Stance Detection with Fine-Tuned Large Language Models

Anonymous ACL submission

Abstract

Stance detection, a key task in natural language processing, determines an author's viewpoint based on textual analysis. This study examines the evolution of stance detection methods, transitioning from early machine learning approaches to the groundbreaking BERT model, and eventually to modern Large Language Models (LLMs) such as ChatGPT. While ChatGPT's closed-source nature and associated costs present challenges, the open-source model LLaMa-2 offers an encouraging alternative. We fine-tuned both ChatGPT and LLaMa-2 on two publicly available datasets: SemEval-2016 and P-Stance. Results highlight the efficacy of fine-tuned LLMs in stance detection, with both models surpassing previous benchmarks. LLaMa-2's performance, despite having fewer parameters than ChatGPT, underscores the efficiency of open-source models. This study emphasizes the potential of LLMs in stance detection and calls for more extensive research in this field. To further contribute to the research community, our code for this study will be made publicly available.

1 Introduction

Stance detection seeks to determine an author's viewpoint—whether supportive, oppositional, or neutral—on a variety of subjects ranging from opinions on political figures to views on pressing environmental policies, based on textual analysis (Hasan and Ng, 2013; Küçük and Can, 2020; Al-Dayel and Magdy, 2021). Given the proliferation of content on social media platforms like X, formerly Twitter, the task of extracting and accurately parsing underlying stances has become paramount (Siddiqua et al., 2019). Interpreting these perspectives not only offers a window into society's collective opinions but also facilitates better insights into societal shifts, directly benefiting areas such as data extraction and policy formulation(Darwish et al., 2017; Glandt et al., 2021). As natural language processing (NLP) and social computing continue to grow and overlap, advancements in these fields allow researchers to improve models, leading to better results in extracting stances from given texts.

Stance detection in textual data began with a heavy emphasis on rule-based and traditional machine learning approaches, with support vector machines (SVM) standing out as an early benchmark (Anand et al., 2011; Walker et al., 2012; Mohammad et al., 2016). Over time, deep learning models started playing a pivotal role in stance detection (Wei et al., 2016; Zarrella and Marsh, 2016). Despite initial challenges, these models, through continuous refinement and innovative strategies, began to outperform the traditional rule-based and machine learning methods (Dey et al., 2018; Huang et al., 2018; Zhang et al., 2019a). The introduction of pretrained language models, particularly BERT (Devlin et al., 2019), marked a significant advancement. A significant shift in stance detection came with Google's BERT model (Devlin et al., 2019). BERT showcased the potential of large pre-trained language models (PLM) in stance detection by employing bidirectional encoders and fine-tuning on vast datasets (Li et al., 2021). This approach not only raised the bar for many NLP tasks but also improved the precision and depth of stance detection models (Allaway and McKeown, 2020; Shin et al., 2020; Wei et al., 2022).

The capabilities of Large Language Models (LLMs) have significantly advanced, enabling marked improvements in NLP (Brown et al., 2020). Trained on large datasets, these models have refined their ability to understand and mimic human language patterns (Wei et al., 2023). With this enhanced capability, LLMs differ from BERT in their approach; while BERT often requires fine-tuning on specific tasks, LLMs, through the use of prompting techniques, can make predictions without the need for fine-tuning. This allows them to become more proficient in accurately detecting stances and understanding the relationship between the target and the text in alignment with the author's viewpoint. ChatGPT[1] and ChatGPT Plus[2] by OpenAI are models that have gained significant attention in the field (OpenAI, 2023).

Much of the recent research on LLMs, particularly ChatGPT, frequently employs zero-shot and, in certain studies, few-shot prompt engineering techniques. Notably, studies like such as (Aiyappa et al., 2023; Chen et al., 2023) have underscored ChatGPT's accuracy and consistency in stance detection. Given that ChatGPT is not open-source and considering the initial guidelines set by OpenAI, these methodologies became the primary approach for many researchers in the field.

The recent introduction of fine-tuning capabilities by OpenAI[3] presents a potential improvement for model performance in stance detection. While ChatGPT exhibits significant potential, its closed-source design poses challenges. Accessing its fine-tuning features necessitates the use of the API, incurring associated costs. For researchers with budgetary constraints, these financial considerations, combined with the model's restricted accessibility, pose significant barriers. In light of these challenges, and given the notable attention LLaMa-2[4], an open-source model, has received since its release by Meta AI (Touvron et al., 2023), we incorporate it into our study alongside ChatGPT.

