Political Analysis (2025), **0**, 1–20
doi:10.1017/pan.2025.10023

PA

ARTICLE

Stay Tuned: Improving Sentiment Analysis and Stance Detection Using Large Language Models

Max Griswold, Michael W. Robbins and Michael S. Pollard

Economics, Sociology, and Statistics Department, RAND Corporation, USA

Corresponding author: Max Griswold; Email: griswold@rand.org

(Received 10 September 2024; revised 26 March 2025; accepted 14 April 2025)

Abstract

Sentiment analysis and stance detection are key tasks in text analysis, with applications ranging from understanding political opinions to tracking policy positions. Recent advances in large language models (LLMs) offer significant potential to enhance sentiment analysis techniques and to evolve them into the more nuanced task of detecting stances expressed toward specific subjects. In this study, we evaluate lexicon-based models, supervised models, and LLMs for stance detection using two corpuses of social media data—a large corpus of tweets posted by members of the U.S. Congress on Twitter and a smaller sample of tweets from general users—which both focus on opinions concerning presidential candidates during the 2020 election. We consider several fine-tuning strategies to improve performance—including cross-target tuning using an assumption of congressmembers' stance based on party affiliation—and strategies for fine-tuning LLMs, including few shot and chain-of-thought prompting. Our findings demonstrate that: 1) LLMs can distinguish stance on a specific target even when multiple subjects are mentioned, 2) tuning leads to notable improvements over pretrained models, 3) cross-target tuning can provide a viable alternative to in-target tuning in some settings, and 4) complex prompting strategies lead to improvements over pretrained models but underperform tuning approaches.

Keywords: computational social science; machine learning; sentiment analysis; stance detection; large language models

Edited by: Daniel J. Hopkins and Brandon M. Stewart

1. Introduction

Stance detection is a natural language processing technique that analyzes a segment of text to estimate the emotive tone expressed in regards to a specific topic, building on sentiment analysis methods (AlDayel and Magdy 2021; Bestvater and Monroe 2023; Mohammad, Sobhani, and Kiritchenko 2017) which simply gauge the overall sentiment of the text. Sentiment analysis methods have been used for a diverse range of tasks within social science and political science research. These techniques have been used to measure media tone (Boukes *et al.* 2020), public opinions (Ceron *et al.* 2014), political polarization (Yarchi, Baden, and Kligler-Vilenchik 2021), politician favorability (Nasukawa and Yi 2003), medical patient experiences (Greaves *et al.* 2013), disaster relief (Beigi *et al.* 2016) and mis- and disinformation detection (Alonso *et al.* 2021; Hardalov *et al.* 2022). More recent work has discussed how stance expressed toward specific subjects within texts is often distinct from general sentiment (Bestvater and Monroe 2023), leading to the development of new approaches to improve methods for stance detection (Burnham 2024; Zhang *et al.* 2024). Stance detection has subsequently been used for a variety of purposes, including annotation of political messages (Törnberg 2023), assessing views on

COVID-19 health mandates (Alizadeh *et al.* 2025), and tracking news media stance on political topics (Mets *et al.* 2024).

A variety of methods have been used to estimate stance, broadly falling into three categories: (1) lexicon-based approaches (LBAs), which assign sentiment scores to texts by aggregating predefined word-level values; (2) supervised language models (SLMs), which are models pretrained on large corpora and fine-tuned for specific classification tasks, including stance detection and natural language inference; and (3) generative large language models (LLMs), which generate text in response to a provided task description (Alturayeif, Luqman, and Ahmed 2023; Burnham 2024; Zhang *et al.* 2024).

Previous work has found that fine-tuning SLMs and LLMs can improve stance detection. (Alizadeh *et al.* 2024; Bestvater and Monroe 2023; Laurer *et al.* 2024b). There are multiple approaches to finetuning, including: in-target tuning, which uses annotated text specific to the stance detection task to refine model performance (Zhang *et al.* 2024); cross-target tuning, which leverages annotated data from related tasks or domains to improve performance on a target task (Osnabrügge, Ash, and Morelli 2023; Xu *et al.* 2018); and for LLMs, prompt engineering techniques such as few-shot prompting, where a small number of labeled examples are included in the task description, and chain-of-thought prompting, which guides the model through intermediate reasoning steps on an example to enhance task comprehension and accuracy (Wei *et al.* 2022).

Across methods and tuning approaches, there has been mixed evidence on which strategies are most effective for improving stance detection. While some work has found that LLMs typically outperform SLMs (Burnham 2024; Maia and da Silva 2024; Mets *et al.* 2024), other studies have found that LLMs can vary widely in performance (Cruickshank and Ng 2024; Kristensen-McLachlan *et al.* 2023; Mu *et al.* 2024). There is also been mixed evidence on few-shot prompting (Gül, Lebret, and Aberer 2024; Perez, Kiela, and Cho 2021) and cross-target tuning performance (Li *et al.* 2021; Ng and Carley 2022).

Additionally, prior research has highlighted the potential for LLMs to produce unreliable estimates due to factors, such as data contamination across model versions (Aiyappa *et al.* 2024), prompting choices (Cruickshank and Ng 2024), and biases embedded in training data (Gover 2023; Motoki, Neto, and Rodrigues 2024; Rozado 2024). However, less applied work has systematically compared the performance of different fine-tuning approaches and prompting strategies, particularly in contexts involving multiple target subjects or continuous measures of stance (Zhang *et al.* 2024).

Existing research primarily benchmarks stance detection methods using binary or categorical measures, which limits the ability to compare variations in sentiment intensity (Alturayeif *et al.* 2023; Schiller, Daxenberger, and Gurevych 2021; Zhang *et al.* 2024). However, several existing stance evaluation benchmarks have used continuous stance scores to capture polarity scales (Saif *et al.* 2013; Thelwall, Buckley, and Paltoglou 2012), emotional intensity (Mohammad and Bravo-Marquez 2017; Mohammad *et al.* 2018), or aspect-based sentiment (Dai *et al.* 2025). Further, more recent work has begun to explore continuous stance estimation, either by annotating stance scores on a continuous scale (Li and Conrad 2024) or by positioning political texts along continuous dimensions (Bergam, Allaway, and Mckeown 2022; Le Mens and Gallego 2024). Evaluating stance with continuous measures is valuable for many political questions, as it allows for a more nuanced expression of stance, enables ranking of statements by intensity, and facilitates richer statistical comparisons across groups. However, there remains limited evidence comparing the performance of stance detection methods using continuous measures.

This article compares the performance of lexicon-based methods, SLMs, and LLMs in detecting stance. We evaluate model performance across two datasets of statements about major U.S. presidential candidates posted on Twitter (now X) before and after the 2020 U.S. Presidential Election. We also assess the impact of three fine-tuning approaches: cross-target tuning (using models trained on a politician's party affiliation), in-target tuning (using models trained on hand-coded stance scores), and prompt engineering. Our results show that while cross-target tuning and prompt engineering improve performance, in-target tuning consistently produces the most accurate models. Notably, in-target tuning is especially important when analyzing statements that reference multiple target subjects, where it significantly outperforms other approaches.

2. Methods Considered

We estimated sentiment scores using six candidate models: two LBAs, VADER (Valence Aware Dictionary for Sentiment Reasoning) (Hutto and Gilbert 2014) and EmoLex (NRC Word-Emotion Association Lexicon) (Mohammad and Turney 2010); two SLMs: SiEBERT, which classifies stance using categorical sentiment scores (Hartmann *et al.* 2023), and DeBERTa, which uses natural language inference to categorize stance (He *et al.* 2020; Laurer *et al.* 2024a); and two LLMs, GPT-3.5 Turbo and GPT-4 Omni (Brown *et al.* 2020).

We present results using a limited selection of methods, though we consider a wider range of approaches in the Supplementary Material.[1] We prioritized evaluating SLMs over other supervised methods, as previous research has shown SLMs consistently outperform approaches, such as support vector machines, naive Bayes, logistic regression, and long short-term memory models (Hartmann *et al.* 2023).

We provide a brief description of each model in the following sections, along with a summary of all included methods in Table A.1 in the Supplemental Material. Code for this project is available through GitHub, along with the input data, estimates, and summary result tables at the Harvard Dataverse. All models were trained and deployed on a vast.ai instance with an RTX A4000 GPU and an Xeon W-2175 CPU, using Python 3.11 and R 4.2.2.

2.1. LBAs

LBAs assign sentiment values to words in sentences using predefined dictionaries. These dictionaries are typically created by either assigning scores to words based on qualitative judgments of their polarity (e.g., the word "good" might have a sentiment score of 0.7, while "great" scores 0.8) or by using models to derive latent sentiment scores from human-coded texts (Pang, Lee, and Vaithyanathan 2002). Each word in the text is scored, and the overall sentiment score is the average of these values. LBAs can vary depending on the method used to construct the dictionary or the specific domain considered for sentiment scoring. For instance, some dictionaries are based on researchers evaluating word sentiment within specialized corpora, such as political speeches or news articles (Deng and Wiebe 2015).

Some lexicon-based methods use expanded approaches that include features like valence shifting and semantic contexts, such as the approach VADER. Valence shifting involves assigning a multiplicative value to words which modify sentiment scores in the included text. Valence shifting assigns a multiplicative factor to words that modify the sentiment of other words in the text. These shifters are pre-specified by the researcher and are typically adjectives that alter the sentiment of nearby words. For example, the word "very" might increase the sentiment value of the next word by a factor of 1.15. In a sentence like "Spot is a very good boy," if the word "good" is assigned a sentiment score of 0.7, the bigram "very good" would receive a sentiment score of $1.15 \times 0.7 = 0.805$.

We estimated stance scores using all LBAs discussed in Section A.1 of the Supplementary Material. We implemented these approaches using the R packages sentimentr v2.9.0. and vader v1.2.1. Results can be found in Section A.10 of the Supplementary Material.

2.2. SLMs

Supervised models have long been used to estimate text sentiment, including approaches, such as Naive Bayes, support vector machines, logistic regression, decision trees, and neural networks (Wankhade, Rao, and Kulkarni 2022). In this article, we evaluate the performance of SLMs, which previous research

[1] This includes additional lexicon-based models: SenticNet 4 (Cambria *et al.* 2016), SO-CAL (Taboada *et al.* 2011), Sentiwordnet3 (Baccianella, Esuli, and Sebastiani 2010), Hu & Liu (Hu and Liu 2004), and MPQA (Deng and Wiebe 2015); additional SLMs, DistilBERT (Sanh *et al.* 2019) and RoBERTa tuned using TweetEval (Loureiro *et al.* 2022), and an additional LLM, GPT-4. We found that these methods underperformed relative to those discussed in the article and chose to present only a limited set of results for the sake of parsimony.

has shown to outperform traditional supervised models in sentiment analysis and stance detection tasks (Devlin *et al.* 2019; Hartmann *et al.* 2023).

