Political DEBATE: Efficient Zero-shot and Few-shot Classifiers for Political Text

Michael Burnham[1], Kayla Kahn[2], Ryan Yang Wang[3], and Rachel X. Peng[3]

[1]Department of Politics, Princeton University
[2]Department of Political Science, The Pennsylvania State University
[3]Manship School of Mass Communication, Louisiana State University

September 2, 2024

Social scientists quickly adopted large language models due to their ability to annotate documents without supervised training, an ability known as zero-shot learning. However, due to their compute demands, cost, and often proprietary nature, these models are often at odds with replication and open science standards. This paper introduces the Political DEBATE (DeBERTa Algorithm for Textual Entailment) language models for zero-shot and few-shot classification of political documents. These models are not only as good, or better than, state-of-the art large language models at zero and few-shot classification, but are orders of magnitude more efficient and completely open source. By training the models on a simple random sample of 10-25 documents, they can outperform supervised classifiers trained on hundreds or thousands of documents and state-of-the-art generative models with complex, engineered prompts. Additionally, we release the PolNLI dataset used to train these models – a corpus of over 200,000 political documents with highly accurate labels across over 800 classification tasks.

1. Introduction

Text classification is widely used in various applications, such as opinion mining and topic classification (Minaee et al., 2021). In the past, classification was a technical and labor intensive task requiring a significant amount of manual labeling and a strong understanding of machine learning methods. Recently developed large language models (LLMs), like ChatGPT, have all but eliminated this barrier to entry due to their ability to label documents without any additional training, an ability known as zero-shot classification (Ziems et al., 2024; Gilardi et al., 2023; Burnham, 2024; Rytting et al., 2023). Because of this, it is little wonder that LLMs have received widespread adoption within political and other social sciences.

Yet, despite their convenience, there are strong reasons why researchers should be hesitant to use LLMs for text analysis. The most widely used and performative models are proprietary, closed models. Historical versions of the models are not archived for replication purposes, and the training data is not publicly released. This makes their use at odds with standards of open science. Further, these models have large compute requirements, and charge for their use – labeling datasets of any significant size can be expensive. We echo the sentiments of Palmer et al. (2024): Researchers should strive to use open sourced models and should provide compelling justification when using closed models.

We aim to narrow this gap between the advantages of closed, state-of-the-art large language models and the best practices of open science. Accordingly, we present two language models named Political DEBATE (DeBERTa Algorithm for Textual Entailment) Large and Political DEBATE Base. The models are trained specifically for zero and few-shot classification of political text. With only 86 million and 304 million parameters (He et al., 2021), the DEBATE models are not only a fraction of the size of proprietary models with tens of billions of parameters, such as Claude 3.5 Sonnet (Anthropic, 2024), but are as good or better at zero-shot classification of political documents. We further demonstrate that the DEBATE models are few-shot learners without any active learning scheme: A simple random sample of only 10–25 labeled documents is sufficient to teach the models complex labeling tasks when necessary.

We accomplish this in two ways. First, we use domain specific training with tightly controlled data quality. By focusing the model on a specific domain, the model size necessary for high performance is significantly reduced. Second, we adopt the natural language inference (NLI) classification framework. This allows us to train encoder language models (e.g. BERT (Devlin et al., 2018)) for zero-shot and few-shot classification. These models are much smaller than the generative language models like GPT-4 (OpenAI, 2023).

Additionally, we release the PolNLI dataset used to train and benchmark the models. The dataset contains over 200,000 political documents with high quality labels from a wide variety of sources across all sub-fields of political science. Finally, in the interests of open science, we commit to versioning both the models and datasets and maintaining historical versions for replication purposes. We outline the details of both the data and the NLI framework in the following sections.

2. Natural Language Inference: What and Why

Natural language inference (also known as textual entailment) can be thought of as a universal classification framework. A document of interest, known as the "premise," is paired with a user generated statement, known as the "hypothesis." The hypothesis are analogous to a very simple prompt given to a model like GPT-4 (OpenAI, 2023) or Llama-3 (AI@Meta, 2024). Given a premise and hypothesis pair, an NLI classifier is trained to determine if the hypothesis is true, given the content of the premise. For example, we might pair a tweet from Donald Trump: "It's freezing and snowing in New York – we need global warming!" with the hypothesis "Donald Trump supports global warming". The model would then give a true or false classification for the hypothesis – in this case, true. Because nearly any classification task can be broken down into this structure, a single language model trained for natural language inference can function as a universal classifier and label documents across many dimensions without additional training.

Natural language inference has a number of advantages and disadvantages in comparison to generative LLMs. Perhaps the most significant advantage is that NLI can be done with much smaller language models. While a standard BERT model with 86 million parameters can be trained for NLI, the smallest generative language models capable of accurate zero-shot classification have 7-8 billion parameters (Wei et al., 2022, e.g.), and state-of-the-art LLMs have tens to hundreds of billions of parameters (Minaee et al., 2024). In practical terms, this is the difference between a model that can feasibly run on a modern laptop, and one that requires a cluster of high-end GPUs.

The primary tradeoff between NLI classifiers and generative LLMs is between efficiency and flexibility. While an NLI classifier can be much smaller than an LLM, they are not as flexible. LLMs like GPT-4 (OpenAI, 2023) and Llama (Touvron et al., 2023) can accept long prompts that detail multiple conditions to be met for a positive classification. In contrast, the hypotheses accepted by an NLI classifier should be short and reduce the task to a relatively simple binary. Many classification tasks are not easily reduced to simple hypothesis statements.

This capability stems from the wide knowledge base about the world that LLMs hold within their weights. Because they are trained on such a massive amount of data, their training distributions contain a wider variety of tasks (e.g. classification, summarizing, programming) and domains (e.g. politics, medicine, history, pop-culture). Such a vast knowledge base requires a much larger model with higher compute demands. Often, much of the knowledge contained in these weights is superfluous to the classification task a researcher may be using them for. Thus, while LLMs have shown impressive capabilities in zero-shot settings (Ziems et al., 2024), they are inherently very inefficient tools for any *particular* classification task.

While we acknowledge that generative LLMs can play a valuable role in political research, their necessarily large size and usually proprietary nature also poses a challenge for open science standards. Their compute demands can be expensive, proprietary models are not archived for scientific replication purposes, and the lack of transparency regarding model architectures and training datasets complicates efforts to replicate or improve these models (Spirling, 2023). As a result, despite their impressive capabilities and ease of use, the use of proprietary LLMs as a classification tool at least merits explicit justification in a scientific setting (Palmer et al., 2024).