In this paper, we want to determine whether fine-tuned LLMs, specifically ChatGPT and LLaMa-2, could outperform previous stance detection benchmarks. Additionally, we aimed to compare the post-fine-tuning performance of these two models to provide insights for ongoing and future research.

2 Methods

2.1 Datasets and Evaluation Metrics

Datasets. To assess the performance of our fine-tuned LLMs, we employed two publicly available datasets. The SemEval-2016 Dataset (Mohammad et al., 2016) addresses several targets that include political figures and broader societal concerns. These targets are categorized into three stances: Favor, Against, and None. The specific targets in the dataset are Atheism (A), Climate Change is a Real Concern (CC), Donald Trump (DT), Feminist Movement (FM), Hillary Clinton (HC), and Legalization of Abortion (LA). The P-Stance Dataset (Li et al., 2021), on the other hand, narrows its focus to the political domain and classifies stances as either Favor or Against. The specific political figures targeted in this dataset are Bernie Sanders, Donald Trump, and Joe Biden.

Evaluation Metrics. In line with the standards set by previous studies (Mohammad et al., 2016, 2017), we adopt F_{avg} as our primary evaluation metric. This metric, F_{avg}, computes the average of the $F1$ scores for the 'favor' and 'against' classes.

2.2 Models

For the fine-tuning of the `ChatGPT` model, which comprises 175 billion parameters, we followed the guidelines provided on the official OpenAI website[5]. After the fine-tuning process, the resulting model is referred to as `ChatGPT-ft`. Notably, the only adjustable hyperparameter available during the fine-tuning process was the number of epochs, which we set to three for our experiments.

For the fine-tuning of the `LLaMa-2` models, specifically `LLaMa-2-7b` representing the version with 7 billion parameters and `LLaMa-2-13b` denoting the one with 13 billion parameters, we adjusted our approach based on the dataset in question: three epochs for SemEval-2016 and one epoch for the P-Stance dataset[6]. Post fine-tuning, the resulting models are labeled as `LLaMa-2-7b-ft` and `LLaMa-2-13b-ft`. For both the SemEval-2016 and P-Stance datasets, we employed the parameter-efficient fine-tuning method with Low-Rank Adaptation (LoRA) using the `Lit-GPT`[7] framework. The specific methodological and hyperparameter details for the fine-tuning process of the `LLaMa-2` models have been included in the Appendix A.

For comparative analysis against the fine-tuned models, we performed zero-shot stance detection using the models: `ChatGPT`, `LLaMa-2-7b-chat`, and `LLaMa-2-13b-chat`.

2.3 Prompting Details

For the ChatGPT model, we employed specific prompting methods for each dataset. For the LLaMa-2 model, our prompting strategy was inspired by the template samples available in HuggingFace's resources.[8] Detailed specifications

[1] https://openai.com/blog/chatgpt

[2] https://openai.com/blog/chatgpt-plus

[3] https://openai.com/blog/gpt-3-5-turbo-fine-tuning-and-api-updates

[4] https://ai.meta.com/llama/

[5] https://platform.openai.com/docs/guides/fine-tuning

[6] The adjustment to one epoch for fine-tuning P-Stance was due to its larger training set size compared to SemEval-2016, minimizing overfitting concerns.

[7] https://github.com/Lightning-AI/lit-gpt

[8] https://huggingface.co/blog/llama2

of the prompts used for each dataset can be found in the Appendix B.

2.4 Baselines

We have selected various stance detection models as our baselines, categorizing them based on their foundational architectures and approaches. From the category of recurrent neural networks (RNN), our choices include the BiLSTM (Augenstein et al., 2016) and BiCond (Augenstein et al., 2016) models, both of which deploy bidirectional LSTM layers for processing. MemNet (Tang et al., 2016) serves as a representative of memory networks, with a primary focus on aspect-level sentiment analysis. Both AoA (Huang et al., 2018) and TAN (Du et al., 2017) employ attention mechanisms, enabling them to effectively weigh different segments of the input text for stance detection purposes. ASGCN (Zhang et al., 2019b) integrates graph-based methodologies for capturing dependencies in text, while AT-JSS-Lex (Li and Caragea, 2019) stands out as a multi-task model, merging sentiment and stance detection while also incorporating a lexicon. On another front, TPDG (Liang et al., 2021) delves into target-centric methodologies, and StSQA (Chen et al., 2023) employs a novel method, teaching ChatGPT stance detection by using a 1-shot example.