SLMs leverage transfer learning, beginning with a transformer-based model that predicts masked words based on context-dependent vectors (Vaswani *et al.* 2017). The model's parameters are then updated through fine-tuning on a classification task, typically using a binary or categorical outcome variable (Hartmann *et al.* 2023; Loureiro *et al.* 2022).

In this analysis, we examine two approaches to SLMs: pretrained transformers and paired-sequence entailment classification. Specifically, we evaluate the performance of SieBERT, a fine-tuned version of the RoBERTa model trained on sentiment datasets from 217 academic publications (Devlin *et al.* 2019; Hartmann *et al.* 2023; Liu *et al.* 2019), and BERT-NLI, a fine-tuned version of the DeBERTa-V3 model trained on various additional datasets (He *et al.* 2020; Laurer *et al.* 2024a, 2024b). These models differ in their classification approaches. Pretrained transformers, such as SieBERT, are first trained using masked language modeling and then fine-tuned on domain-specific tasks (in this case, stance detection) to predict labels (Hartmann *et al.* 2023). In contrast, paired-sequence entailment models, like BERT-NLI, use natural language inference to perform universal classification tasks, determining whether input text *entails* a given hypothesis (in this case, a hypothesis related to the stance of the text) (Laurer *et al.* 2024a).

We implemented these approaches using the Python packages Transformers and Torch. Additional details on the implementation of these methods are available in Section A.3 of the Supplementary Material.

2.3. Generative LLMs

Generative LLMs leverage advanced architectures to understand language in context, building on transformer models and incorporating much larger datasets than SLMs. These models are further refined through reinforcement learning, based on human–AI interactions. The most widely used approach in large language modeling is generative pre-trained transformers (GPTs), which generate text based on human-written prompts (OpenAI 2022; Radford *et al.* 2018). In this study, we used two versions of GPT: ChatGPT-3.5 Turbo (v. 0125) and ChatGPT-4 Omni (v. 2024-08-06) to estimate stance. All models were set with a temperature hyperparameter of zero to produce more deterministic outputs (Alizadeh *et al.* 2024; Burnham 2024). Across multiple runs, we found that a temperature of zero allowed us to reproduce evaluation metrics consistently, though we observed slight fluctuations in the estimated scores on specific tasks, typically within a 0.1-point margin.

2.3.1. Prompts

For LLMs, we used four prompts to estimate stance scores:

- A sentiment prompt, which tasked the model with estimating a continuous sentiment score of the provided text.
- A stance prompt, which tasked the model with estimating a continuous stance score of the provided text.
- A few-shot prompt, which tasked the model with estimating a continuous stance score of the provided text, including four examples of existing tweets along with an associated stance score.
- A chain-of-thought prompt, which used three successive prompts to estimate continuous stance, using prompt (2) first, then tasking the model to explain the reasoning behind the estimated score, then including this reasoning in a final prompt as an example (. He *et al.* 2024; Kojima *et al.* 2022; Wei *et al.* 2022).

For each prompt, we appended text from tweets to the end of the prompt input. For prompts 2–4, we included information on whether the target subject was "Biden" or "Trump." All calls to the GPT API were made in separate sessions to ensure independence in model responses, and the temperature parameter was set to zero to ensure less stochastic responses (Alizadeh *et al.* 2024; Burnham 2024).

In addition to these prompts, we tested two alternative formulations of prompt (2): one which changed the prompt language slightly and a separate prompt which used the same language but tasked the model with generating a binary stance score (−1 or 1). Section A.8 of the Supplementary Material includes all prompt language and additional details on prompting strategies. After receiving responses from the GPT API, we did additional post-processing to ensure results were in a vector format by using regular expressions on the obtained responses to extract stance scores.

3. Datasets

We obtained two corpuses of data containing texts posted to Twitter (i.e., tweets), which we used to evaluate the performance of each candidate method. Note that such texts are brief, containing no more than 280 characters each. Providing additional details on each corpus:

Politicians: First, we assembled a "politician dataset" which contains all tweets posted by sitting members of the U.S. Congress (Senate and House of Representatives) during the 2020 U.S. Presidential election, between September 20, 2020 and January 21, 2021. We further narrowed the set of tweets to those that referenced either Joseph Biden or Donald Trump. We ultimately compiled 20,442 texts, composed of 5,508 tweets referencing Biden (3,902 from Democrats and 1,606 from Republicans), in addition to 14,934 tweets referencing Trump (4,109 from Republicans and 10,825 from Democrats). From this set of tweets, we randomly sampled 100 tweets concerning Biden and 100 tweets concerning Trump, which were scored by the study team and used to investigate how well party affiliation correlated with human-coded stance scores (we discuss this further in Section A.4 of the Supplementary Material).

Users: Second, we assembled a "user dataset" which contained tweets from the 1% sample of Twitter users. To assemble the list of users, we collected all available tweets from the 1% Twitter Firehose API between January 5, 2020 and August 15, 2022. We note that the resulting user list was skewed toward high-volume users. For users on this list, we collected tweets posted during the study period from September 20, 2020 to January 21, 2021. We then narrowed the included texts to those containing either the word "Trump" or "Biden," then sampled 250 texts from this narrowed set to those that mentioned Joe Biden alone (and not Donald Trump), 250 texts that mentioned Donald Trump alone (and not Joe Biden), and an additional 126 texts that mentioned both Joe Biden and Donald Trump. Each sampled text was produced by a different Twitter user. We also sampled an additional 250 tweets mentioning Joe Biden (and not Donald Trump) and 250 tweets mentioning Donald Trump (and not Joe Biden), which were used exclusively for model tuning.

In addition to these two datasets, we applied models and estimated stance scores using data available in two previous studies: Li *et al.* 2021 and Kawintiranon and Singh 2021. We provide additional analyses regarding these studies in Section A.5 of the Supplementary Material.

For texts obtained by the study team, we collected information on the date the text was posted, IDs for the author, a tweet ID, and a binary indicator determining if the text was a retweet (i.e., one user sharing another user's tweet). We replaced any letters containing accents with an ASCII equivalent letter without an accent. We also processed texts to remove any emoticons (e.g., ";)"), website addresses, HTML code, images, or emojis.

3.1. Benchmark and Training Data

We evaluated methods' performance in each datset by comparing estimated stance scores with benchmark scores. We used different benchmark scores in the politician dataset and the user dataset.

In the politician dataset, we used the political affiliation of each Congress person to develop a benchmark score. Specifically, for each text, we set a benchmark score to equal 1 if the party affiliation (Democrat or Republican) of a representative aligned with the party affiliation of a given subject ("Biden" or "Trump") and −1 if otherwise. This coding was meant to replicate an assumption that a member of Congress will speak positively concerning presidential candidates who belong to the same party and negatively of candidates belonging to the opposing party.[2] For example, a tweet posted by Nancy Pelosi (a Democrat) containing "Biden" was assigned a benchmark score of 1, whereas a text by Nancy Pelosi containing "Trump" was assigned a score of −1.

For the user dataset, we developed stance scores by having two human coders review each text and assign a continuous stance value between −1 and 1 concerning each subject. We then averaged the human-coded stance scores to obtain a benchmark score. We evaluated inter-rater reliability across the two coders, finding high agreement in coded scores, with an intraclass correlation of 0.85 (95% CI: 0.835, 0.864). More details on inter-rater reliability can be found in Section A.2 of the Supplementary Material.

To facilitate evaluations with binary estimated and benchmark scores, all benchmark scores were also reformulated as binary by setting values below zero to "Negative" and values above zero to "Positive."

We hypothesize that hand-coded scores are a better representation of true stance than our party affiliation proxy. Accordingly, our primary conclusions are based on evaluating performance using the users dataset. However, we also evaluate performance using the politician dataset, which provides a robustness check for the performance of cross-target tuning.

These benchmarks also provide a means by which to fine-tune the methods. In particular, the party affiliation benchmarks are used for in-target tuning of the politicians dataset and cross-target tuning of the users dataset, whereas the human-coded benchmarks are used for cross-target tuning of the politicians dataset and in-target tuning of the users dataset. We provide more details on tuning in the following section.

In Section A.4 of the Supplementary Material, we evaluate the validity of using party affiliation as a measure of stance by comparing it to human-coded stance scores in the politician dataset. In summary, we found that political affiliation was highly correlated with human-coded stance scores ($r = 0.87$). We also investigate using DW-Nominate scores as a proxy measure (Carroll *et al.* 2009; Poole and Rosenthal 1985), though we find this measure does not exhibit as strong a correlation with human-coded stance scores.

Note that all analyses presented in the results section (Section 6) involve comparisons using continuous or binary benchmark scores. In Section A.6 of the Supplementary Material, we also consider performance using categorical (ordinal) benchmark scores.

4. Model Tuning

We tuned SLMs and LLMs models using data from both the politician dataset (using a party affiliation proxy) and the user dataset (using human-coded stance scores). Each of these models was then applied to both datasets, yielding a set of in-target tuned models and a set of cross-target tuned models for each of the two data sources.

The benchmark scores from each respective dataset were used as the outcome variable for tuning models. Note that tuned models were constructed independently for each target ("Biden" or "Trump"). For both datasets, we split available data for each subject into a tuning sample, separated further into a training, test, and evaluation set.

For the politician dataset, 80% of the texts for each subject were used for tuning (further separating into 80% for training and 20% for testing), and the remaining 20% was withheld for out-of-sample evaluation. For the user dataset, 250 texts mentioning only "Trump" and 250 texts mentioning only "Biden" were used as the tuning samples (200 in the training set and 50 in the test set), and an additional 250 texts mentioning only "Trump" and 250 texts mentioning only "Biden" were withheld

[2] Four politicians in the dataset were identified as independents. For these politicians, we recoded party affiliation to match their congressional caucus. Accordingly, we recoded Bernard Sanders and Angus Stanley King, Jr. to "Democrat," and Justin Amash and Paul Mitchell to "Republican."

for evaluation.[3] The 126 user texts mentioning both subjects was used only for evaluation—we applied the subject-specific tuned models separately to each of these texts (to generate two scores for the text—one for each subject).