Here, we demonstrate much smaller models can often offer the convenience and performance of generative LLMs by adopting the NLI classification framework and narrowing its domain of expertise from the entire world to the political world. The advantages of our models presented here over LLMs is first, that they are smaller and thus can be more easily trained or deployed on local or free hardware. Second, they are similarly performative to state-of-the-art LLMs on tasks within their domain. Third, they can be easily versioned and archived for reproduciblity. And finally, they are truly open source in that the model architecture and all of its training data is publicly available for scrutiny or future development.

3. The PolNLI Dataset

To train our models, we compiled the PolNLI dataset – a corpus of 201,691 documents and 852 unique entailment hypothesis. We group these hypotheses into four tasks: stance detection (or opinion classification), topic classification, hate-speech and toxicity detection, and event extraction. Table 2 presents the number of datasets, unique hypotheses, and documents that were collected for each task. PolNLI a wide variety of sources including social media, news articles, congressional newsletters, legislation, crowd-sourced responses, and more. We also adapted several widely used academic datasets such as the Supreme Court Database (Spaeth et al., 2023) by attaching case summaries to the dataset's topic labels. The vast majority of text included in PolNLI is human generated — only a single dataset containing 1,363 documents is generated by an LLM.

In constructing the PolNLI dataset, we prioritized both the quality of the labels and the

Task	Datasets	Hypotheses	Documents
Stance Detection	11	361	66,581
Topic Classification	5	278	62,005
Hate-Speech/Toxicity	2	177	41,871
Event Extraction	4	36	31,234
Total	22	852	201,691

Table 1: Summary of Tasks, Datasets, Hypotheses, and Documents

diversity of the data sources. We used a five step process to accomplish this:

1. Collecting and vetting datasets.
2. Cleaning and preparing data.
3. Validating labels.
4. Hypothesis Augmentation.
5. Splitting the data.

3.1. Collecting and Vetting Datasets

We identified a total of 48 potential datasets from replication archives, the HuggingFace hub, academic projects, and government documents. A complete list of datasets we used is located in Appendix A. Several of the collected datasets had been compiled by their authors for other classification tasks while others — like the Global Terrorism Database (START, 2022) and the Supreme Court Database (Spaeth et al., 2023) — were adapted from general purpose public datasets. We also compiled several new datasets specifically for this project in order to address gaps in the training data. For each dataset, we reviewed the scope of the data, the collection and labeling process, and made a qualitative assessment of the data quality. Datasets for which we determined the quality of the data to be too low or redundant with sources already collected were omitted.

3.2. Cleaning and Preparing Data

To clean the data, we took care to remove any superfluous information from documents that the models might learn to associate with a particular label. This includes aspects like news outlet identifiers in the headings of articles or event records that start each entry with a date. No edits were made to document formatting, capitalization, or punctuation in order to maintain variety in the training data.

For each unique label in the data, we manually created a hypothesis that correlated with that label. For example, documents that were labeled for topic or event were paired with the hypothesis "This text is about (topic/event type)" and documents labeled for stance were paired with the hypothesis "The author of this document supports (stance)." Most hypotheses are framed as descriptive statements about the document, as in the two previous examples.

Finally, each document-hypothesis pair was assigned an entail/not entail label based on the label from the original dataset. For example, a document labeled as an expression of concern over global warming would be paired with the hypothesis "The author of this text believes climate change is a serious concern" and be assigned the "entail" label.[1]

One challenge with this approach is that topic and event data only contained positive entailment labels. That is, if an event summary was about a terrorist attack, it was with the hypothesis "This document is about a terrorist attack" and the entailment labels for these were initially always true. However, we wanted to train the model to not only recognize what is a terrorist attack, but what is not a terrorist attack. To accomplish this for datasets and documents that needed negative cases, we duplicated the documents and then randomly assigned one of the other topic or event hypotheses, and then assigned a "not entail" label. One concern is that documents can contain multiple topics, and might be assigned a topic they are related to by chance. This concern is addressed through the validation process outlined in the next section.

3.3. Validating Labels

The original curators of the collected datasets used many approaches to labeling their data with varying levels of rigor. The accuracy of labels is critically important to training and validating models, and thus we wanted to ensure that only high quality labels were retained in our data. To meet this objective, we leveraged the much larger language models, GPT-4 and GPT-4o. Recent research has shown that LLMs are as good, or better, than human coders for similar classification tasks (Burnham, 2024; Chang et al., 2024; Gilardi et al., 2023). We thus used these proprietary LLMs to reclassify each collected document with a prompt containing

[1] While several other NLI datasets, such as SNLI, have adopted an entail, neutral, contradict labeling scheme, we opted for the simpler entail/not entail because it was a common scheme that all of the collected datasets could be adapted to. Accordingly, neutral and contradiction labels were combined into the "not entail" label.

an explanation of the task and the entailment hypotheses we generated. A template for the prompt is contained in Appendix B. We then removed documents where the human labelers and the LLM disagreed. To ensure that the LLMs were generating high quality labels, we took a random sample of 400 documents labeled by GPT-4o and manually reviewed the labels again. We agreed with the GPT-4o labels 92.5% of the time, with a Cohen's κ of 0.85. Of the 30 documents where there was disagreement, 16 were judged to be reasonable disagreements where the document could be interpreted either way. The remaining 14, or 3.5% of all documents, were labeled incorrectly by the LLM.

3.4. Hypothesis Augmentation

An ideal NLI classifier will produce identical labels if a document is paired with different, but synonymous, hypotheses (e.g. the hypotheses "This document is about Trump" and "This text discusses Trump" should yield similar classifications). To make our model more robust to the various phrasings researchers might use for hypotheses, we presented each hypothesis to GPT-4o and then asked it to write three synonymous sentences. We then manually reviewed the LLM generated hypotheses and removed any that we felt were not sufficiently similar in meaning. Each document was then randomly assigned an "augmented hypothesis" from a set containing the original hypothesis and the generated alternatives. Finally, we manually varied hypotheses by randomly substituting a few very common words with synonymous words (e.g. text/document, supports/endorses). In total, this increased the number of unique entailment phrases to 2,834.

