oxford Handbook of Digital Politics*, pages 477–493.

Garg, N., Schiebinger, L., Jurafsky, D., and Zou, J. (2018). Word embeddings quantify 100 years of gender and ethnic stereotypes. *Proceedings of the National Academy of Sciences*, 115(16):E3635–E3644.

Gelman, A., Carlin, J., Stern, H., Dunson, D., Vehtari, A., and Rubin, D. (2014). *Bayesian Data Analysis*. Chapman and Hall/CRC, Boca Raton, FL, 3rd edition.

Gentzkow, M. and Shapiro, J. M. (2017). Measuring polarization in high-dimensional data: Method and application to congressional speech. *Quarterly Journal of Economics*, 132(4):1633–1685.

Gohdes, A. (2020). Repression technology: Internet accessibility and state violence. *American Journal of Political Science*, 64(3):488–503.

Greene, D. and Cunningham, P. (2006). Practical solutions to the problem of diagonal dominance in kernel document clustering. In *Proc. 23rd International Conference on Machine learning (ICML'06)*, pages 377–384. ACM Press.

Greene, K., Park, B., and Colaresi, M. (2019). Machine learning human rights and wrongs: How the successes and failures of supervised learning algorithms can inform the debate about information effects. *Political Analysis*, 27(2):223–230.

Grimmer, J., Roberts, M., and Stewart, B. (2022). *Text as Data: A New Framework for Machine Learning and the Social Sciences*. Princeton University Press, Princeton, NJ.

Grimmer, J. and Stewart, B. M. (2013). Text as data: The promise and pitfalls of automatic content analysis methods for political texts. *Political Analysis*, 21(3):267–297.

Grossmann, I. and Varnum, M. E. (2015). Explaining the rise in individualism among us residents, 1960–2000. *Social Psychological and Personality Science*, 6(1):81–90.

Guidotti, R., Monreale, A., Ruggieri, S., Turini, F., Giannotti, F., and Pedreschi, D. (2018). A survey of methods for explaining black box models. *ACM computing surveys (CSUR)*, 51(5):1–42.

Gutmann, A. and Thompson, D. F. (2004). *Why Deliberative Democracy?* Princeton University Press.

Habermas, J. (1981). *Theorie des kommunikativen Handelns*, volume 2. Suhrkamp, Frankfurt am Main.

Habermas, J. (1992). *Faktizität und Geltung: Beiträge zur Diskurstheorie des Rechts und des demokratischen Rechtsstaats*. Suhrkamp, Frankfurt am Main.

Habermas, J. (1996). *Between Facts and Norms: Contributions to a Discourse Theory of Law and Democracy*. MIT press.

Halevy, A., Norvig, P., and Pereira, F. (2009). The unreasonable effectiveness of data. *IEEE Intelligent Systems*, 24(2):8–12.

Hangartner, D., Baächtiger, A., Grünenfelder, R., and Steenbergen, M. R. (2007). Empirical analysis of deliberative democracy: The 2005 discourse quality index. *Manuscript, University of Bern*.

Haoran, H. and et al. (2022). Large language models for automated content analysis. *Journal of Communication*.

Hastie, T., Tibshirani, R., and Friedman, J. (2009). *The Elements of Statistical Learning*. Springer, New York, NY.

Hillard, D., Purpura, S., and Wilkerson, J. (2008). Computer-assisted topic classification for mixed-methods social science research. *Journal of Information Technology & Politics*, 4(4):31–46.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8):1735–1780.

Hogan, A. (2022). Knowledge graphs: A guided tour. In *International Research School in Artificial Intelligence in Bergen (AIB 2022)*. Schloss Dagstuhl-Leibniz-Zentrum für Informatik.

Hoi, S., Jin, R., and Lyu, M. (2006). Large-scale text categorization by batch mode active learning. In *WWW 06: Proceedings of the 15th International Conference on World Wide Web*, volume 26. Edinburgh, Scotland: WWW 06. 633–642.

Howard, J. and Ruder, S. (2018). Universal language model fine-tuning for text classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Melbourne, AU: ACL. 328–339.

Ishibashi, H. and Hino, H. (2020). Stopping criterion for active learning based on deterministic generalization bounds. In Chiappa, S. and Calandra, R., editors, *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, volume 108 of *Proceedings of Machine Learning Research*. Online: PLMR. 26–28.