3 Results

3.1 Zero-shot vs. Fine-Tuning

In Tables 1 and 2, we present the performance scores of LLMs, `ChatGPT` and `Llama`, in a zero-shot setting. Although these models exhibit impressive zero-shot performance, our evaluations highlight that their true potential is unlocked post fine-tuning. Notably, the zero-shot evaluations on the SemEval-2016 and P-Stance datasets utilized the same prompts as those used during the fine-tuning phase.

Within the SemEval-2016 dataset, `ChatGPT`'s zero-shot capability stood out as superior compared to both `Llama` models. A parallel trend is observed in the P-Stance dataset, where `ChatGPT` similarly outperformed its counterparts in a zero-shot setting.

A notable difference emerged in prediction times. Predictions using the zero-shot approach, specifically with `LLaMa-2-7b-chat`, took about 39 minutes for the SemEval-2016 test set, while its fine-tuned counterpart completed in just 2 minutes. The extended runtime of zero-shot models stems from their generation of full answer sentences, in contrast to the fine-tuned models which are optimized to output just a single token indicating the stance.

The observed differences in performance between `ChatGPT` and the `LLaMa-2` models can be partly attributed to the Reinforcement Learning from Human Feedback (RLHF) employed by `ChatGPT`[9]. This training strategy, which is absent in the `LLaMa-2` models, incorporates feedback loops with human input. This could provide `ChatGPT` with insights into the training data we're using,

Models	**FM**	**HC**	**LA**
BiLSTM	52.2	57.4	54.0
BiCond	61.4	59.8	54.5
MemNet	57.8	60.3	61.0
TAN	58.3	67.7	65.7
AoA	60.0	58.2	62.4
ASGCN	58.5	64.3	62.9
AT-JSS-Lex	61.5	68.3	68.4
TPDG	67.3	73.4	74.7
		Zero-shot	
ChatGPT	74.6	82.8	59.6
LLaMa-2-7b-chat	51.6	63.9	49.2
LLaMa-2-13b-chat	55.0	61.5	45.9
		Fine-tuned	
ChatGPT-ft	**79.7**	83.4	**72.6**
LLaMa-2-7b-ft	73.3	84.2	71.2
LLaMa-2-13b-ft	76.0	**84.8**	72.5

Table 1: SemEval-2016 Dataset performance comparison (using F_{avg} scores)

Models	**Bernie**	**Biden**	**Trump**
BiLSTM	63.9	69.5	72.0
BiCond	64.6	69.4	73.0
MemNet	72.8	77.6	77.7
TAN	72.0	77.9	77.5
AoA	71.7	77.8	77.7
ASGCN	70.8	78.4	77.0
StSQA	80.8	82.6	85.7
		Zero-shot	
ChatGPT	75.2	82.6	73.7
LLaMa-2-7b-chat	48.3	52.9	43.6
LLaMa-2-13b-chat	49.8	53.7	45.3
		Fine-tuned	
ChatGPT-ft	**81.8**	**89.7**	**91.9**
LLaMa-2-7b-ft	79.0	87.2	89.8
LLaMa-2-13b-ft	81.0	89.0	88.9

Table 2: P-Stance Dataset performance comparison (using F_{avg} scores)

[9]https://openai.com/blog/chatgpt

potentially leading to domain-specific contamination and explaining its stronger performance in a zero-shot setting. However, this advantage diminishes when both models are fine-tuned.

Comparing zero-shot and fine-tuned results as presented in Tables 1 and 2, `ChatGPT`, which stood out in its zero-shot evaluations, exhibited even more impressive results after fine-tuning. Conversely, the `LLaMa-2` models, which started with lower performance scores in the zero-shot setting, demonstrated substantial improvements with fine-tuning. This highlights that while task-specific tuning is beneficial for both models, `ChatGPT`'s initial lead might be influenced by its RLHF training, potentially exposing it to targets available in the datasets.