The models tuned using party affiliation were applied to estimate stance scores on both the politician dataset (to evaluate in-target tuning performance on the party affiliation measure) and user datasets (to evaluate cross-target tuning performance). Likewise, the models tuned using the user dataset were applied to estimate stance scores on the politician dataset (to evaluate cross-target tuning performance on a party affiliation classifier) and the user evaluation sample (to evaluate in-target tuning on the hand-coded stance scores).

Conventionally, researchers have used fivefold or tenfold cross-validation to evaluate model accuracy under slightly different samples. We were limited to a single fold for cross-validation due to the costs involved in tuning GPT models. We discuss this limitation further in Section 7.1.

Lastly, we also tested tuned models using the politician dataset and DW-Nominate as a proxy measure but found these models underperformed models tuned using political affiliation as a proxy (more details on these results can be found in Section A.4 of the Supplementary Material). We also provide some additional details on tuning SLMs in Section A.3 of the Supplementary Material.

5. Evaluation Metrics

We summarize model performance primarily by calculating Pearson correlations between the estimated scores and the benchmark scores. The Pearson correlation is used to measure the association between continuous variables. However, we are also interested in assessing the models' ability to quantify stance using binary (positive/negative) or categorical (e.g., positive/neutral/negative) measures. To make these comparisons, we transform the estimated scores to the desired scale and then apply Matthews' correlation coefficient (phi) (Matthews 1975) for binary scores and Cramér's V (Cramér 1999) for categorical scores. Additionally, we use the point-biserial correlation (Sheskin 2003) when measuring the association between binary and continuous scores.[4] Results for binary and categorical scores are provided in the Supplementary Material.

The Pearson correlation is used because of its advantages in assessing model performance. Specifically, Pearson correlations are invariant to the labeling of positive and negative classes and are less sensitive to class imbalances, unlike metrics such as the F1-score, which can be influenced by class distribution (Baldi *et al.* 2000; Chicco and Jurman 2020, 2023).

We also evaluated model performance using root mean squared error (RMSE) and mean squared error (MSE) for continuous benchmark scores, and F1-score, accuracy, precision, and recall for binary benchmark scores. However, it is important to note that these metrics are not suitable for comparing results across different types of data. For instance, RMSE should not be used to compare continuously estimated scores with binary benchmarks, such as the party affiliation metric in the politician dataset.

Across datasets, summary statistics are reported separately for "Biden" and "Trump" as a target subject. For the user dataset, we also separated out texts by the number of subjects, with results indicating whether a text mentioned a single subject or both subjects.

6. Results

In the following sections, we detail method performance. In Section 6.1, we report results using the politician dataset and show how methods without additional tuning distinguish stance across party affiliation. Then, in Section 6.2, we compare method performance without additional tuning on hand-coded stance scores within the user dataset, comparing results across texts that contain a single subject

[3] We chose this amount since previous work has suggested that between 250 and 500 examples are sufficient to train models (Alizadeh *et al.* 2024; Laurer *et al.* 2024b)

[4] Note that Matthews' and point-biserial correlation coefficients are mathematically equivalent to the Pearson correlation.

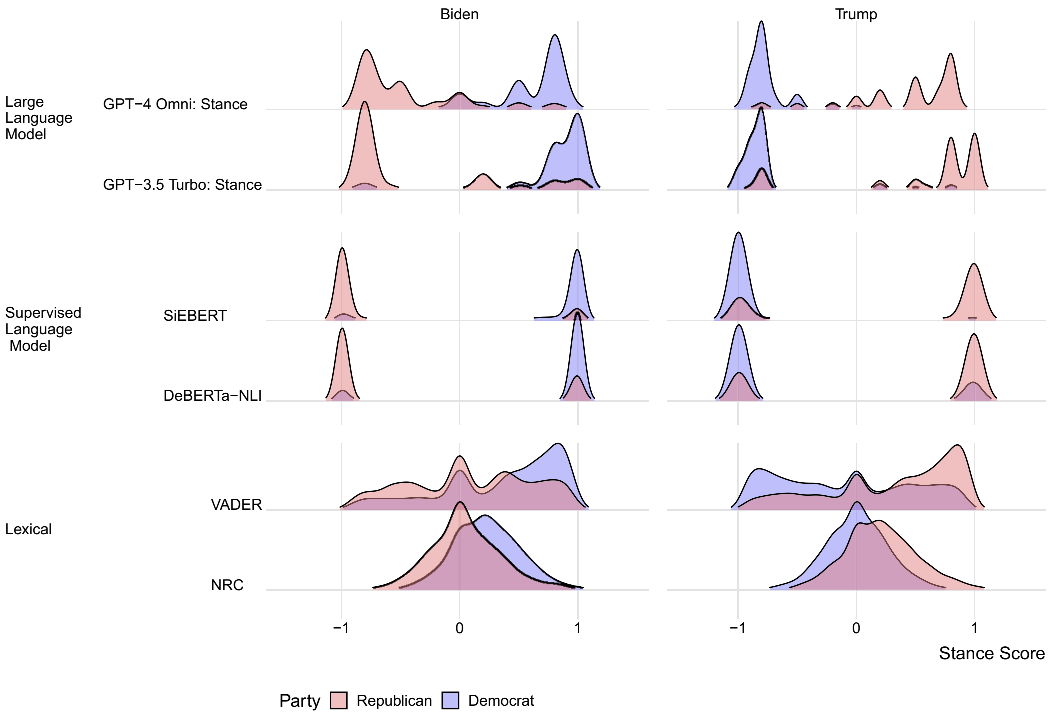

Figure 1. Distribution of estimated stance scores by subject for politician texts.

or multiple subjects. Section 6.3 then compares method performance after tuning, showing cross-target and in-target tuning performance using the party affiliation and hand-coded stance score benchmarks. We further compare performance when benchmarks are measured as a binary variable. Lastly, we compare LLM results across several different prompting strategies in Section 6.4. We also provide example texts from each dataset, along with estimated stance scores by method, in Section A.9 of the Supplementary Material.

6.1. Do Stance Scores Align With Party Affiliation Without Tuning?

Figure 1 displays the distribution of estimated stance scores across political affiliations. We assume that public statements by legislators will express positive stance concerning a political candidate who belongs to their own political party and negative sentiment concerning candidates from the opposing party (see Section A.4 of the Supplementary Material discussing this assumption). In short, results are expected to show separation of stance scores based on party affiliation.

The left facet of this figure displays scores for texts containing the subject “Biden,” while the right facet displays scores for texts concerning the subject “Trump.” The blue density shows estimated scores for Democrat members, while the red density shows estimated scores for Republican members. The purple density shows the overlap in the two empirical distributions.

As shown in Figure 1, the LBAs—VADER and NRC—estimate a similar distribution of scores across political affiliations and target subject, showing little separation based on the author’s party affiliation. Conversely, SLMs and LLMs produce more divergent scores between the two political parties. We observe that SLMs produce scores clustered around −1 and +1, which is a consequence of these methods classifying texts as a binary variable. The LLMs produce continuous-valued scores; however, there remain few neutral-valued scores for either subject or political party.

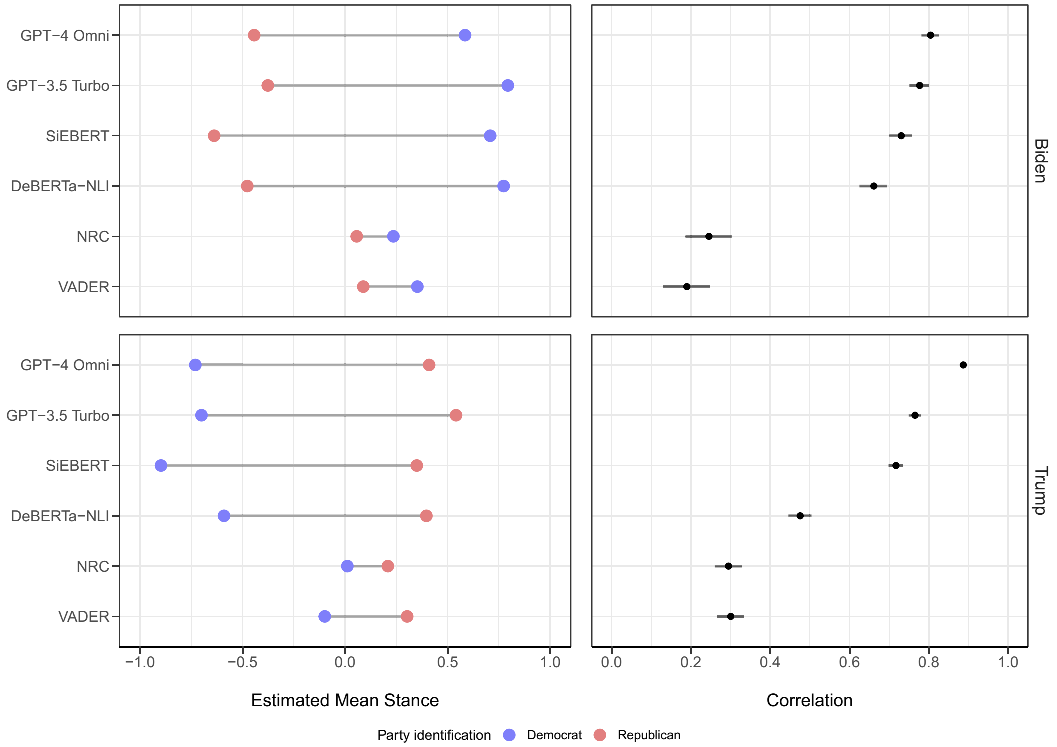

Figure 2. Estimated mean stance score by political affiliation and target subject (left) and correlation between party affiliation and estimated stance scores by target subject (right).

To further determine if estimated scores align party affiliation, we display the mean of estimated stance scores by party affiliation and target subject in Figure 2, along with the Pearson correlation (and the 95% confidence interval for the correlation) between estimated scores and party affiliation. Lexicon-based methods do not estimate a meaningful difference in scores between the political parties for either subject, and exhibit a small correlation between estimated scores and the benchmark.

SLMs and LLMs produce more divergent mean scores between the parties for both subjects, with the SiEBERT model having the widest difference between the parties of 1.35 for the subject "Biden" and 1.24 for the subject "Trump." For all methods displayed in Figure 2, we find the difference between the two parties is significant (using an unpaired t-test with unequal variance for the mean difference).