Ji, S., Pan, S., Cambria, E., Marttinen, P., and Philip, S. Y. (2021). A survey on knowledge graphs: Representation, acquisition, and applications. *IEEE transactions on neural networks and learning systems*, 33(2):494–514.

Ji, Z., Lee, N., Frieske, R., Yu, T., Su, D., Xu, Y., Ishii, E., Bang, Y. J., Madotto, A., and Fung, P. (2023). Survey of hallucination in natural language generation. *ACM Computing Surveys*, 55(12):1–38.

Kim, I. S., Londregan, J., and Ratkovic, M. (2018). Estimating spatial preferences from votes and text. *Political Analysis*, 26(2):210–229.

Kim, Y. (2014). Convolutional neural networks for sentence classification. *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pages 1746–1751.

King, G., Pan, J., and Roberts, M. (2017). How the chinese government fabricates social media posts for strategic distraction, not engaged argument. *American Political Science Review*, 111(3):484–501.

Knox, D., Lucas, C., and Cho, W. K. T. (2022). Testing causal theories with learned proxies. *Annual Review of Political Science*, 25(1):419–441.

Knut, J. (2019). Mermaid: A language for visualization of complex systems. *Journal of Visual Languages & Computing*, 52:1–15.

Kojima, T., Gu, S. S., Reid, M., Matsuo, Y., and Iwasawa, Y. (2022). Large language models are zero-shot reasoners. *arXiv preprint arXiv:2205.11916*.

Kotsiantis, S. (2007). Supervised machine learning: A review of classification techniques. *Emerging Artificial Intelligence Applications in Computer Engineering*, 160(1):3–24.

Larsen, M. D. and Rubin, D. B. (2001). Iterative automated record linkage using mixture models. *Journal of the American Statistical Association.*, 96(453):32–41.

Lauderdale, B. E. and Herzog, S. (2016). Measuring ideology with words: A new approach. *American Political Science Review*, 110(4):702–718.

Laver, M., Benoit, K., and Garry, J. (2003). Extracting policy positions from political texts using words as data. In *The Interpretation of Economic and Social Data*, pages 2–3. Wiley.

Lewis, D. and Gale, W. (1994). A sequential algorithm for training text classifiers. In Croft, B. and van Rijsbergen, C., editors, *Proceedings of the Seventeenth Annual International ACM-SIGIR Conference*. London UK: Springer. 3–12.

Li, P., Yao, H., Zhang, Z., Wang, W., Xie, X., Zhang, K.-F., and Chen, D. (2022). Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Computing Surveys*.

Lin, S., Hilton, J., and Evans, O. (2021). Truthfulqa: Measuring how models mimic human falsehoods. *arXiv preprint arXiv:2109.07958*.

Liu, H., Tam, D., Muqeeth, M., Mohta, J., Huang, T., Bansal, M., and Raffel, C. (2022). Few-shot parameter-efficient fine-tuning is better and cheaper than in-context learning. *arXiv preprint arXiv:2205.05638*.

Liu, P., Yuan, W., Fu, J., Jiang, Z., Hayashi, H., and Neubig, G. (2021). Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *arXiv preprint arXiv:2107.13586*.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., and Stoyanov, V. (2019). Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Lowande, K. (2018). Who polices the administrative state? *American Political Science Review*, 112(4):874–890.

Luskin, R. C., Fishkin, J. S., and Jowell, R. (2002). Considered opinions: Deliberative polling in britain. *British Journal of Political Science*, 32(3):455–487.

Mayhew, D. R. (1991). Divided we govern: Party control, lawmaking, and investigations, 1946–1990. *Yale University Press*.

McCallum, A., Nigam, K., et al. (1998). Employing em and pool-based active learning for text classification. In *Proceedings of the Fifteenth International Conference on Machine Learning*, volume 98. San Francisco, CA: ICML. 350–358.

McLachlan, G., Lee, S., and Rathnayake, S. (2019). Finite mixture models. *Annual Review of Statistics and Its Application*, 6(1):355–378.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013). Distributed representations of words and phrases and their compositionality. *Advances in Neural Information Processing Systems*, 26.