3.5. Splitting the Data

To split the data into training, validation, and test sets we proportionally sampled from each of the four tasks to construct testing and validation sets of roughly 15,000 documents each. The rest of the data were allocated to the training set. Because we wanted to evaluate model performance in a zero-shot context, a simple random sampling approach to splitting the data would not work. Instead, we randomly sampled from the set of unique hypotheses and allocated all documents with those hypotheses to the test set. This ensures that models did not see any of the test set hypotheses, or their synonymous AI generated variants, during training. The validation set consists of roughly 10,000 documents with hypotheses that are not in the training set, and 5,000 documents with hypotheses that are in the training set. This allows us to both estimate the model's zero-shot performance during testing, as well as look for evidence of overfitting if performance diverges between the hypotheses seen and not seen during training.

4. Training

The foundation models we used for training were a pair of DeBERTa V3 base and large models fine tuned for general purpose NLI classification by Laurer et al. (2023). We use these models for a number of reasons: First, the DeBERTa V3 architecture is the most performative on NLI tasks among transformer language models of this size (Wang et al., 2019). Second, using models already trained for general purpose NLI classification allows us to more efficiently leverage transfer learning. Before we used these models for our application, they were trained on five large datasets for NLI, and 28 smaller text classification datasets. This means that we begin training with a model that already understands the NLI framework and general classification tasks, allowing it to more quickly adapt to the specific task of classifying political texts (Laurer et al., 2022).

We used the Transformers library (Wolf et al., 2020) to train the model and monitored training progress with the Weights and Biases library (Biewald, 2020). After each training epoch (an entire pass through of the training data), model performance was evaluated on the validation set and a checkpoint of the model was saved. We selected the best model from these checkpoints using both quantitative and qualitative approaches. The model's training loss, validation loss, Matthew's Correlation Coefficient (MCC), F1, and accuracy was reported for each checkpoint. We then tested the best performing models according to these metrics by examining performance on the validation set for each of the four classification tasks, and across each of the datasets. This helped us to identify models with consistent performance across task and document type.

Finally, we qualitatively assessed the models by examining their behavior on individual documents. This included introducing minor edits or re-phrasings of the documents or hypotheses so that we could identify models with stable performance that were less sensitive to arbitrary changes to features like punctuation, capitalization, or synonymous word choice. Hyperparameters used to train the models are in Appendix C.

5. Zero-shot Learning Performance

We benchmark our models on the PolNLI test set against four other models that represent a range of options for zero-shot classification. The first two models are the DeBERTa base and DeBERTa large general purpose NLI classifiers trained by Laurer et al. (2023). These are currently the best NLI classifiers that are publicly available (Laurer et al., 2023). We also test the performance of Llama 3.1 8B, an open source generative LLM released by Meta (AI@Meta, 2024). This model is the smallest version of Llama 3.1 released and represents a generative LLM that can feasibly be run on a desktop computer with a high end GPU, or a CPU with an integrated GPU like the Apple M series chips in modern macbooks. Finally, we benchmark Calude 3.5 Sonnet (Anthropic, 2024). This model is a state-of-the-art proprietary LLM. At the time of writing, it is widely considered to be among the best models available (Syed et al., 2024). Notably, we do not include GPT-4o in our benchmark because it was used in the validation process from which the final labels were derived. We discourage bench-marking OpenAI models on the PolNLI dataset for this reason.

We use MCC as our primary performance metric due to its relative robustness to other metrics like F1 and accuracy on binary classification tasks (Chicco and Jurman, 2020, 2023). MCC is a special case of the Pearson correlation coefficient and can be interpreted similarly. It rangers from -1 to 1 with higher values indicating greater performance.

5.1. PolNLI Test Set

Figure 1 plots performance with bootstrapped standard errors across all four tasks for each model. We observe that the DEBATE models are more performative than alternatives when all tasks and datasets are combined.

In figure 2 we break out performance across our four tasks: Topic classification, stance detection, event extraction, and hate-speech identification. While all models perform well on topic classification, significant gaps emerge on the other tasks. The DEBATE models and Claude 3.5 Sonnet perform significantly better than the other models on stance detection. On event extraction tasks we see comparable performance between the DEBATE models and the two generative LLMs, Claude 3.5 and Llama 3.1. Perhaps the most notable gap in performance is on the hate-speech detection task – the DEBATE models perform significantly better than the other models. We think that this is likely because hate-speech is a highly subjective concept

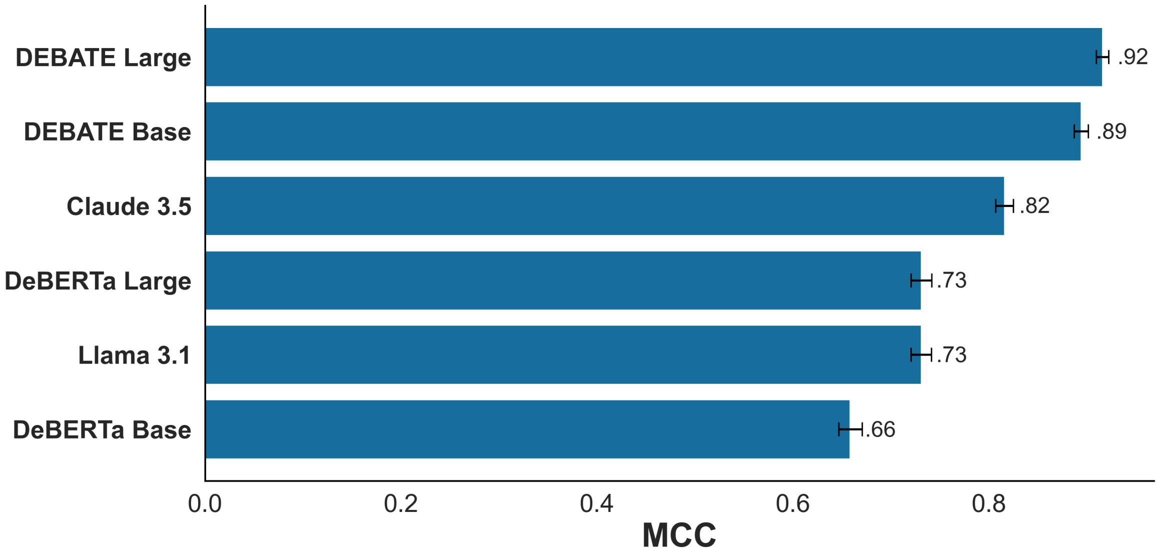


Figure 1: Zero-shot performance of all four tasks for each model

and our models are better tuned to the particular definitions used in the datasets we collected.