However, despite the size of mean differences being comparable across SLMs and LLMs, there is a wide range of correlations between estimated scores and party affiliation. LLM models have the highest correlations for both subjects, with GPT-4 Omni having a correlation of 0.80 (0.78, 0.83) and 0.89 (0.87, 0.89) for texts with "Biden" and "Trump," respectively. Lexicon-based models display the lowest correlation with the benchmark, with VADER having a correlation of 0.19 (0.13, 0.25) for "Biden" texts and NRC having a correlation of 0.29 (0.26, 0.32) for "Trump" texts.

Overall, we find that SLMs and LLM models estimate stance scores which correlate well with party affiliation, while lexicon-based methods fail to produce separation in mean scores across parties.

6.2. Do Stance Scores Align With Human-Coded Stance Scores Without Tuning?

We next evaluated each method's performance when estimating stance on user texts. We hypothesized that all approaches would struggle to recover stance without additional tuning (Bestvater and Monroe 2023), particularly when texts mention both subjects. Accordingly, we present results for user data

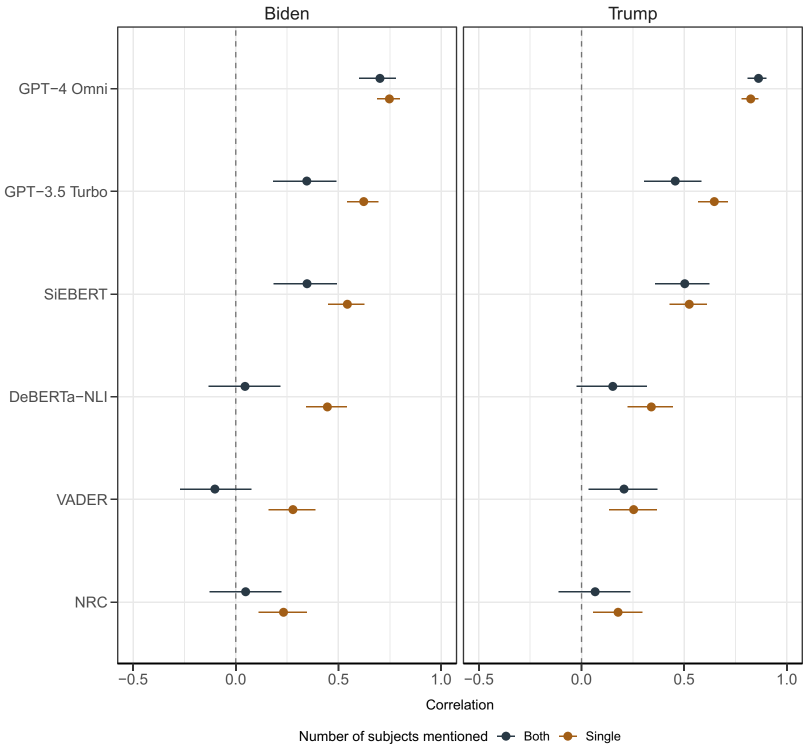

Figure 3. Correlation of estimated stance scores with human-coded stance scores, by target subject and number of target subjects.

separately by whether a text mentions a single subject or both subjects. In addition to these results, we also provide scatter plots of estimated scores compared to human-coded stance scores in Section A.11 of the Supplementary Material.[5]

Figure 3 displays correlations between estimated scores and human-coded stance scores for each method. Facets in this figure show results stratified by the target subject ("Biden" or "Trump"), while color displays correlations separated by whether a text mentioned a single subject or both subjects. The figure shows that GPT-4 Omni has the highest correlation, 0.783 (0.777, 0.789), compared to alternative methods, across each subject and the number of subjects. For all methods, estimated correlations are lower when both subjects are mentioned in a text compared to a single subject, except GPT-4 Omni and the subject "Trump" (though this result is statistically non-significant).

Estimates from the SLMs and LBAs show weak or inverse correlations with the benchmark scores, especially when the subject is "Biden." Among the methods considered, VADER exhibits the weakest correlation, with an inverse correlation of −0.1 (−0.27, 0.075) when the text includes both subjects and scores are estimated for "Biden." In contrast, SiEBERT achieves the highest correlation among the SLMs and LBAs, with a correlation of 0.54 (0.45, 0.62) for texts mentioning "Biden" as the sole subject. Overall, these results suggest that LLM models, particularly GPT-4 Omni, perform well in estimating scores without additional tuning. The performance gap between LLMs and other models is most pronounced when multiple subjects are mentioned, with SLMs and LBAs underperforming relative to when only a single target subject is present.

[5] Similar scatter plots for all datasets are available in this section.

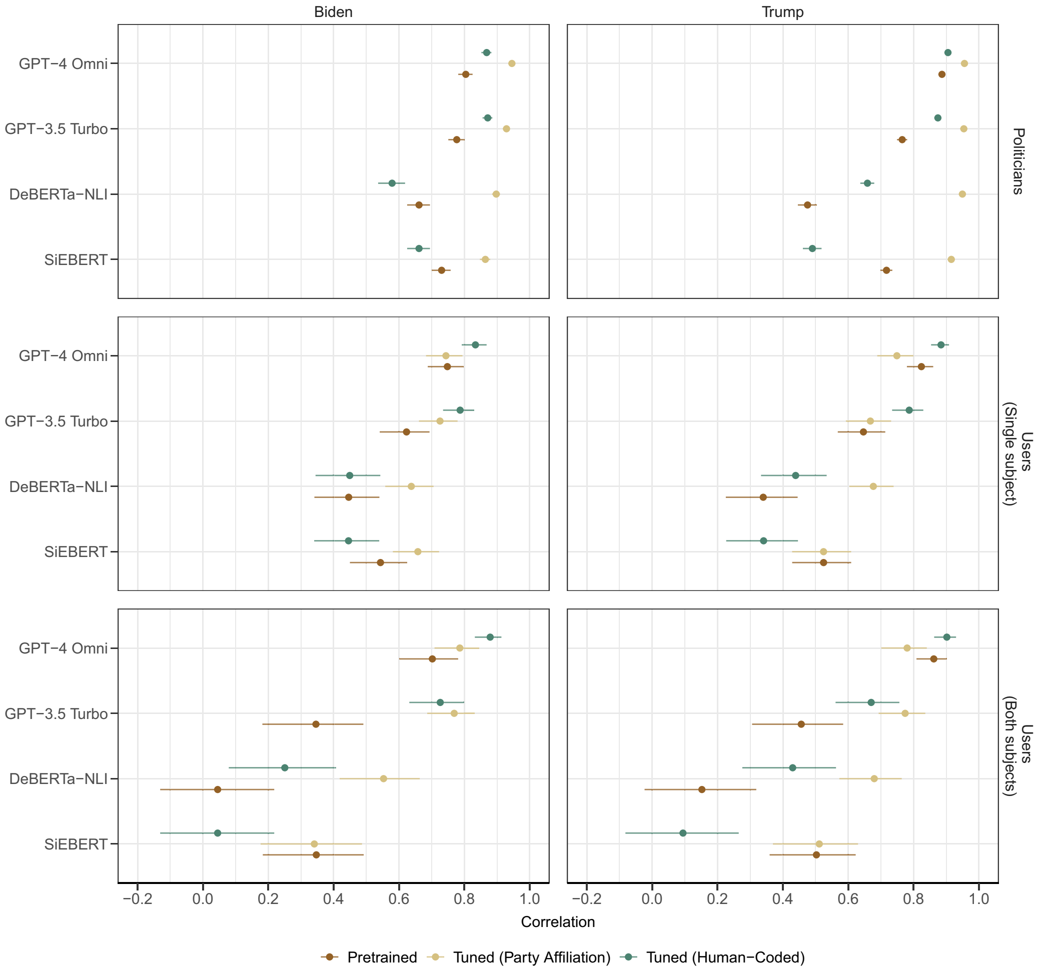

Figure 4. Correlation of estimated stance scores with party affiliation (politician data) and human-coded stance scores (user data), by method, tuning approach, target subject, and number of subjects.

6.3. How Does Tuning Change Performance?

To improve the performance of supervised methods and LLM models, we attempted to tune these approaches using both in-target and cross-target tuning. We present the results of tuned models on out-of-sample politician data and user data in Figure 4.

Note that for results in the politician data, the benchmark variable is party affiliation. Accordingly, models tuned using party affiliation would be an “in-target” approach, while those tuned using human-coded stance scores would be “cross-target.” Conversely, for the user data which uses human-coded stance scores as a benchmark variable, tuning using human-coded stance scores would be “in-target” while those tuned using party affiliation would be “cross-target.” We again separate user data into texts which mention either a single subject or both subjects.

Figure 4 shows correlations of estimated scores with each dataset’s respective benchmark: political affiliation and human-coded stance scores. We see that after some form of tuning, all methods improve in performance. Across datasets and target subjects, in-target tuned GPT-4 Omni models consistently perform the best. Some form of tuning leads to improvement over pretrained models for all methods (with the possible exception of SiEBERT), although the magnitude of this improvement is possibly smaller for GPT-4 Omni. Otherwise, specific model improvements vary depending on the tuning approach and benchmark variable.

In the politician dataset, we find that in-target tuning (with party affiliation) consistently leads to dramatic improvements in model performance (compared to pretrained models) across methods and subjects. Conversely, we see evidence that cross-target tuning (with human-coded stance scores) degrades the performance of the SLMs (DeBERTa NLI and SiEBERT) over pretrained analogs. For the LLMs (GPT), we see that cross-target tuning leads to a moderate improvement in model performance over the pretrained versions (while still under-performing compared to in-target tuning). Overall, these results indicate that (1) tuned models using the proxy measure reliably capture political affiliation, which provides evidence that the cross-target tuning models in the user dataset will reflect performance exclusively on the stance detection task (since the proxy measure appears to be well calibrate) and (2) cross-target tuned models using hand-coded scores lead to slight improvements in estimating party affiliation.

By contrast, the user dataset (which may be better suited to produce generalizable conclusions regarding estimation of stance since evaluations are based on human-coded benchmarks and not party affiliation) yields similar findings as well as additional nuance when evaluating performance. To start with the SLMs, in-target tuning (with human-coded stance scores) fails to yield statistically significant improvements over the pretrained models while potentially degrading performance in some cases (e.g., the SiEBERT model for Trump when both subjects are mentioned). However, cross-target tuning (with party affiliation) leads to statistically significant improvements over the pretrained versions of the DeBERTa-NLI models but fails to show similar improvement with the SiEBERT model. For the GPT models, both in-target and cross-target tuning improve performance over pretrained models; however, the benefits offered by in-target tuning are more substantial.