Miller, B., Linder, F., and Mebane, W. (2020). Active learning approaches for labeling text: Review and assessment of the performance of active learning approaches. *Political Analysis*, 28(4):532–551.

Miller, D. and Uyar, H. (1996). A mixture of experts classifier with learning based on both labelled and unlabelled data. In Mozer, M., Jordan, M., and Petsche, T., editors, *Advances in Neural Information Processing Systems*, volume 9. Denver, CO: NIPS. 571-577.

Mishra, S., Khashabi, D., Baral, C., and Hajishirzi, H. (2022). Natural instructions v2: Benchmarking generalization to new tasks from natural language instructions. *arXiv preprint arXiv:2204.07705*.

Motolinia, L. (2021). Electoral accountability and particularistic legislation: Evidence from an electoral reform in mexico. *American Political Science Review*, 115(1):97–113.

Neblo, M. A., Esterling, K. M., and Lazer, D. M. (2018). *Politics with the people: Building a directly representative democracy*, volume 555. Cambridge University Press.

Neo4j (2022). Neo4j. *Neo4j*.

Ng, A. and Jordan, M. (2001). On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes. *Advances in neural information processing systems*, 14.

Nielsen, R. (2017). *Deadly Clerics: Blocked Ambition and the Paths to Jihad*. Cambridge University Press, New York, NY.

Nigam, K., McCallum, A. K., Thrun, S., and Mitchell, T. (2000). Text classification from labeled and unlabeled documents using em. *Machine Learning*, 39(2-3):103–134.

OpenAI (2023). Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C. L., Mishkin, P., Zhang, C., Agarwal, S., Slama, K., Ray, A., et al. (2022). Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744.

Park, B., Greene, K., and Colaresi, M. (2020). Human rights are (increasingly) plural: Learning the changing taxonomy of human rights from large-scale text reveals information effects. *American Political Science Review*, 114(3):888–910.

Pennington, J., Socher, R., and Manning, C. D. (2014). Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Peterson, A. and Spirling, A. (2018). Classification accuracy as a substantive quantity of interest: Measuring polarization in westminster systems. *Political Analysis*, 26(1):120–128.

Petroni, F., Rocktäschel, T., Riedel, S., Lewis, P., Bakhtin, A., Wu, Y., and Miller, A. (2019). Language models as knowledge bases? *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, pages 2463–2473.

Proksch, S.-O. and Slapin, J. B. (2010). Position taking in parliamentary questions: An analysis of german parliamentary questions. *Legislative Studies*, 35(1):71–98.

Quinn, K. M., Monroe, B. L., and Colaresi, M. (2010). Analyzing the dynamics of party competition in the u.s. senate: A dynamic topic modeling approach. *Political Analysis*, 18(4):447–475.

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., and Sutskever, I. (2019). Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Radford, A., Wu, K., Child, R., Luan, D., Amodei, D., and Sutskever, I. (2018). Improving language understanding by generative pre-training. *OpenAI*.

Rae, J. W., Borgeaud, S., Cai, T., Millican, K., Hoffmann, J., Song, F., Aslanides, J., Henderson, S., Ring, R., Young, S., et al. (2021). Scaling language models: Methods, analysis & insights from training gopher. *arXiv preprint arXiv:2112.11446*.

Raffel, C., Shinn, J., Painter, B., and et al. (2020). Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv preprint arXiv:1910.10683*.

Reynolds, L. and McDonell, K. (2021). Prompt programming for large language models: Beyond the few-shot paradigm. *Extended Abstracts of the 2021 CHI Conference on Human Factors in Computing Systems*, pages 1–7.

Ribeiro, M. T., Singh, S., and Guestrin, C. (2016). Why should i trust you?: Explaining the predictions of any classifier. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1135–1144.

Rodriguez, P. L. and Spirling, A. (2022). Word embeddings: What works, what doesn't, and how to tell the difference for applied research. *The Journal of Politics*, 84(1):101–115.

Ruder, S., Peters, M., Swayamdipta, S., and Wolf, T. (2019). Transfer learning in natural language processing. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Tutorials*. Minneapolis, MN: ACL. 15–18.

Rudin, C. (2019). Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence*, 1(5):206–215.