Finally, in figure 3, we plot this distribution of performance across all datasets in the test set. We again observe that the DEBATE models are more consistently performative than alternatives. For most models, the Polistance Quote Tweets dataset was the most challenging dataset, with the DeBERTa Large model having a negative correlation with the correct classification. This dataset measures stance detection and is particularly challenging for language models to parse because quote tweets often contain two opinions from two different people. The model has to parse both of these opinions and correctly attribute stances to the right authors. Even the state-of-the-art Claude 3.5 had an MCC of only 0.29 on the task. However, because the Political DEBATE models were explicitly trained to parse such documents, the base and large models were able to achieve MCCs of 0.62 and 0.88 respectively.

6. Few-shot Learning Performance

One advantage of the NLI classification framework is that models trained for NLI can more quickly adapt to other classification tasks (Laurer et al., 2022). Few-shot learning refers to the ability to learn a new classification task with only a few examples. Whereas a conventional supervised classifier usually requires hundreds or even thousands of labeled documents to train, models like GPT-4 and Claude 3.5 have demonstrated the ability to improve classification with

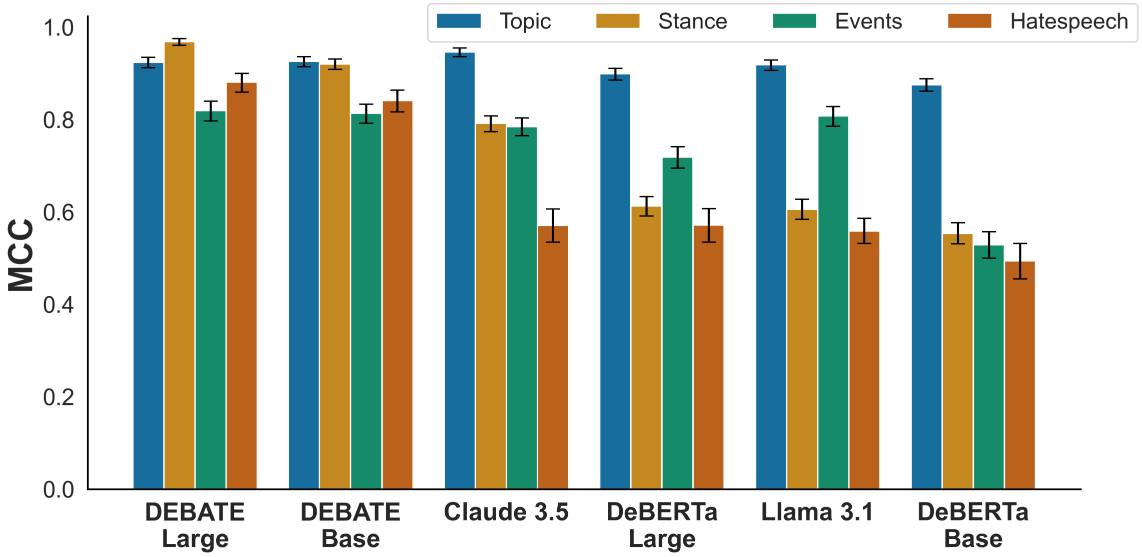


Figure 2: Zero-shot performance of each task for each model

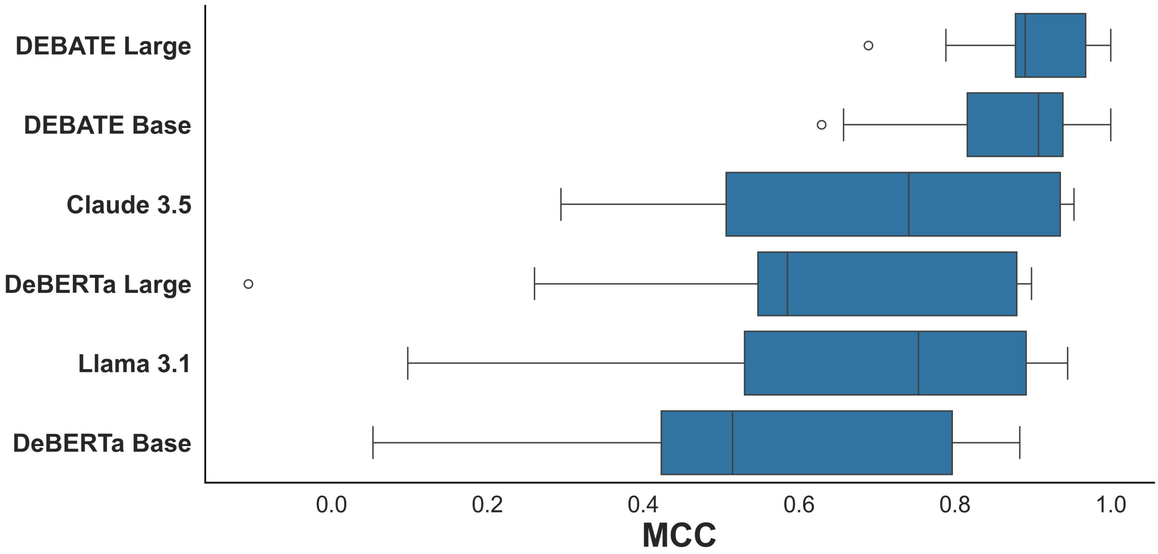


Figure 3: Zero-shot learning MCC distribution of four tasks for each model

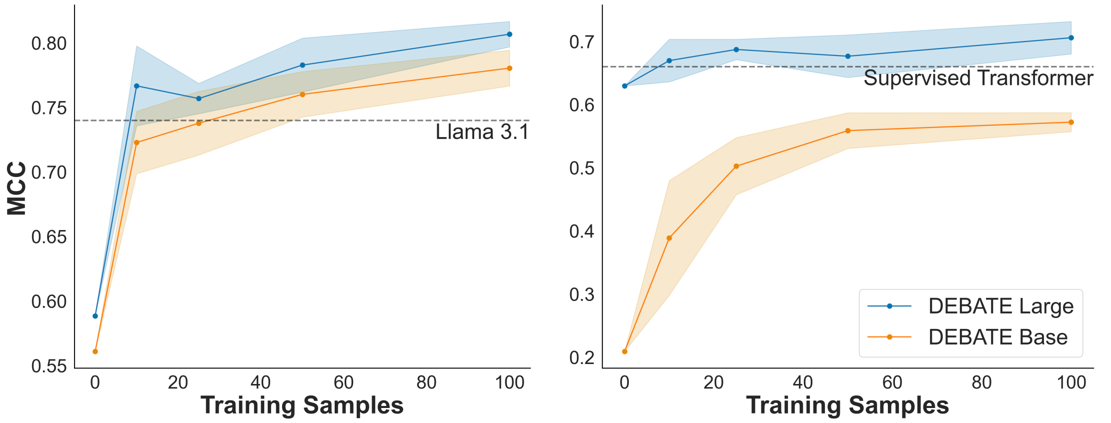
only a handful of examples provided in the prompt.