This pattern of improvement across both datasets underscores the pivotal role of fine-tuning. While LLMs inherently possess strong generalization abilities, adapting them to specific tasks through fine-tuning is essential. This adaptation through fine-tuning not only enhances their performance but also ensures LLMs reach their full potential in specific tasks.

3.2 Fine-Tuned Models vs. Baselines

In the results presented in Table 1, we can observe the prominence of the `ChatGPT-ft` model across all targets. It becomes clear that its performance is above the average when compared to other models in the table. Moreover, the other fine-tuned LLMs, `LLaMa-2-7b-ft` and `LLaMa-2-13b-ft`, also consistently delivered good results. The difference in performance underscores the unique strengths of LLMs, especially when fine-tuned for specific tasks.

Transitioning to Table 2, the stance prediction performance across different political figures is presented. Again, `ChatGPT-ft` stands out, but it's closely followed by the `LLaMa-2` models. The difference between these fine-tuned LLMs and the rest is evident and substantial. Such a distinction in scores not only emphasizes the superiority of the fine-tuned models but also raises questions about how other models could be improved.

For a more detailed analysis of the SemEval-2016 results, please refer to Appendix C.

4 Discussion

Our experiments with the SemEval-2016 and P-Stance 2021 datasets highlight the effectiveness of fine-tuned LLMs in stance detection. Specifically, the `ChatGPT-ft` model consistently outperformed other models in our tests, as shown in Tables 1 and 2. The LLaMa-2 models also performed notably well, further indicating the power of LLMs in this domain.

However, there were intriguing variations. Despite being larger, the `LLaMa-2-13b-ft` model didn't consistently outperform the smaller `LLaMa-2-7b-ft`. This suggests that model size alone doesn't determine success. Fine-tuning, dataset specifics, and architecture also play crucial roles.

Differences in performance across targets hint at these models being sensitive to specific domains. For instance, while `ChatGPT-ft` excelled in many categories, it faced challenges matching the performance of `LLaMa-2-13b-ft` in the Hillary Clinton domain. This variance might also be attributed to the datasets used during the initial pre-training of LLMs, which can introduce biases or domain knowledge that influence their subsequent fine-tuned performance. This shows that a model's general effectiveness can be influenced by topic-specific factors.

Compared to other models we evaluated, LLMs consistently stood out, highlighting their significant potential in modern NLP tasks. The evident differences in results indicate that both the data-intensive training and the size of LLMs could be crucial contributors to their enhanced performance. These findings open doors for further research, suggesting that refining LLM techniques and architectures could lead to even more advanced results.

In a broader context, the strong performance of LLMs in our study highlights their potential in real-world stance detection tasks, such as identifying the stance of news articles and analyzing public opinions on key societal issues.

5 Conclusion

In conclusion, our exploration of stance detection, particularly using ChatGPT and LLaMa-2, provides clear insights into the significant potential these models offer. Their superior performance, as demonstrated in our results, firmly establishes them as frontrunners in the domain. Understanding stance detection remains a multifaceted challenge, and while LLMs have made notable progress, their role in guiding the future trajectory of NLP is evident. As we anticipate further advancements, the evolution of LLMs and their broader applications will be of great interest. These developments signal a new era of refined and accurate NLP models, bringing significant benefits to the wider academic community.

Limitations

In conducting this research, several limitations pertaining to the use of ChatGPT were encountered. First and foremost, the exclusive nature of ChatGPT means that it is accessible solely via its designated API. This limited the extent of model adjustments, with the number of epochs during fine-tuning being the only modifiable hyperparameter at the time of our experimentation. Furthermore, financial considerations present an additional constraint. As per the current pricing structure, the cost for training ChatGPT stands at $0.008 per 1,000 tokens[10]. Fine-tuning a dataset with 100,000 tokens over three epochs is estimated to cost about $2.40 USD. To put this in perspective, the estimated cost for training the SemEval-2016 dataset was around $21.77 USD. Given such pricing, the act of fine-tuning becomes financially challenging without a substantial budget.