For models estimating stance, across models, subjects, and approaches to tuning (with the possible exception of GPT-4 Omni with Trump), the benefits of tuning appear to be most substantial when both subjects are mentioned, compared to single subjects. That is, we find that tuning narrows the performance gap for estimating stance between texts containing single or multiple subjects. This finding is most dramatic for both subject texts. Additionally, the tuning approach which most improves performance can differ by method.

While human-coded stance scores in the user data are continuous, estimated stance scores using the cross-target tuned model (based on party affiliation) are nearly binary. As such, it may be more feasible for in-target tuning (which is designed to yield continuous estimated scores) to maintain the distributional properties of the hand-coded benchmarks. Consequentially, findings could differ if the objective was to estimate binary stance (positive or negative), ignoring gradations.

To investigate this, we transformed the human-coded stance scores and all estimated scores to binary values which indicate “positive” or “negative” stance.[6] For this analysis, we used a modified prompt for GPT procedures, which specifically requests a binary output. Figure 5 shows results that are analogous to those in Figure 4 when binary benchmarks and estimated scores are used.

The results in Figure 5 are similar to those displayed in Figure 4. However, we find that overall performance tends to decrease when using binary values compared to the results shown in Figure 4. The results also show that GPT-3.5 Turbo and SLM models in the user dataset indicate cross-target tuning (with party affiliation) outperforms in-target tuning. For GPT-4 Omni for Biden, cross-target tuning offers the best performance for Biden, whereas for Trump, in-target tuning appears to be optimal (but note that the differences are not statistically significant in either case).

[6] To make this transformation, we take a conservative assumption to recode zero-valued benchmarks, assuming a benchmark of zero is against expectation for binary re-coding. For example, a Republican politician posting a neutral tweet about Biden will be recoded as “positive” text.

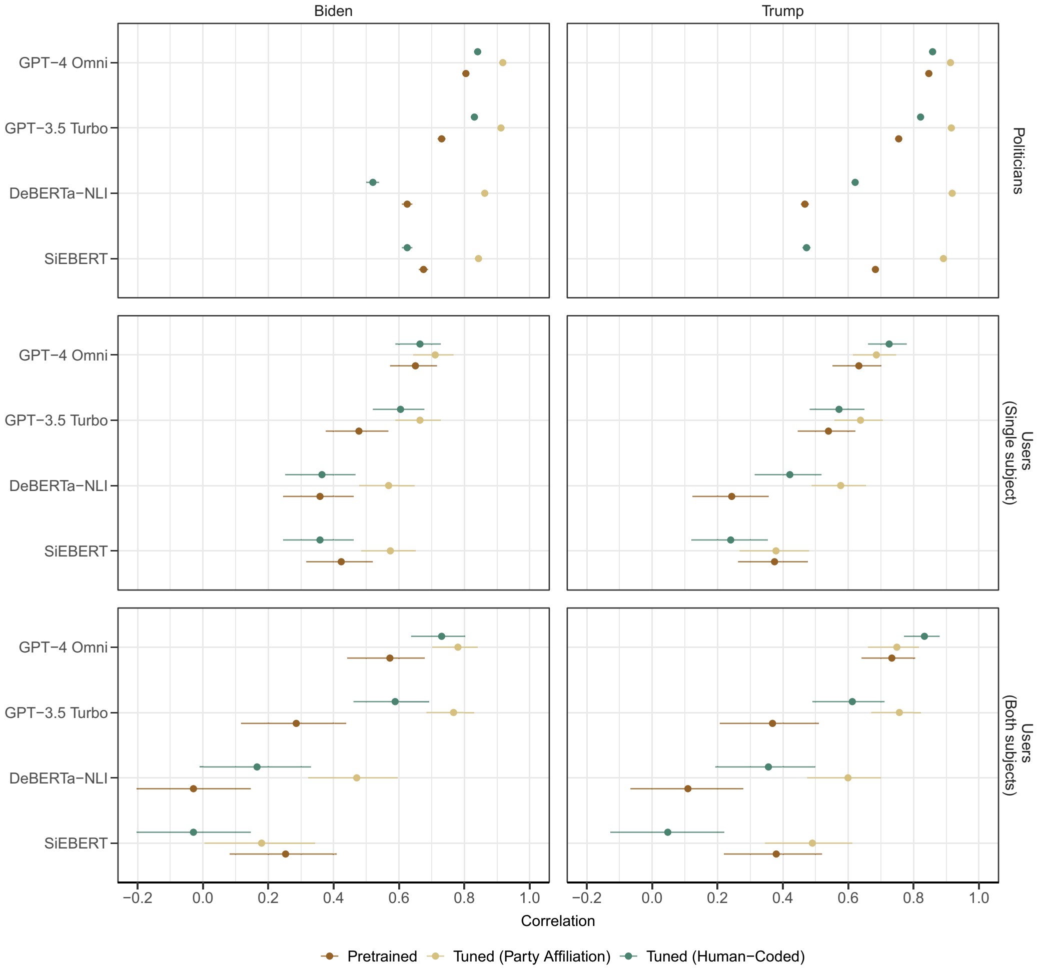

Figure 5. Correlation of binary stance scores with party affiliation (politicians dataset) and human-coded stance scores (users dataset), by method tuning approach, target subject, and number of subjects.

6.4. How Does Prompting Change Performance in LLMs?

LLMs have the added complexity of requiring a prompt which specifies a model task. In the previous section, we presented all results using a simple prompt which tasked the LLMs with providing a continuous (or for Figure 5, binary) stance score. Here, we present results across the wider array of prompts that were detailed in Section 2.3.1 and Section A.8 of the Supplementary Material. These are summarized as Chain-of-thought, Few-shot, Stance, and Sentiment.

The Stance prompt was used for all results presented in the previous sections. Complex prompting (Chain-of-thought and Few-shot) is often described as an alternative to tuning. As such, we compare these prompting strategies to in-target tuned models using the stance prompt.

Results for each prompt are provided in Figure 6, segmented by target subject, dataset (with user data also segmented by the number of subjects), and model (GPT-3.5 Turbo and GPT-4 Omni).

Overall, the evidence suggests that complex prompting strategies, such as chain-of-thought, may lead to improved model performance compared to simpler prompts. In several instances, most notably for GPT-4 Omni models, these prompting strategies lead to estimates which are comparable in performance to tuned model results. Nonetheless, the tuned models consistently yield the highest correlations; however, the differences between tuned models, chain-of-thought, and few-shot are not statistically

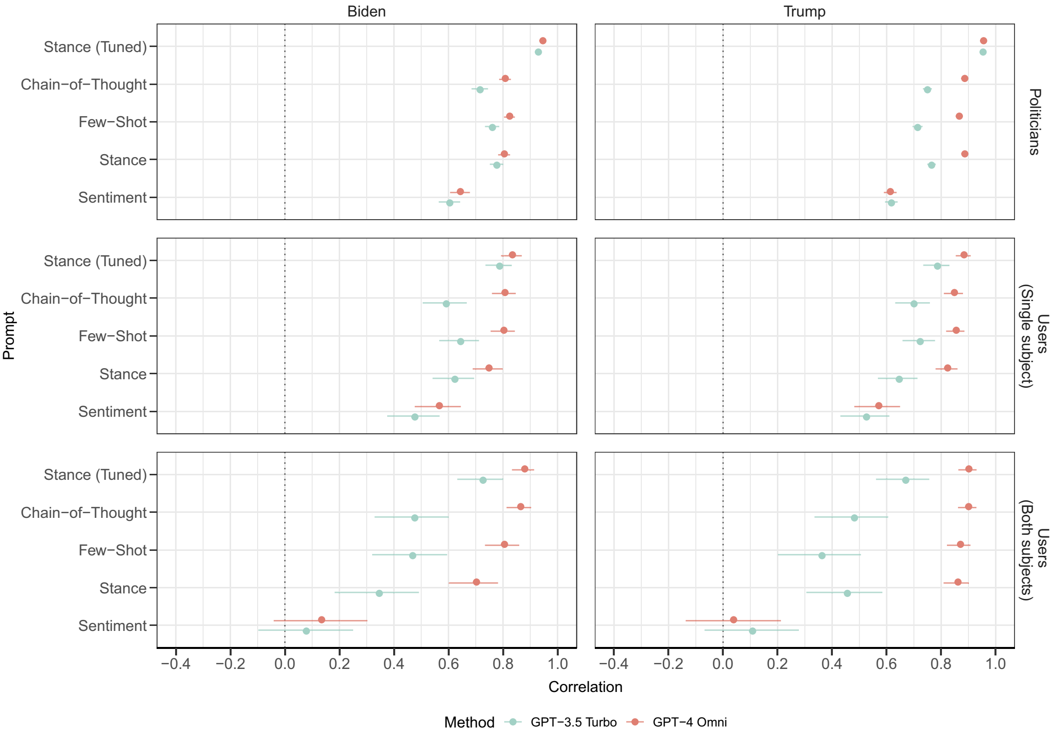

Figure 6. Correlation of continuous stance scores with party affiliation (politicians dataset) and hand-coded stance scores (user dataset), by prompt, method, target subject, and number of subjects.

significant in several cases when GPT-4-Omni is used (the difference is more distinct for GPT-3.5 Turbo). The improvements offered by tuning over alternative prompting strategies are largest when evaluating performance capturing the proxy measure of political affiliation, noting that the politician dataset had a greater sample size when conducting in-target tuning.

We see that a sentiment prompt performs relatively poorly, in particular for texts from the user dataset involving both subjects. This points to the benefits of directing the LLM toward a specific subject for stance detection and the sensitivity of models to prompting.

Pretrained models also tend to have higher performance when the subject target is Trump rather than Biden. Performance differences between subjects are non-significant in the user dataset but significant in the politician dataset. For example, the GPT-4 Omni Chain-of-Thought has a correlation of 0.8 (0.79, 0.83) for Biden texts and 0.88 (0.87, 0.89) for Trump, performing an average 9.7% worse.

Lastly, the results in Figure 6 indicate, similar to those shown in the previous sections, that more recent LLM models, specifically GPT-4 Omni, outperform older GPT models in both datasets. However, when evaluating results on hand-coded stance scores (users dataset), average differences are non-significant. This result suggests that for more recent LLMs, the relative benefit of tuning for estimating stance scores may be muted, compared to earlier model versions such as GPT 3.5 Turbo.