Here, we demonstrate that domain adapted NLI classifiers are efficient few-shot learners. With a random sample of 10–25 documents and no active learning scheme, these models can learn new classification tasks at levels comparable to, or better than, supervised classifiers and generative language models. We use two examples from other research projects to illustrate this capability. The first comes from the Mood of the Nation poll, a regular poll issued by the McCourtney Institute for Democracy which recently begun using Llama 3.1 as part of its annotation process for open-text survey questions (Berkman and Plutzer, 2024). The second is from Block Jr. et al. (2022) who trained a transformer model on roughly 2,000 tweets to identify posts that minimize the threat of COVID-19.

For our testing procedure we first use both DEBATE models and a simple hypothesis for zero-shot classification on each document. We then take four simple random samples of 10, 25, 50, and 100 documents, train each of the two DEBATE models on these random samples, and then estimate performance of both models for the respective sample size on the rest of the documents. We repeat this 10 times for each training sample size and calculate a 95% confidence interval. Importantly, we did not search for the best performing hypothesis statements or model hyper-parameters. We simply used the default learning rate and then trained the model for 5 epochs. We felt this was important because a few-shot application assumes researchers do not have a large sample of labeled data to search for the best performing parameters. Rather, few-shot learning should work out-of-the-box to be useful. We also note that that while training these language models on large data sets like PolNLI can take hours or days with a high end GPU, training time in a few shot context is reduced down to seconds or minutes and can be done without high-end computing hardware.

6.1. Mood of the Nation: Liberty and Rights

One of the questions on the Mood of the Nation poll is an open text form asking, “What does democracy mean to you?” The administrators of this poll had a team of research assistants manually label answers to this question that were related to “liberty and rights.” This category was broadly defined as responses that discuss freedoms and rights generally, or specific rights such as speech, religion, or the contents of the bill of rights. However, if the document was exclusively about voting rights, it was assigned to another category. If it mentioned voting rights in addition to other rights, it was still classified as “liberty and rights.” This classification



(a) Performance on Mood of the Nation dataset (compared with Llama 3.1)

(b) Performance on COVID-19 dataset (compared with Electra transformer)

Figure 4: Few-shot learning performance of DEBATE models

task is somewhat difficult to reduce down to a simple hypothesis statement, and is thus a good candidate for few-shot training.

Berkman and Plutzer (2024) wanted to automate the labeling of short answer responses and wanted to use open source models to do so. This was motivated both by open science standards, and privacy concerns over uploading responses to a proprietary API like GPT-4o. Llama 3.1 worked well, and classified documents with an MCC of 0.74 and accuracy of 88%. Discrepancies between the LLM and human coders were judged to be primarily reasonable disagreements.

In a zero-shot context, Llama 3.1 comfortably out performs our models due to its ability to accept prompts with more detailed instructions. However, after only 10 training samples we see a large jump in performance with both the large and base DEBATE models, with Llama not significantly different than either. At 25 documents the large model significantly outperforms llama 3.1, and at 50 documents the base model does as well.

6.2. COVID-19 Threat Minimization

Block Jr. et al. (2022) classified Twitter posts about COVID-19 based on whether or not they minimized the threat of COVID-19. Threat minimization was defined as anti-vaccination or anti-masking rhetoric, comparisons to the flu, statements against stay-at-home orders, claims that COVID-19 death counts were faked, or general rhetoric that the disease did not pose a significant health threat. This presents a particularly difficult classification challenge because threat minimization of COVID-19 is a somewhat abstract concept and can be expressed in many different ways across disparate topics. To address this, Block Jr. et al. (2022) trained an Electra transformer on 2,000 tweets with a Bayesian sweep of the hyper-parameter space. This process involved training 30 iterations of the model to find the best performing hyper-parameters. The final model achieved an MCC of 0.66.

For an NLI classifier, the above classification criteria are too numerous to elegantly fit into a single entailment hypothesis. While Burnham (2024) demonstrated that such tasks can be done zero-shot by dividing it into smaller tasks (e.g. classify the documents once for anti-vaccination rhetoric, another time flu comparisons, and so forth), few-shot learning provides a more elegant solution. To test the models, we use the basic hypothesis "The author of this tweet does not believe COVID is dangerous." Here, we observe that the base model largely fails at the task in a zero-shot context, and fails to match the accuracy of the supervised classifier with 100 training samples. The large model proves more capable of learning the task, matching the supervised classifier at 10 training samples (MCC = 0.67, accuracy = 87%), and exceeding it at only 25 training samples (MCC = 0.69, accuracy = 88%).

7. Timing Benchmarks

To assess cost-effectiveness, we ran our two DEBATE models and Llama 3.1 across a diverse range of hardware.[2] We did so with a random sample of 5,000 documents from the PolNLI test set and the simple hypothesis "This text is about politics." We selected four different types of hardware. First, the NVIDIA GeForce RTX 3090 GPU provides high-performance, consumer-grade machine learning capabilities, making it a suitable choice for intensive computational tasks. Second, the NVIDIA Tesla T4 is a free GPU available through Google Colab. In contrast to the RTX 3090, the T4 is easy for researchers to access free of charge. Third, we used a Macbook Pro with the M3 max chip. This is a common laptop with a built-in GPU that is integrated in the system-on-chip, as opposed to the RTX 3090 and Tesla T4 which are discrete GPUs. Finally, the AMD Ryzen 9 5900x CPU was utilized to evaluate performance on a general purpose CPU.[3]

We observe that the DEBATE models offer massive speed advantages over even small gen-

[2] We exclude the DeBERTa models used in the performance bench-marking above because they have the same architecture as the DEBATE models, and thus label documents at the same speed. We also do not time proprietary LLMs because their speed is highly determined by server traffic and we cannot test them in a controlled setting on common hardware. Classification speed is also of more concern when using local hardware because it occupies computing resources that might be need during the run time.

[3] We do not test Llama 3.1 on the Tesla T4 GPU or the Ryzen 5900x CPU. The model is too large to run on the Tesla T4, and slow enough on on a CPU that it's not recommended to do so in any context.