Rudkowsky, E., Haselmayer, M., Wastian, M., Jenny, M., Šmejkal, Š., and Emrich, M. (2018). More than bags of words: Sentiment analysis with word embeddings. *Communication Methods and Measures*, 12(2-3):140–157.

Sagi, E. and Deignan, A. (2013). Semantic shift detection across corpora using distributional semantic models. *Proceedings of the 4th International Conference on Corpus Linguistics*, pages 81–91.

Sanh, V., Debut, L., Chaumond, J., and Wolf, T. (2019). Distilbert, a distilled version of bert: Smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*.

Settles, B. (2011). *Synthesis Lectures on Artificial Intelligence and Machine Learning : Active Learning*. Springer, New York, NY.

Shin, T., Razeghi, Y., Logan IV, R. L., Wallace, E., and Singh, S. (2020). Autoprompt: Eliciting knowledge from language models with automatically generated prompts. *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, pages 4222–4235.

Slapin, J. B. and Proksch, S.-O. (2008). Scaling policy preferences from coded speeches: A new approach. In *Political Analysis*, volume 16, pages 49–70. Cambridge University Press.

Soares, N. and Fallenstein, B. (2016). The value learning problem. In *AAAI Workshop: AI, Ethics, and Society*.

Spörndli, M. (2004). *Diskurs und Entscheidung: Eine empirische Analyse kommunikativen Handelns im deutschen Vermittlungsausschuss*. PhD thesis, University of Bern.

Steenbergen, M. R., Bächtiger, A., Spörndli, M., and Steiner, J. (2003). Measuring political deliberation: A discourse quality index. *Comparative European Politics*, 1:21–48.

Steiner, J., Bächtiger, A., Spörndli, M., and Steenbergen, M. R. (2004). Deliberative politics in action: Analyzing parliamentary discourse. *test*.

Steiner, J., Bächtiger, A., Spörndli, M., and Steenbergen, M. R. (2012). Deliberative politics in action: Analyzing parliamentary discourse. *Cambridge University Press*.

Stewart, B. and Zhukov, Y. (2009). Use of force and civil–military relations in russia: an automated content analysis. *Small Wars & Insurgencies*, 20(2):319–343.

Strubell, E., Ganesh, A., and McCallum, A. (2019). Energy and policy considerations for deep learning in nlp. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Florence, ITA: ACL. 3645–3650.

Tausczik, Y. R. and Pennebaker, J. W. (2010). The psychological meaning of words: Liwc and computerized text analysis methods. *Journal of Language and Social Psychology*, 29(1):24–54.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*, 30.

Waugh, A. S., Pei, L., Fowler, J. H., Mucha, P. J., and Porter, M. A. (2009). Party polarization in congress: A network science approach. *arXiv preprint arXiv:0907.3509*.

Weidinger, L., Uesato, J., Rauh, M., Griffin, C., Huang, P.-S., Mellor, J., Glaese, A., Cheng, M., Balle, B., Kasirzadeh, A., et al. (2022). Taxonomy of risks posed by language models. In *Proceedings of the 2022 ACM Conference on Fairness, Accountability, and Transparency*, pages 214–229.

Wilkerson, J., Smith, D., and Stramp, N. (2015). Tracing the flow of policy ideas in legislatures: A text reuse approach. *American Journal of Political Science*, 59(4):943–956.

Yang, Y., Ma, Z., Nie, F., Chang, X., and Hauptmann, A. (2015). Multi-class active learning by uncertainty sampling with diversity maximization. *International Journal of Computer Vision*, 113:113–127.

Yue, Y., Zuo, S., Jiang, H., Ren, W., Zhao, T., and Zhang, C. (2021). Fine-tuning pre-trained language model with weak supervision: A contrastive-regularized self-training approach. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics*. Online: NAACL. 1063–1077.

Zhu, Q., Guo, D., Shao, Z., Yang, D., Wang, P., Xu, R., Wu, Y., Li, Y., Gao, H., Ma, S., et al. (2024). Deepseek-coder-v2: Breaking the barrier of closed-source models in code intelligence. *arXiv preprint arXiv:2406.11931*.

Zhu, X. and Goldberg, A. (2022). *Introduction to Semi-Supervised Learning*. Springer Nature, Berlin, Germany.