7. Conclusion

7.1. Limitations

These results carry several limitations. First, we used a limited number of out-of-sample cross-validation folds to evaluate model performance given computational costs. This also limited our ability to estimate replicate runs of LLMs. These models can exhibit variability in outputs from the same input (Aiyappa *et al.* 2024), though we aimed to minimize stochastic responses by setting a low value for the temperature hyperparameter. In addition, empirical work has found fewer folds may be preferable for method selection procedures, and results in Figure 4 display reasonably small uncertainty intervals on the test data (Zhang and Yang 2015). Also, many of our findings are based on comparisons of uncertainty intervals that did not overlap, indicating these findings would likely still hold with additional added cases.

Second, we did not include open-source large language architectures. Additional research should test whether these findings hold when using open-source LLM models. Previous evidence has suggested that SLMs and proprietary models may outperform fine-tuned open-source models (Alizadeh *et al.* 2024; Cruickshank and Ng 2024). However, there are notable research benefits to using open-source models beyond performance, such as better cost-effectiveness, transparency in the model parameters, and improved reproducibility (Alizadeh *et al.* 2024; Burnham 2024).

Third, researchers should also test additional LLMs for stance detection since LLMs can exhibit bias, which can vary across stance topics and model versions. For example, existing work has found LLMs can display political biases (Gover 2023; Motoki *et al.* 2024; Rozado 2024) and produce biased results on specific topics (Zhang *et al.* 2024). Certain prompting strategies, like chain-of-thought prompting, can further exacerbate existing model bias (Ng, Cruickshank, and Lee 2024), and later versions of a model can incorporate data leakage from previous user queries, leading to biased results (Aiyappa *et al.* 2024).

7.2. Discussion

In this article, we compared the performance of stance detection methods across tuning approaches and prompting strategies on multiple datasets, focusing on estimating stance regarding political candidates in the 2020 election. Overall, we find that tuned LLMs outperformed tuned SLMs and LBAs. This result was most significant when evaluating performance on texts that contained more than one subject.

Pretrained SLMs exhibited weak correlations with human-coded stance scores, indicating it is risky for researchers to use these methods without additional tuning. However, LLMs were able to identify stance scores with reasonable accuracy even without tuning, particularly when using strategies like few-shot prompting and chain-of-thought prompting, or the most recent available LLM. LBAs consistently, and dramatically, underperformed compared to other methods. In tandem, these results suggest that researchers should consider using LLMs to classify stance, particularly for more complex texts which contain multiple subjects. Additionally, researchers should avoid the use of LBAs in favor of state-of-the-art methods (Bestvater and Monroe 2023).

Most methods performed better when estimating stance scores on texts containing a single subject rather than multiple subjects. However, the gap in performance is much wider for SLMs and LBAs, which often fail to disambiguate stance when texts contain multiple subjects and can exhibit inverse correlations with human-coded stance scores. For SLMs, cross-target tuning led to a significantly narrowed gap in performance between single subject and multiple subject texts. Ultimately, the results suggest that for more complex texts with multiple subjects, researchers should use tuned models and consider using LLMs, along with tuning or prompting strategies.

We also found that methods had higher performance when using continuous measures of stance to tune models. Across multiple methods, we found that estimated correlations were larger when tuned using a continuuous measure and more reliably captured the distributional properties of human-coded stance scores. As such, we recommend researchers consider collecting continuous stance measures for tuning purposes, particularly since continuous measures are known to contain more statistical information than binary variables which can aid model performance (Altman and Royston 2006; Cohen 1983).

Our work also illustrates throughout multiple analyses that both in-target and cross-target tuning led to improvements. Although in-target tuning is usually recommended in favor of cross-target tuning (Osnabrügge *et al.* 2023; Vamvas and Sennrich 2020), our use of politicial affiliation to tune estimates of stance illustrates a circumstance where cross-target tuning may outperform in-target tuning across several methods. Our work suggests that researchers should consider additional opportunities to tune language models when the outcome (in this case, stance) is well correlated with a proxy measure. Researchers could also consider additional modeling strategies to leverage cross-target tuning, including multi-domain prompting strategies (Ding *et al.* 2024; Khiabani and Zubiaga 2025) and ensemble models (Li and Caragea 2021; Sobhani, Inkpen, and Zhu 2019).

Across datasets, more detailed prompting strategies tended to improve performance but often did not improve on in-target tuned model results. However, when using GPT-4 Omni, few-shot prompting led to performance which nearly matched tuned model results. This result suggests that more recent LLM architectures may benefit less from in-target tuning. This finding stands in contrast to existing work which has found few-shot prompting can have highly variable performance (Zhao *et al.* 2021) and may not improve on simpler prompts (Burnham 2024). For older architectures—GPT-3.5 Turbo and GPT-4—prompting strategies exhibited much wider variation in performance across datasets (Zhu *et al.* 2023).

These results also show that pretrained LLMs tend to have higher performance when texts mention Trump rather than Biden, which may be reflective of data used to train foundational models. This observation held across methods, datasets, and prompts. This suggests researchers should exercise caution when using LLMs for stance detection, comparing and validating results across multiple approaches (Grimmer and Stewart 2013).

7.3. Future Directions

More work is needed to determine how prompting strategies might improve stance estimation. Burgeoning evidence indicates that conventional human-language prompts may not lead to the best performance. Instead, researchers may need to use additional modeling strategies to develop better prompts, such as reinforcement learning (Deng *et al.* 2022), automated prompt generation (Gao, Fisch, and Chen 2021), or collaborative LLM models (Lan *et al.* 2024), which could differ lead to prompts which differ substantially from conventional language. As such, the findings in this article may underestimate the potential performance benefits of prompting strategies, since these prompts were not optimized.

It's also worth noting that while LLMs led to better performance in the present study, researchers need to evaluate performance *in context*. For applied work, a model which classifies 99% of texts correctly might be useless if misclassification costs are sufficiently large (Elkan 2001; Hand 2012). Accordingly, future work should consider comparing methods by incorporating classification costs into performance metrics, using approaches like weighting (Zadrozny, Langford, and Abe 2003). We also found in the present study that human-coded stance scores had high inter-rater reliability. Accordingly, additional work is needed to determine how models perform in settings where more disagreement occurs across annotators, which could impact method choices. We also did not systematically explore how models performed on more sarcastic texts within each dataset, which could be a useful extension to consider comparing method performance.

We found that for the most recent model architecture we considered, GPT-4 Omni, tuning led to only slight performance gains and in several instances, improvements were non-significant compared to prompting strategies. These results suggest that there may be contexts where fine tuning is not needed to achieve reliable performance. However, additional research is needed to determine the exact data settings and models where tuning may not be needed to achieve consistent performance.

Future work on stance detection should also prioritize collecting training data and estimating stance scores on a continuous scale. Our findings indicate that methods which performed best using continuous measures also performed well when measures were binary or categorical. This suggests that benchmarking stance models using continuous scores provides a more comprehensive assessment of performance. Moreover, continuous measures offer inherent flexibility, enabling binary or categorical transformations as needed, which is not possible when data is collected using alternative measures. Estimating stance continuously could also enhance downstream analyses, facilitate better comparisons across stratified groups, and capture variation in stance magnitude that may otherwise be overlooked.

7.4. Recommendations

We find that tuned LLMs consistently outperform lexical and SLM approaches in the critical task of stance detection, with the most significant advantages observed in texts containing multiple target subjects. Prompt engineering strategies, while nearly as effective as tuned models, exhibit variable performance depending on the target subject, highlighting the need for cautious application.

We recommend that researchers use tuned LLMs for stance detection tasks and adopt continuous stance measures to enable richer analysis. However, due to differences in performance across target subjects and methods, we strongly encourage analysts to validate results against human-coded benchmarks and test findings across multiple stance detection approaches to ensure robustness in findings.

Acknowledgements. We extend our gratitude to Eli Kohlenberg, Sascha Ishikawa, and J. C. Bartle for their assistance in accessing the databases and APIs used in this study.

Funding Statement. This research was supported by the National Science Foundation (IIS/BIGDATA) Award no. 1837959.

Data Availability Statement. Instructions for replication are available on GitHub at https://github.com/maxgriswold/Stay-Tuned—Improving-Sentiment-Analysis-and-Stance-Detection-Using-Large-Language-Models. Replication code and datasets for this article have also been published in the Political Analysis Harvard Dataverse at https://doi.org/10.7910/DVN/KHNBZL (Griswold, Robbins, and Pollard 2025).

Supplementary Material. The supplementary material for this article can be found at https://doi.org/10.1017/pan.2025.10023.

References

Aiyappa, R., J. An, H. Kwak, and Y.-Y. Ahn. 2024. “Can We Trust the Evaluation on ChatGPT?.” Preprint, arXiv:2303.12767. https://doi.org/10.48550/arXiv.2303.12767.

AlDayel, A., and W. Magdy. 2021. “Stance Detection on Social Media: State of the Art and Trends.” *Information Processing & Management* 58 (4): 102597.

Alizadeh, M., et al. 2025. “Open-Source LLMs for Text Annotation: A Practical Guide for Model Setting and Fine-Tuning.” *Journal of Computational Social Science* 8 (1): 17. https://doi.org/10.48550/arXiv.2307.02179.

Alonso, M. A., D. Vilares, C. Gómez-Rodríguez, and J. Vilares. 2021. “Sentiment Analysis for Fake News Detection.” *Electronics* 10 (11): 1348. https://doi.org/10.3390/electronics10111348.

Altman, D. G., and P. Royston. 2006. “The Cost of Dichotomising Continuous Variables, 1080.” *BMJ* 332 (7549): 1080.

Alturayeif, N., H. Luqman, and M. Ahmed. 2023. “A Systematic Review of Machine Learning Techniques for Stance Detection and its Applications.” *Neural Computing and Applications* 35 (7): 5113–5144.

Baccianella, S., A. Esuli, and F. Sebastiani. 2010. “Sentiwordnet 3.0: An Enhanced Lexical Resource for Sentiment Analysis and Opinion Mining.” In *International Conference on Language Resources and Evaluation (LREC)*, vol. 10, 2200–2204. Valletta.

