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

Mitchell Bosley

September 3, 2024

Abstract

Analyzing the linguistic, psychological, and social dimensions of large textual corpora has traditionally involved a tradeoff between the richness of the constructs measured and the scalability of measurement. While qualitative approaches like expert human coding can capture complex, high-dimensional constructs, they are often too costly and time-consuming to apply to large datasets. Automated computational methods, on the other hand, scale efficiently but typically measure relatively simplistic constructs. I propose a set of novel techniques using large language models (LLMs) to move past this tradeoff, with the goal of enabling rich, qualitative measurement of complex constructs at scale, and show that by carefully designing prompts that imbue LLMs with the knowledge and reasoning abilities of human experts we can elicit high-quality annotations of latent constructs directly from textual data. I apply this approach to the Discourse Quality Index (DQI), a widely used framework for assessing the deliberative quality of political communication, and show that LLMs can automate the coding of the DQI in a sample of parliamentary speeches at a performance level comparable to human annotators. By comparing a human-annotated database of over 1000 speeches from the US Congress to those generated by LLMs, I demonstrate that by carefully designing prompts with a combination of instructions, contextual data, and a handful of high quality examples of the desired annotation behavior, Generative LLMs can "learn" to perform complex, multidimensional annotations of political speech at the level of expert coders, and at a fraction of the time and effort.

Contents

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 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.

2.1 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 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

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

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 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 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

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

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.

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 1: DQI dimensions and scoring criteria

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 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.

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.

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).

[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.

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 1, each dimension is scored on an ordinal scale with higher scores indicating higher deliberative quality.[4]

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 1, where each of the curly-braced components is replaced with the relevant information for the specific annotation task.[5]

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 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 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 4 for the JSON[6] output format

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

[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.