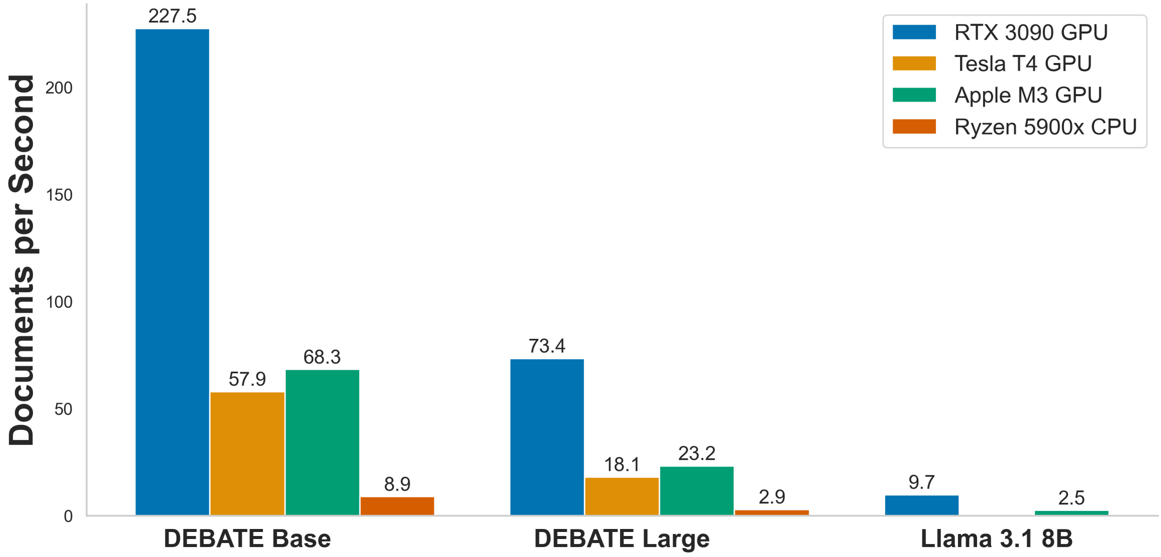


Figure 5: The DEBATE models offer a massive efficiency advantage over generative language models.

erative LLMs like Llama 3.1 8B. While discrete GPUs like the RTX 3090 do offer a large performance advantage, Documents can still be classified at a relatively brisk pace with a laptop GPU like on the M3, or a free cloud GPU like the T4.

8. Limitations and Model Use

The model and dataset can be downloaded for free on the HuggingFace hub. We recommend using Python's Transformers (Wolf et al., 2020) and Datasets (Lhoest et al., 2021) libraries to use the models and data. In most cases, models and data can be deployed with only a few lines of code. We include boilerplate code for both zero-shot and few-shot applications on the github repository for this paper. While we offer brief advice on application here, for a more thorough exploration of best practices when using NLI classifiers we defer to Burnham (2024).

8.1. Which Model Should I Use?

We offer the following two guidelines for selecting a model:

- Use the large model for zero-shot classification.
- Use the large model for most few-shot applications.
- Use the base model for simple few-shot tasks or supervised classification.

Both our extensive use of NLI models and previous research (Burnham, 2024) indicates that larger models are much better at generalizing to unseen tasks. However, for tasks that are more explicitly within the training distribution such as hate-speech detection or approval of politicians, we expect comparable performance between the large and base models, with the base model offering a significant advantage in efficiency. In the few-shot context we expect similar performance between the large and base model given the results above. However, We also observed that the large model more quickly learns tasks. There is also no clear measurement of a task's simplicity, only qualitative judgements. Thus, we recommend using the large model whenever feasible.

8.2. When Should I Consider Few-shot or Supervised Training?

An NLI classifier will be most performative in a zero-shot context under the following conditions:

- Labels can reasonably be derived from only the text of a document and do not require meta-knowledge about the document such as who wrote it, when, and under what circumstances.
- Labels are for concepts that are commonly understood (e.g. support/opposition to a person or policy) rather than bespoke concepts for a particular research project, or require specialized domain knowledge (e.g. documents about political rights except the right to vote, "threat minimization" of COVID-19).
- Documents are short, generally a sentence or paragraph, or can be segmented into short documents.

If any of these conditions are not met, you should consider few-shot or supervised training. Whether or not these conditions are met is a qualitative judgment that should be made based on familiarity with the data and task. As with any classification task, you should always validate your results with some manually labeled data.

8.3. How Should I Construct Hypotheses?

We recommend using short, simple hypotheses similar to the templates used in the training data. Fore example:

- "This text is about (topic or event)"

- “The author of this text supports (politician or policy position)”
- “This text is attacking (person or group)”
- “This document is hate-speech”

While researchers can certainly deviate from these templates, few-shot training may be appropriate for tasks that require long hypotheses with multiple conditions.

8.4. Other Limitations

Despite the impressive results demonstrated here, we want to emphasize that researchers should not expect the DEBATE models to outperform proprietary LLMs on all classification tasks. The massive size and training sets of proprietary models inevitably means a larger variety of tasks are in their training distribution. Accordingly, we expect that LLMs will more robustly generalize in the zero-shot context for tasks that are less proximate to what is contained in the PolNLI data set. We recommend few-shot training for such tasks.

We also note that these models are trained exclusively for English documents, and it is unknown how the models would perform if re-trained for non-English documents.

9. Conclusion and Future Work

The presented zero and few-shot entailment models, currently effective in stance, topic, hate-speech, and event classification, shows immense potential for open, accessible, and reproducible text analysis in political science. Future research should explore expanding the capabilities of these models to new tasks (such as identifying entities, and relationships) and new document sources. While we think that these models can be immensely valuable to researchers now, we hope that this is only the first step in developing efficient, open source models tailored for specific domains. We think that there is significant room to further expand the PolNLI data set and, as a result, train better models that more widely generalize across political communication. We believe that domain adapted language models can be a public good for the research community and hope that researchers studying politics will collaborate to share data and expand the training corpus for these models.

Further, we also believe that open source LLM-based chat bots could benefit greatly from our approach of domain adaptation and entailment classification. Thus, in future work we hope to adapt and expand the PolNLI data set to make it suitable for generative language models. By doing so, it is plausible that not only could generative models smaller than Llama 8B achieve state-of-the-art classification performance, but researchers would be able to use these models for tasks encoder models are not capable of, such as synthetic data generation or summarization.