In the fine-tuning process of the Llama 2 models, we encountered certain limitations. We were able to successfully fine-tune the Llama 2 7b and Llama 2 13b models using the NVIDIA A100 GPU with 40GB. However, due to the more extensive structure of the Llama 2 70b model, we needed a more powerful GPU to fine-tune it. This emerged as a constraint that we couldn't overcome with our current resources.

In the SemEval-2016 dataset, a notable limitation was the training dataset size for the targets. In comparison, there was a more extensive training resource available for P-Stance. With more training data for each target in SemEval-2016, the LLMs could likely achieve better stance detection results.

Ethical Considerations

In the course of this research, it's crucial to acknowledge the potential limitations of Large Language Models. Both ChatGPT and Llama 2, like other LLMs, may produce inaccurate information about targets present in stance detection datasets. Such inaccuracies can emerge from various factors inherent to algorithmic predictions and inherent model limitations.

This research relied on publicly available datasets for the fine-tuning of LLMs. The primary goal in using these datasets was academic research. At no stage was there an intention to produce or support biased predictions. For transparency and further review, both the predictions made by the fine-tuned models and the code used in the research will be made publicly available.

[10] https://openai.com/pricing

References

Rachith Aiyappa, Jisun An, Haewoon Kwak, and Yong-yeol Ahn. 2023. Can we trust the evaluation on ChatGPT? In *Proceedings of the 3rd Workshop on Trustworthy Natural Language Processing (TrustNLP 2023)*, pages 47–54, Toronto, Canada. Association for Computational Linguistics.

Abeer AlDayel and Walid Magdy. 2021. Stance detection on social media: State of the art and trends. *Information Processing & Management*, 58(4):102597.

Emily Allaway and Kathleen McKeown. 2020. Zero-Shot Stance Detection: A Dataset and Model using Generalized Topic Representations. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8913–8931, Online. Association for Computational Linguistics.

Pranav Anand, Marilyn Walker, Rob Abbott, Jean E. Fox Tree, Robeson Bowmani, and Michael Minor. 2011. Cats rule and dogs drool!: Classifying stance in online debate. In *Proceedings of the 2nd Workshop on Computational Approaches to Subjectivity and Sentiment Analysis (WASSA 2.011)*, pages 1–9, Portland, Oregon. Association for Computational Linguistics.

Isabelle Augenstein, Tim Rocktäschel, Andreas Vlachos, and Kalina Bontcheva. 2016. Stance detection with bidirectional conditional encoding. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 876–885, Austin, Texas. Association for Computational Linguistics.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Jiuhai Chen, Lichang Chen, Heng Huang, and Tianyi Zhou. 2023. When do you need chain-of-thought prompting for chatgpt? *arXiv preprint arXiv:2304.03262*.

Kareem Darwish, Walid Magdy, and Tahar Zanouda. 2017. Trump vs. hillary: What went viral during the 2016 us presidential election. In *Social Informatics: 9th International Conference, SocInfo 2017, Oxford, UK, September 13-15, 2017, Proceedings, Part I 9*, pages 143–161. Springer.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina 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), pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Kuntal Dey, Ritvik Shrivastava, and Saroj Kaushik. 2018. Topical stance detection for twitter: A two-phase lstm model using attention. In *Advances in Information Retrieval: 40th European Conference on IR Research, ECIR 2018, Grenoble, France, March 26-29, 2018, Proceedings 40*, pages 529–536. Springer.

Jiachen Du, Ruifeng Xu, Yulan He, and Lin Gui. 2017. Stance classification with target-specific neural attention networks. International Joint Conferences on Artificial Intelligence.

Kyle Glandt, Sarthak Khanal, Yingjie Li, Doina Caragea, and Cornelia Caragea. 2021. Stance detection in COVID-19 tweets. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1596–1611, Online. Association for Computational Linguistics.

Kazi Saidul Hasan and Vincent Ng. 2013. Stance classification of ideological debates: Data, models, features, and constraints. In *Proceedings of the Sixth International Joint Conference on Natural Language Processing*, pages 1348–1356, Nagoya, Japan. Asian Federation of Natural Language Processing.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models.

Binxuan Huang, Yanglan Ou, and Kathleen M Carley. 2018. Aspect level sentiment classification with attention-over-attention neural networks. In *Social, Cultural, and Behavioral Modeling: 11th International Conference, SBP-BRiMS 2018, Washington, DC, USA, July 10-13, 2018, Proceedings 11*, pages 197–206. Springer.