[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.

```
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 1: High level overview of the DQI annotation prompt.

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 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 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.

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

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: 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 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 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 5: Closing Instructions for Speech Annotation Prompt

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 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 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.

[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.

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]

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.

[8]Formally, the MAE is defined as:

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

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.

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.

3.4 Results

3.4.1 Model Performance

Figure 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 6a and by individual dimension 6b).

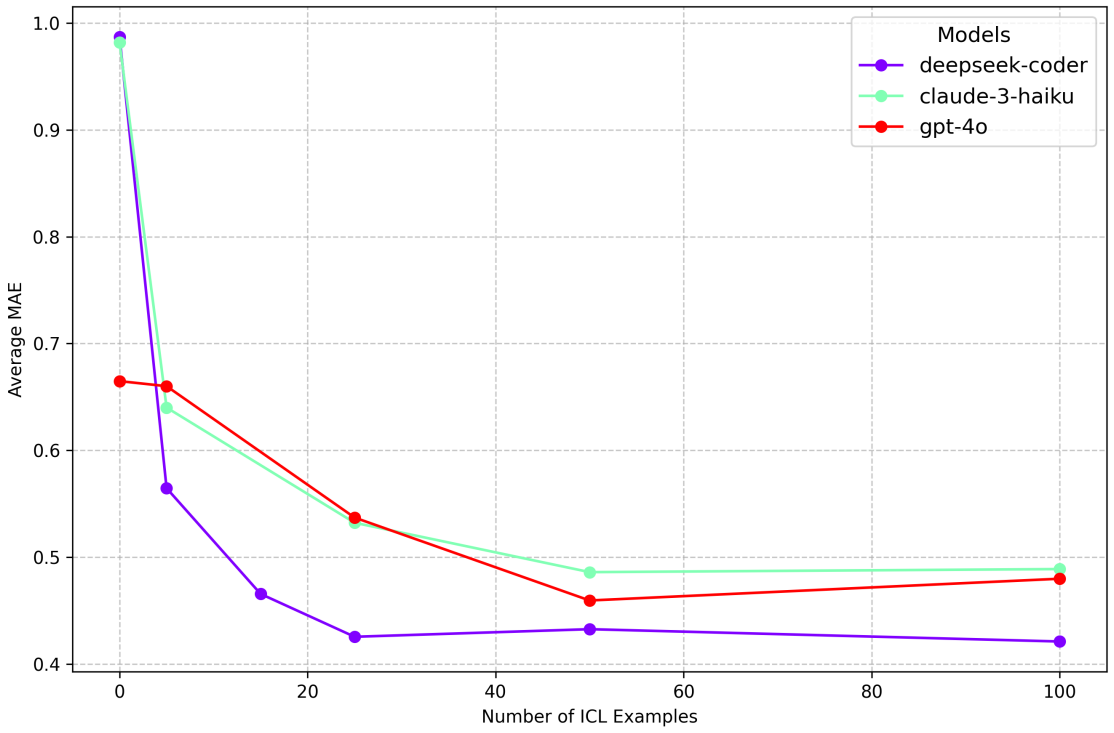
In both of these figures, the x-axis represents the number of in-context learning examples provided to the model, and the y-axis represents the mean absolute error of the model's annotations compared to the expert annotations in the validation dataset. The colored lines represent the performance of different 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 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.

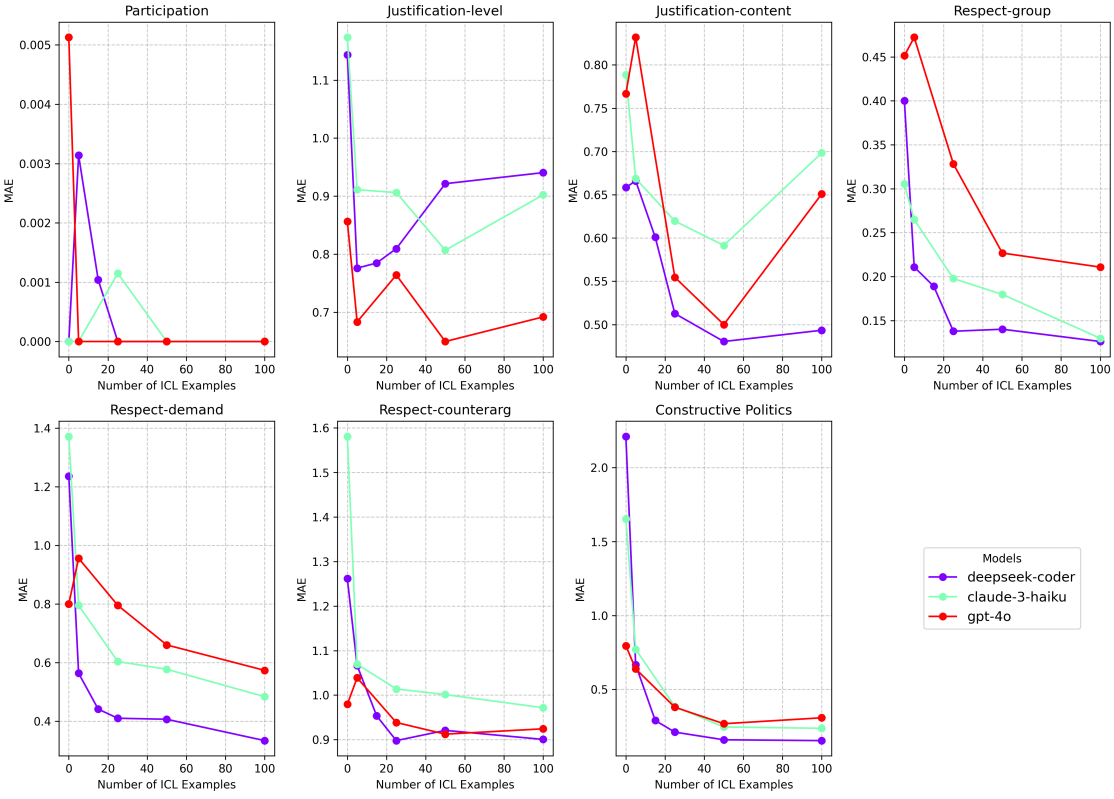
As we move from left to right in the figure, we see that the performance of the models generally improves as the number of ICL examples increases, and that there is variation in the model's rate of improvement with additional examples. GPT-4o, which performed the best with zero-shot learning, surprisingly 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 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



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

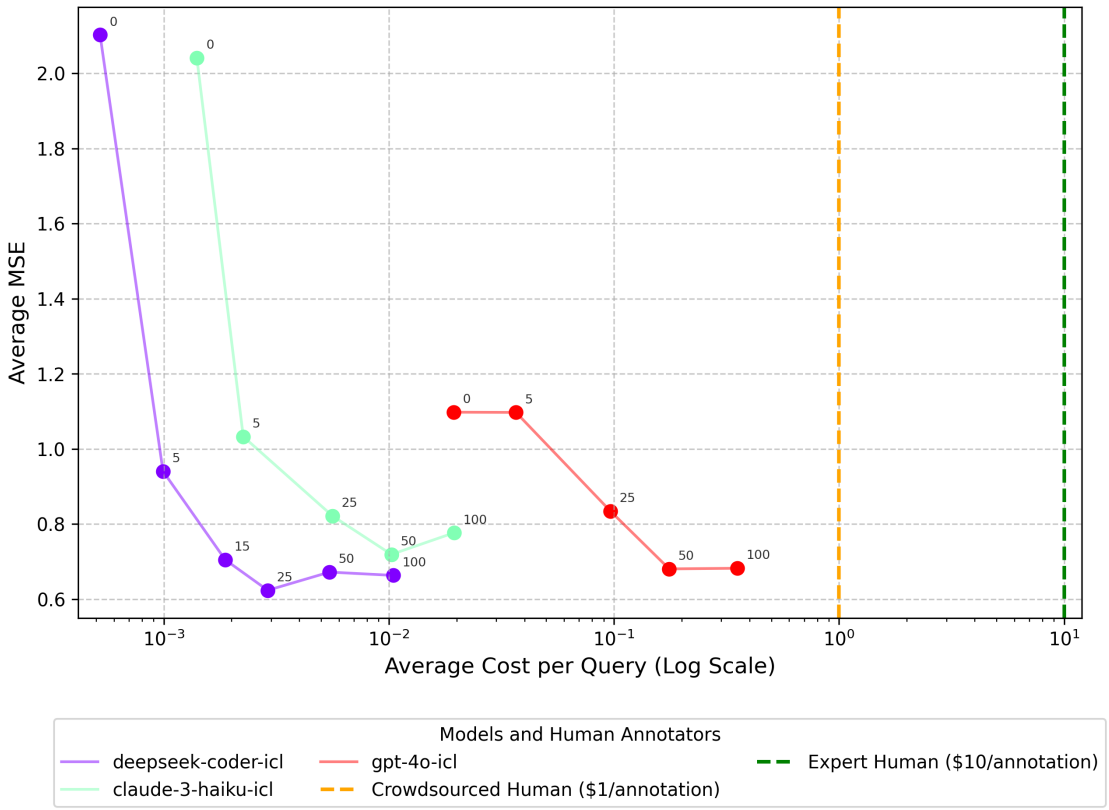


Figure 7: Model Performance (Accuracy) vs Model Cost

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.

3.4.2 Model Cost

Figure 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, 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: Model Comparison with Optimal In-Context Learning Performance for DQI Task

Figure 7 shows the relationship between average model accuracy and average cost per API query.[9] Table 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 7.

3.5 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 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.

[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.

4 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.

References

Anthropic (2023). Introducing claude. *Anthropic Blog*.

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.

Blei, D. M., Ng, A. Y., and Jordan, M. I. (2003). Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022.

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*.

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*.

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.

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.

Cui, L., Wu, Y., Liu, S., Zhang, Y., and Zhou, M. (2021). Template-based named entity recognition using bart. *arXiv preprint arXiv:2106.01760*.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). 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*.

Elman, J. L. (1990). Finding structure in time. *Cognitive Science*, 14(2):179–211.

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.

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.

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.

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*.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8):1735–1780.

Kim, Y. (2014). Convolutional neural networks for sentence classification. *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pages 1746–1751.

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*.

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*.

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.

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.

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*.

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.

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.

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.

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.

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*.

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.

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*.

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.

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*.

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.

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*.

A Appendix: Large Language Models for Text Classification

A.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.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.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. (2019). 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.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.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).

B 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.

B.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.

B.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)

B.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).

B.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 counterarguments 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.