References

AI@Meta (2024). Llama 3 model card.

Anthropic (2024). Claude 3.5 sonnet.

Anti-Defamation League (2023). HEAT Map: Adl h.e.a.t. map.

Ballard, A. O., R. DeTamble, S. Dorsey, M. Heseltine, and M. Johnson (2023). Dynamics of polarizing rhetoric in congressional tweets. *Legislative Studies Quarterly 48*(1), 105–144.

Bar-Haim, R., I. Bhattacharya, F. Dinuzzo, A. Saha, and N. Slonim (2017, April). Stance classification of context-dependent claims. In M. Lapata, P. Blunsom, and A. Koller (Eds.), *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, Valencia, Spain, pp. 251–261. Association for Computational Linguistics.

Berkman, M. B. and E. Plutzer (2024). Generational differences in the meaning of 'democracy' and its conserquences for diffuse support in the united states. *Working paper*.

Biewald, L. (2020). Experiment tracking with weights and biases. Software available from wandb.com.

Bird, C., M. Whyman, B. D. Jones, F. R. Baumgartner, S. M. Theriault, D. A. Epp, C. Lee, and M. E. Sullivan (2009). Policy agendas project: Supreme court cases. `https://www.comparativeagendas.net/`. Accessed: [Insert access date here].

Block Jr., R., M. Burnham, K. Kahn, R. Peng, J. Seeman, and C. Seto (2022). Perceived risk, political polarization, and the willingness to follow covid-19 mitigation guidelines. *Social Science & Medicine*, 115091.

Burnham, M. (2024). Stance detection: A practical guide to classifying political beliefs in text. *Political Science Research and Methods*.

Chang, Y., X. Wang, J. Wang, Y. Wu, L. Yang, K. Zhu, H. Chen, X. Yi, C. Wang, Y. Wang, et al. (2024). A survey on evaluation of large language models. *ACM Transactions on Intelligent Systems and Technology 15*(3), 1–45.

Chicco, D. and G. Jurman (2020). The advantages of the matthews correlation coefficient (mcc) over f1 score and accuracy in binary classification evaluation. *BMC genomics 21*(1), 1–13.

Chicco, D. and G. Jurman (2023). The matthews correlation coefficient (mcc) should replace the roc auc as the standard metric for assessing binary classification. *BioData Mining 16*(1), 4.

Devlin, J., M.-W. Chang, K. Lee, and K. Toutanova (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Durmus, E., L. Lovitt, A. Tamkin, S. Ritchie, J. Clark, and D. Ganguli (2024). Measuring the persuasiveness of language models.

Gilardi, F., M. Alizadeh, and M. Kubli (2023). Chatgpt outperforms crowd-workers for text-annotation tasks. *arXiv preprint arXiv:2303.15056*.

Gretz, S., R. Friedman, E. Cohen-Karlik, A. Toledo, D. Lahav, R. Aharonov, and N. Slonim (2019). A large-scale dataset for argument quality ranking: Construction and analysis. *CoRR abs/1911.11408*.

He, P., J. Gao, and W. Chen (2021). Debertav3: Improving deberta using electra-style pre-training with gradient-disentangled embedding sharing.

Jones, B. D., F. R. Baumgartner, S. M. Theriault, D. A. Epp, R. Eissler, C. Lee, and M. E. Sullivan (2023). Policy agendas project: State of the union speeches. `https://www.comparativeagendas.net/`. Accessed: [Insert access date here].

Kawintiranon, K. and L. Singh (2022). Polibertweet: A pre-trained language model for analyzing political content on twitter. In *Proceedings of the Language Resources and Evaluation Conference*. European Language Resources Association.

Kennedy, C. J., G. Bacon, A. Sahn, and C. von Vacano (2020). Constructing interval variables via faceted rasch measurement and multitask deep learning: a hate speech application. *arXiv preprint arXiv:2009.10277*.

Laurer, M., W. v. Atteveldt, A. S. Casas, and K. Welbers (2022). Less annotating, more classifying – addressing the data scarcity issue of supervised machine learning with deep transfer learning and bert - nli. *Open Science Framework Preprint*.

Laurer, M., W. van Atteveldt, A. Casas, and K. Welbers (2023, December). Building Efficient Universal Classifiers with Natural Language Inference. arXiv:2312.17543 [cs].

Lhoest, Q., A. V. del Moral, P. von Platen, T. Wolf, M. Sasko, Y. Jernite, A. Thakur, L. Tunstall, S. Patil, M. Drame, et al. (2021). Datasets: A community library for natural language processing. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pp. 175–184.

Luo, Y., D. Card, and D. Jurafsky (2020, November). Detecting stance in media on global warming. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, Online, pp. 3296–3315. Association for Computational Linguistics.

Minaee, S., N. Kalchbrenner, E. Cambria, N. Nikzad, M. Chenaghlu, and J. Gao (2021). Deep learning–based text classification: a comprehensive review. *ACM computing surveys (CSUR) 54*(3), 1–40.

Minaee, S., T. Mikolov, N. Nikzad, M. Chenaghlu, R. Socher, X. Amatriain, and J. Gao (2024). Large language models: A survey. *arXiv preprint arXiv:2402.06196*.

OpenAI (2023). Gpt-4 technical report.

Palmer, A., N. A. Smith, and A. Spirling (2024). Using proprietary language models in academic research requires explicit justification. *Nature Computational Science 4*(1), 2–3.

Raleigh, C., R. Kishi, and A. Linke (2023). Political instability patterns are obscured by conflict dataset scope conditions, sources, and coding choices. *Humanities and Social Sciences Communications 10*, 74.

Rytting, C. M., T. Sorensen, L. Argyle, E. Busby, N. Fulda, J. Gubler, and D. Wingate (2023). Towards coding social science datasets with language models. *arXiv preprint arXiv:2306.02177*.

Salehyan, I., C. S. Hendrix, J. Hamner, C. Case, C. Linebarger, E. Stull, and J. Williams (2012). Social conflict in africa: A new database. *International Interactions 38*(4), 503–511.

Sobhani, P., D. Inkpen, and X. Zhu (2017). A dataset for multi-target stance detection. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pp. 551–557.

Spaeth, H. K., L. Epstein, A. D. Martin, J. A. Segal, T. J. Ruger, and S. C. Benesh (2023). Supreme court database. *Version 2023 Release 01*.