Baldi, P., S. Brunak, Y. Chauvin, C. A. F. Andersen, and H. Nielsen. 2000. “Assessing the Accuracy of Prediction Algorithms for Classification: An Overview.” *Bioinformatics* 16 (5): 412–424.

Beigi, G., X. Hu, R. Maciejewski, and H. Liu. 2016. “An Overview of Sentiment Analysis in Social Media and its Applications in Disaster Relief.” In *Sentiment Analysis and Ontology Engineering: An Environment of Computational Intelligence*, Studies in Computational Intelligence, edited by W. Pedrycz and S.-M. Chen, 313–340. Cham: Springer. https://doi.org/10.1007/978-3-319-30319-2˙13.

Bergam, N., E. Allaway, and K. Mckeown. 2022. “Legal and Political Stance Detection of Scotus Language.” Preprint, arXiv:2211.11724.

Bestvater, S. E., and B. L. Monroe. 2023. “Sentiment Is Not Stance: Target-Aware Opinion Classification for Political Text Analysis.” *Political Analysis* 31 (2): 235–256. https://doi.org/10.1017/pan.2022.10.

Boukes, M., B. van de Velde, T. Araujo, and R. Vliegenthart. 2020. “What’s the Tone? Easy Doesn’t Do it: Analyzing Performance and Agreement between off-the-Shelf Sentiment Analysis Tools.” *Communication Methods and Measures* 14 (2): 83–104. https://doi.org/10.1080/19312458.2019.1671966.

Brown, T., et al. 2020. “Language Models Are Few-Shot Learners.” *Advances in Neural Information Processing Systems* 33: 1877–1901.

Burnham, M. 2024. “Stance Detection: A Practical Guide to Classifying Political Beliefs in Text.” *Political Science Research and Methods*: 1–18. https://doi.org/10.48550/arXiv.2305.01723.

Cambria, E., S. Poria, R. Bajpai, and B. Schuller. 2016. “SenticNet 4: A Semantic Resource for Sentiment Analysis Based on Conceptual Primitives.” In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, 2666–77.

Carroll, R., J. B. Lewis, J. Lo, K. T. Poole, and H. Rosenthal. 2009. “Measuring Bias and Uncertainty in DW-NOMINATE Ideal Point Estimates Via the Parametric Bootstrap.” *Political Analysis* 17 (3): 261–275.

Ceron, A., L. Curini, S. M . Iacus, and G. Porro. 2014. "Every Tweet Counts? How Sentiment Analysis of Social Media Can Improve our Knowledge of Citizens' Political Preferences with an Application to Italy and France." *New Media & Society* 16 (2): 340–358. https://doi.org/10.1177/1461444813480466.

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: 1–13.

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. https://doi.org/10.1186/s13040-023-00322-4.

Cohen, J. 1983. "The Cost of Dichotomization." *Applied Psychological Measurement* 7 (3): 249–253.

Cramér, H. 1999. "Chapter 21: The Two-Dimensional Case." In *Mathematical Methods of Statistics*, 260–320. Princeton University Press. Accessed January 20, 2025. http://www.jstor.org/stable/j.ctt1bpm9r4.10.

Cruickshank, I. J., and L. H. X. Ng. 2024. "Prompting and Fine-Tuning Open-Sourced Large Language Models for Stance Classification." Preprint, arXiv:2309.13734. https://doi.org/10.48550/arXiv.2309.13734.

Dai, W., W. Kong, T. Shang, J. Feng, W. Jiaji, and Q. Tan. 2025. "Guideline for Novel Fine-Grained Sentiment Annotation and Data Curation: A Case Study." *Expert Systems* 42 (4): e70022.

Deng, L., and J. Wiebe. 2015. "MPQA 3.0: An Entity/Event-Level Sentiment Corpus." In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, edited by R. Mihalcea, J. Chai, and A. Sarkar, 1323–1328. Denver, CO: Association for Computational Linguistics. https://doi.org/10.3115/v1/N15-1146.

Deng, M., et al. 2022. "Rlprompt: Optimizing Discrete Text Prompts with Reinforcement Learning." Preprint, arXiv:2205.12548.

Devlin, J., M.-W. Chang, K. Lee, and K. Toutanova. 2019. "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding." In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies,* volume 1 (long and short papers), 4171–86. https://doi.org/10.48550/arXiv.1810.04805.

Ding, D., et al. 2024. "Cross-Target Stance Detection by Exploiting Target Analytical Perspectives." In *ICASSP 2024—2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 10651–10655. https://doi.org/10.1109/ICASSP48485.2024.10448397.

Elkan, C. 2001. "The Foundations of Cost-Sensitive Learning." In *International Joint Conference on Artificial Intelligence*, 17, 973–978. Lawrence Erlbaum Associates Ltd. https://cseweb.ucsd.edu/~elkan/rescale.pdf.

Gao, T., A. Fisch, and D. Chen. 2021. "Making Pre-trained Language Models Better Few-shot Learners." Preprint, arXiv:2012.15723. https://doi.org/10.48550/arXiv.2012.15723.

Gover, L. 2023. "Political Bias in Large Language Models." *The Commons, The Commons: Puget Sound Journal of Politics*. 4 (1): 11–22. University of Puget Sound. Accessed December 11, 2024. https://jstor.org/stable/community.36741697.

Greaves, F., D. Ramirez-Cano, C. Millett, A. Darzi, and L. Donaldson. 2013. "Use of Sentiment Analysis for Capturing Patient Experience from Free-Text Comments Posted Online." *Journal of Medical Internet Research* 15 (11): e2721. https://doi.org/10.2196/jmir.2721.

Grimmer, J., and B. M. Stewart. 2013. "Text as Data: The Promise and Pitfalls of Automatic Content Analysis Methods for Political Texts." *Political Analysis* 21 (3): 267–297.

Griswold, M., M. Robbins, and M. Pollard. 2025. "Replication Data for: "Stay Tuned—Improving Sentiment Analysis and Stance Detection Using Large Language Models." https://doi.org/10.7910/DVN/KHNBZL.

Gül, İ., R. Lebret, and K. Aberer. 2024. "Stance Detection on Social Media with Fine-Tuned Large Language Models." Preprint, arXiv:2404.12171. http://arxiv.org/abs/2404.12171.

Hand, D. J. 2012. "Assessing the Performance of Classification Methods." *International Statistical Review* 80 (3): 400–414.

Hardalov, M., A. Arora, P. Nakov, and I. Augenstein. 2022. "A Survey on Stance Detection for Mis- and Disinformation Identification." Preprint, arXiv:2103.00242 [cs]. https://doi.org/10.48550/arXiv.2103.00242.

Hartmann, J., M. Heitmann, C. Siebert, and C. Schamp. 2023. "More than a Feeling: Accuracy and Application of Sentiment Analysis." *International Journal of Research in Marketing* 40 (1): 75–87. https://doi.org/10.1016/j.ijresmar.2022.05.005.

He, P., X. Liu, J. Gao, and W. Chen. 2020. "*Deberta: Decoding-Enhanced Bert with Disentangled Attention*." Preprint, arXiv:2006.03654.

He, X., et al. 2024. "AnnoLLM: Making Large Language Models to Be Better Crowdsourced Annotators." Preprint, arXiv:2303.16854. https://doi.org/10.48550/arXiv.2303.16854.

Hu, M., and B. Liu. 2004. "Mining and Summarizing Customer Reviews." In *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 168–177. Seattle WA: ACM. https://doi.org/10.1145/1014052.1014073.

Hutto, C., and E. Gilbert. 2014. "VADER: A Parsimonious Rule-Based Model for Sentiment Analysis of Social Media Text." *Proceedings of the International AAAI Conference on Web and Social Media* 8 (1): 216–225. https://doi.org/10.1609/icwsm.v8i1.14550.

Kawintiranon, K., and L. Singh. 2021. "Knowledge Enhanced Masked Language Model for Stance Detection." In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics:*

Human Language Technologies, edited by K. Toutanova, et al., 4725–4735. Association for Computational Linguistics. https://doi.org/10.18653/v1/2021.naacl-main.376.

Khiabani, P. J., and A. Zubiaga. 2025. "Cross-Target Stance Detection: A Survey of Techniques, Datasets, and Challenges." *Expert Systems with Applications* 2025: 127790.

Kojima, T., S. S. Gu, M. Reid, Y. Matsuo, and Y. Iwasawa. 2022. "Large Language Models Are Zero-Shot Reasoners." *Advances in Neural Information Processing Systems* 35: 22199–213. https://doi.org/10.48550/arXiv.2205.11916.

Kristensen-McLachlan, R. D., M. Canavan, M. Kardos, M. Jacobsen, and L. Aarøe. 2023. "Chatbots Are Not Reliable Text Annotators." Preprint, arXiv:2311.05769. https://doi.org/10.48550/arXiv.2311.05769.

Lan, X., C. Gao, D. Jin, and Y. Li. 2024. "Stance Detection with Collaborative Role-Infused Llm-Based Agents." In *Proceedings of the International AAAI Conference on Web and Social Media*, 18, 891–903.

Laurer, M., W. van Atteveldt, A. Casas, and K. Welbers. 2024a. "Building Efficient Universal Classifiers with Natural Language Inference." Preprint, arXiv:2312.17543 [cs]. https://doi.org/10.48550/arXiv.2312.17543.

Laurer, M., W. van Atteveldt, A. Casas, and K. Welbers. 2024b. "Less Annotating, More Classifying: Addressing the Data Scarcity Issue of Supervised Machine Learning with Deep Transfer Learning and BERT-NLI." *Political Analysis* 32 (1): 84–100. https://doi.org/10.1017/pan.2023.20.

Li, M., and F. Conrad. 2024. "Advancing Annotation of Stance in Social Media Posts: A Comparative Analysis of Large Language Models and Crowd Sourcing." Preprint, arXiv:2406.07483. Accessed October 23, 2024. http://arxiv.org/abs/2406.07483.

Li, Y., and C. Caragea. 2021. "A Multi-Task Learning Framework for Multi-Target Stance Detection." In *Findings of the Association for Computational Linguistics: ACL-IJCNLP*, 2021, 2320–2326.

Li, Y., T. Sosea, A. Sawant, A. J. Nair, D. Inkpen, and C. Caragea. 2021. "P-Stance: A Large Dataset for Stance Detection in Political Domain." In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, edited by C. Zong, F. Xia, W. Li, and R. Navigli, 2355–2365. Association for Computational Linguistics. https://doi.org/10.18653/v1/2021.findings-acl.208.

Liu, Y., et al. 2019. "RoBERTa: A Robustly Optimized BERT Pretraining Approach." Preprint, arXiv:1907.11692 [cs]. https://doi.org/10.48550/arXiv.1907.11692.

Loureiro, D., F. Barbieri, L. Neves, L. E. Anke, and J. Camacho-Collados. 2022. "Timelms: diachronic language models from twitter." Preprint, arXiv:2202.03829 [cs.CL]. https://arxiv.org/abs/2202.03829.

Maia, D., and N. F. F. da Silva. 2024. "Enhancing Stance Detection in Low-Resource Brazilian Portuguese Using Corpus Expansion Generated by Gpt-3.5." In *Proceedings of the 16th International Conference on Computational Processing of Portuguese*, 503–508.

Matthews, B. W. 1975. "Comparison of the Predicted and Observed Secondary Structure of t4 Phage Lysozyme. *Biochimica et Biophysica Acta (BBA)-Protein*." *Structure* 405 (2): 442–451.

Le Mens, G., and A. Gallego. 2025. "Positioning Political Texts with Large Language Models by Asking and Averaging." *Political Analysis* 33 (3): 274–82. https://doi.org/10.48550/arXiv.2311.16639.

Mets, M., A. Karjus, I. Ibrus, and M. Schich. 2024. "Automated Stance Detection in Complex Topics and Small Languages: The Challenging Case of Immigration in Polarizing News Media." *PLoS One* 19 (4): e0302380. https://doi.org/10.1371/journal.pone.0302380.

Mohammad, S., F. Bravo-Marquez, M. Salameh, and S. Kiritchenko. 2018. "Semeval-2018 Task 1: Affect in Tweets." In *Proceedings of the 12th International Workshop on Semantic Evaluation*, 1–17.

Mohammad, S., and P. Turney. 2010. "Emotions Evoked by Common Words and Phrases: Using Mechanical Turk to Create an Emotion Lexicon." In *Proceedings of the NAACL HLT 2010 Workshop on Computational Approaches to Analysis and Generation of Emotion in Text*, edited by D. Inkpen and C. Strapparava, 26–34. Los Angeles, CA: Association for Computational Linguistics.

Mohammad, S. M., and F. Bravo-Marquez. 2017. "Wassa-2017 Shared Task on Emotion Intensity." Preprint, arXiv:1708.03700.

Mohammad, S. M., P. Sobhani, and S. Kiritchenko. 2017. "Stance and Sentiment in Tweets." *ACM Transactions on Internet Technology (TOIT)* 17 (3): 1–23.

Motoki, F., V. P. Neto, and V. Rodrigues. 2024. "More Human than Human: Measuring ChatGPT Political Bias." *Public Choice* 198 (1): 3–23. https://doi.org/10.1007/s11127-023-01097-2.

Mu, Y., et al. 2024. "Navigating Prompt Complexity for Zero-Shot Classification: A Study of Large Language Models in Computational Social Science." Preprint, arXiv:2305.14310. https://doi.org/10.48550/arXiv.2305.14310.

Nasukawa, T., and J. Yi. 2003. "Sentiment Analysis: Capturing Favorability Using Natural Language Processing." In *Proceedings of the 2nd International Conference on Knowledge Capture K-CAP '03*, 70–77. New York, NY: Association for Computing Machinery. https://doi.org/10.1145/945645.945658.

Ng, L. H. X., and K. M. Carley. 2022. "Is my Stance the Same as your Stance? A Cross Validation Study of Stance Detection Datasets." *Information Processing & Management* 59 (6): 103070. https://doi.org/10.1016/j.ipm.2022.103070.

Ng, L. H. X., I. Cruickshank, and R. K.-W. Lee. 2024. "Examining the Influence of Political Bias on Large Language Model Performance in Stance Classification." Preprint, arXiv:2407.17688. https://doi.org/10.48550/arXiv.2407.17688.

OpenAI. 2022. "*Introducing ChatGPT*." Accessed January 12, 2024. https://openai.com/blog/chatgpt.

Osnabrügge, M., E. Ash, and M. Morelli. 2023. "Cross-Domain Topic Classification for Political Texts." *Political Analysis* 31 (1): 59–80. https://doi.org/10.1017/pan.2021.37.

Pang, B., L. Lee, and S. Vaithyanathan. 2002. "*Thumbs up? Sentiment Classification using Machine Learning Techniques.*" Preprint, arXiv:cs/0205070. https://doi.org/10.48550/arXiv.cs/0205070.

Perez, E., D. Kiela, and K. Cho. 2021. "True Few-Shot Learning with Language Models." *Advances in Neural Information Processing Systems* 34: 11054–70. https://doi.org/10.48550/arXiv.2105.11447.

Poole, K. T., and H. Rosenthal. 1985. "A Spatial Model for Legislative Roll Call Analysis." *American Journal of Political Science* 357–384.

Radford, A., K. Narasimhan, T. Salimans, and I. Sutskever. 2018. "Improving Language Understanding by Generative Pre-Training."

Rozado, D. 2024. "The Political Preferences of LLMs." *PloS one* 19 (7): e0306621. https://doi.org/10.48550/arXiv.2402.01789.

Saif, H., M. Fernandez, Y. He, and H. Alani. 2013. "Evaluation Datasets for Twitter Sentiment Analysis." *Emotion and Sentiment in Social and Expressive Media*: 9.

Sanh, V., L. Debut, J. Chaumond, and T. Wolf. 2019. "Distilbert, a Distilled Version of Bert: Smaller, Faster, Cheaper and Lighter." Preprint, arXiv:1910.01108.

Schiller, B., J. Daxenberger, and I. Gurevych. 2021. "Stance Detection Benchmark: How Robust Is your Stance Detection?" *KI-K ü nstliche Intelligenz* 35 (3): 329–41.

Sheskin, D. 2003. *Handbook of Parametric and Nonparametric Statistical Procedures*. Chapman and Hall.

Sobhani, P., D. Inkpen, and X. Zhu. 2019. "Exploring Deep Neural Networks for Multitarget Stance Detection." *Computational Intelligence* 35 (1): 82–97.

Taboada, M., J. Brooke, M. Tofiloski, K. Voll, and M. Stede. 2011. "Lexicon-Based Methods for Sentiment Analysis." *Computational Linguistics* 37 (2): 267–307. https://doi.org/10.1162/COLI˙a˙00049.

Thelwall, M., K. Buckley, and G. Paltoglou. 2012. "Sentiment Strength Detection for the Social Web." *Journal of the American Society for Information Science and Technology* 63 (1): 163–173.

Törnberg, P. 2023. "ChatGPT-4 Outperforms Experts and Crowd Workers in Annotating Political Twitter Messages with Zero-Shot Learning." Preprint, arXiv:2304.06588. https://doi.org/10.48550/arXiv.2304.06588.

Vamvas, J., and R. Sennrich. 2020. "X-Stance: A Multilingual Multi-Target Dataset for Stance Detection." Preprint, arXiv:2003.08385.

Vaswani, A., et al. 2017. "Attention Is all you Need." *Advances in Neural Information Processing Systems*, 30. https://proceedings.neurips.cc/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

Wankhade, M., A. C. S. Rao, and C. Kulkarni. 2022. "A Survey on Sentiment Analysis Methods, Applications, and Challenges." *Artificial Intelligence Review* 55 (7): 5731–5780. https://doi.org/10.1007/s10462-022-10144-1.

Wei, J., et al. 2022. "Chain-of-Thought Prompting Elicits Reasoning in Large Language Models." *Advances in Neural Information Processing Systems* 35: 24824–37. https://doi.org/10.48550/arXiv.2201.11903.

Xu, C., C. Paris, S. Nepal, and R. Sparks. 2018. "Cross-Target Stance Classification with Self-Attention Networks." In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, edited by I. Gurevych and Y. Miyao, 778–783. Melbourne: Association for Computational Linguistics. https://doi.org/10.18653/v1/P18-2123.

Yarchi, M., C. Baden, and N. Kligler-Vilenchik. 2021. "Political Polarization on the Digital Sphere: A Cross-Platform, Over-Time Analysis of Interactional, Positional, and Affective Polarization on Social Media." *Political Communication* 38 (1–2): 98–139. https://doi.org/10.1080/10584609.2020.1785067.

Zadrozny, B., J. Langford, and N. Abe. 2003. "Cost-Sensitive Learning by Cost-Proportionate Example Weighting." In *Third IEEE International Conference on Data Mining*, 435–442. IEEE.

Zhang, B., G. Dai, F. Niu, N. Yin, X. Fan, and H. Huang. 2024. "A Survey of Stance Detection on Social Media: New Directions and Perspectives." Preprint, arXiv:2409.15690. http://arxiv.org/abs/2409.15690.

Zhang, B., et al. 2024. "Investigating Chain-of-thought with ChatGPT for Stance Detection on Social Media." Preprint, arXiv:2304.03087. https://doi.org/10.48550/arXiv.2304.03087.

Zhang, W., Y. Deng, B. Liu, S. Pan, and L. Bing. 2024. "Sentiment Analysis in the Era of Large Language Models: A Reality Check." In *Findings of the Association for Computational Linguistics: NAACL 2024*, edited by K. Duh, H. Gomez, and S. Bethard, 3881–3906. Mexico City: Association for Computational Linguistics. https://doi.org/10.18653/v1/2024.findings-naacl.246.

Zhang, Y., and Y. Yang. 2015. "Cross-Validation for Selecting a Model Selection Procedure." *Journal of Econometrics* 187 (1): 95–112.

Zhao, Z., E. Wallace, S. Feng, D. Klein, and S. Singh. 2021. "Calibrate before Use: Improving Few-Shot Performance of Language Models." In *Proceedings of the 38th International Conference on Machine Learning*, 12697–12706. PMLR, Accessed November 20, 2024. https://proceedings.mlr.press/v139/zhao21c.html.

Zhu, Y., P. Zhang, E.-U. Haq, H. Pan, and G. Tyson. 2023. "Can ChatGPT Reproduce Human-Generated Labels? A Study of Social Computing Tasks." Preprint, arXiv:2304.10145. Accessed November 20, 2024. http://arxiv.org/abs/2304.10145.

Cite this article: Griswold, M., Robbins, M. W. and Pollard, M. S. (2025). Stay Tuned: Improving Sentiment Analysis and Stance Detection Using Large Language Models. *Political Analysis*, 1–20. https://doi.org/10.1017/pan.2025.10023