Spirling, A. (2023). Why open-source generative ai models are an ethical way forward for science. *Nature 616*(7957), 413–413.

START (2022). Global terrorism database. Accessed on June 20, 2024.

Syed, U., E. Light, X. Guo, L. Q. Huan Zhanga, Y. Ouyang, and B. Hu (2024). Benchmarking the capabilities of large language models in transportation system engineering: Accuracy, consistency, and reasoning behaviors. *arXiv:2408.08302*.

Touvron, H., T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar, et al. (2023). Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Wang, A., Y. Pruksachatkun, N. Nangia, A. Singh, J. Michael, F. Hill, O. Levy, and S. Bowman (2019). Superglue: A stickier benchmark for general-purpose language understanding systems. *Advances in neural information processing systems 32*.

Wei, J., M. Bosma, V. Y. Zhao, K. Guu, A. W. Yu, B. Lester, N. Du, A. M. Dai, and Q. V. Le (2022). Finetuned language models are zero-shot learners. *arXiv arXiv:2109.01652*.

Wolbrecht, C., B. Shannon, E. Fagan, B. D. Jones, F. R. Baumgartner, S. M. Theriault, D. A. Epp, C. Lee, and M. E. Sullivan (2023a). Policy agendas project: Democratic party platform. `https://www.comparativeagendas.net/`.

Wolbrecht, C., B. Shannon, E. Fagan, B. D. Jones, F. R. Baumgartner, S. M. Theriault, D. A. Epp, C. Lee, and M. E. Sullivan (2023b). Policy agendas project: Republican party platform. `https://www.comparativeagendas.net/`.

Wolf, T., L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, et al. (2020). Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 conference on empirical methods in natural language processing: system demonstrations*, pp. 38–45.

Ziems, C., W. Held, O. Shaikh, J. Chen, Z. Zhang, and D. Yang (2024). Can large language models transform computational social science? *Computational Linguistics 50*(1), 237–291.

A. Data Sources

Table 2: Data Sets Overview

Data Set	**Source**	**Task**	**Notes**
Multi-target Stance Detection	Sobhani et al. (2017)	Stance	Stance labeled tweets, each containing multiple politicians.
PoliBERTweet Training	Kawintiranon and Singh (2022)	Stance	Tweets about Trump and Biden.
Polistance Affect	New Dataset	Stance	Tweets labeled for stance towards 20+ members of congress.
Polistance Quote Tweets	New Dataset	Stance	Quote tweets labeled for stance towards 20+ members of congress.
Newsletter Sentences	New Dataset	Stance	Newsletter sentences collected from DC Inbox. Labeled for stance towards 20+ members of congress
Political Tweets	Huggingface Hub	Stance	Tweets from senators and representatives labeled for stance on political issues.
ADL Heat Map Dataset	Anti-Defamation League (2023)	Events	Description of antisemitic incidents with category and type labels.
State of the Union Speeches	Jones et al. (2023)	Topic	Sentences from State of the Union speeches coded by topic and subtopic.
Democratic Party Platforms	Wolbrecht et al. (2023a)	Topic	Sentences from Democratic party platforms coded by topic and subtopic.
Republican Party Platforms	Wolbrecht et al. (2023b)	Topic	Sentences from Republican party platforms coded by topic and subtopic.
The Supreme Court Database	(Spaeth et al., 2023) and Bird et al. (2009)	Topic	Summaries of court cases labeled by legal topic. Summaries were taken from the Comparative Agendas Project.
Argument Quality Ranking	Gretz et al. (2019)	Stance	Crowd sourced arguments for or against 71 different propositions. Subset to include only political topics.
Global Warming Media Stance	Luo et al. (2020)	Stance	News leads labeled for if they portray global warming as a threat.
Claim Stance	Bar-Haim et al. (2017)	Stance	Claims from Wikipedia across 55 topics.
Claim Stance	Bar-Haim et al. (2017)	Topic	Claims from Wikipedia across 55 topics.
ACLED	Raleigh et al. (2023)	Events	Descriptions and headlines of violent events and political demonstrations.
SCAD	Salehyan et al. (2012)	Events	Summaries of conflict events in Africa and Latin America labeled by event type.
Measuring Hate Speech	Kennedy et al. (2020)	Hate	Hate speech and counter hate speech. Crowd sourced labels.
Anthropic Persuasion	Durmus et al. (2024)	Stance	Arguments generated by Claude 2 and 3 across 75 topics. Subset to political topics.
Polarizing Rhetoric Tweets	Ballard et al. (2023)	Hate	Tweets labeled by whether or not they use polarizing rhetoric.
Bill Summaries	Huggingface Hub	Topic	Bill summaries and labels from congress.gov.
Political or Not	New Dataset	Topic	News articles combined with samples from the other data sets.

B. LLM Prompts

B.1. GPT-4/4o Label Validation Prompts and Arguments

"You are a classifier that can only respond with 0 or 1. I'm going to show you a short text sample and I want you to determine if {hypothesis}. Here is the text:
{document}
If it is true that {hypothesis}, return 0. If it is not true that {hypothesis}, return 1. Do not explain your answer, and only return 0 or 1."

B.2. GPT-4o Hypothesis Augmentation Prompt

"Write 3 sentences that are synonymous to this sentence:
{hypothesis}
Format your output as a python list named 'hypoths."'

B.3. GPT-4/4o Model Arguments

model = "gpt-4-1106-preview" (for GPT-4 queries)

model = "gpt-4o-2024-05-13" (for GPT-4o queries)

system_message = "You are a text classifier and are only allowed to respond with 0 or 1"

max_tokesn = 1

temperature = 0

logit_bias = {15:100, 16:100}

C. Training Parameters

C.1. Base Model

lr_scheduler_type= “linear”

group_by_length=False

learning_rate=2e-5

per_device_train_batch_size=8

per_device_eval_batch_size=8

num_train_epochs=20

warmup_ratio=0.06

weight_decay=0.01

fp16=True

fp16_full_eval=True

eval_strategy=“epoch”

seed=1

save_strategy=“epoch”

dataloader_num_workers = 12

C.2. Large Model

lr_scheduler_type= “linear”

group_by_length=False

learning_rate=9e-6

per_device_train_batch_size=4

per_device_eval_batch_size=8

gradient_accumulation_steps=4

num_train_epochs=20

warmup_ratio=0.06

weight_decay=0.01

fp16=True

fp16_full_eval=True

eval_strategy=“epoch”

seed=1

save_strategy="epoch"

dataloader_num_workers = 12