Dilek Küçük and Fazli Can. 2020. Stance detection: A survey. *ACM Comput. Surv.*, 53(1).

Yingjie Li and Cornelia Caragea. 2019. Multi-task stance detection with sentiment and stance lexicons. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 6299–6305, Hong Kong, China. Association for Computational Linguistics.

Yingjie Li, Tiberiu Sosea, Aditya Sawant, Ajith Jayaraman Nair, Diana Inkpen, and Cornelia Caragea. 2021. P-stance: A large dataset for stance detection in political domain. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 2355–2365, Online. Association for Computational Linguistics.

Bin Liang, Yonghao Fu, Lin Gui, Min Yang, Jiachen Du, Yulan He, and Ruifeng Xu. 2021. Target-adaptive graph for cross-target stance detection. In *Proceedings of the Web Conference 2021*, pages 3453–3464.

Saif Mohammad, Svetlana Kiritchenko, Parinaz Sobhani, Xiaodan Zhu, and Colin Cherry. 2016. SemEval-2016 task 6: Detecting stance in tweets. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 31–41, San Diego, California. Association for Computational Linguistics.

Saif M Mohammad, Parinaz Sobhani, and Svetlana Kiritchenko. 2017. Stance and sentiment in tweets. *ACM Transactions on Internet Technology (TOIT)*, 17(3):1–23.

OpenAI. 2023. Gpt-4 technical report.

Taylor Shin, Yasaman Razeghi, Robert L. Logan IV, Eric Wallace, and Sameer Singh. 2020. AutoPrompt: Eliciting Knowledge from Language Models with Automatically Generated Prompts. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4222–4235, Online. Association for Computational Linguistics.

Umme Aymun Siddiqua, Abu Nowshed Chy, and Masaki Aono. 2019. Tweet stance detection using an attention based neural ensemble model. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1868–1873, Minneapolis, Minnesota. Association for Computational Linguistics.

Duyu Tang, Bing Qin, and Ting Liu. 2016. Aspect level sentiment classification with deep memory network. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 214–224, Austin, Texas. Association for Computational Linguistics.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Marilyn A Walker, Pranav Anand, Rob Abbott, Jean E Fox Tree, Craig Martell, and Joseph King. 2012. That is your evidence?: Classifying stance in online political debate. *Decision Support Systems*, 53(4):719–729.

Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, et al. 2022. Emergent abilities of large language models. *arXiv preprint arXiv:2206.07682*.

Jerry Wei, Jason Wei, Yi Tay, Dustin Tran, Albert Webson, Yifeng Lu, Xinyun Chen, Hanxiao Liu, Da Huang, Denny Zhou, et al. 2023. Larger language models do in-context learning differently. *arXiv preprint arXiv:2303.03846*.

Wan Wei, Xiao Zhang, Xuqin Liu, Wei Chen, and Tengjiao Wang. 2016. pkudblab at SemEval-2016 task 6 : A specific convolutional neural network system for effective stance detection. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 384–388, San Diego, California. Association for Computational Linguistics.

Guido Zarrella and Amy Marsh. 2016. MITRE at SemEval-2016 task 6: Transfer learning for stance detection. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 458–463, San Diego, California. Association for Computational Linguistics.

Chen Zhang, Qiuchi Li, and Dawei Song. 2019a. Aspect-based sentiment classification with aspect-specific graph convolutional networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4568–4578, Hong Kong, China. Association for Computational Linguistics.

Chen Zhang, Qiuchi Li, and Dawei Song. 2019b. Aspect-based sentiment classification with aspect-specific graph convolutional networks. *arXiv preprint arXiv:1909.03477*.

A Fine-tuning Details for `Llama 2` Models

The LoRA method was particularly designed to emphasize the queries and values in the self-attention modules (Hu et al., 2021). The hyperparameters for LoRA were set with a rank of 8, an α of 16, and a dropout rate of 0.05. We employed a warmup strategy, utilizing 10% of the training data. Training was set to run for three epochs with a learning rate of 3×10^{-4} and a batch size of 128. We trained the models with bfloat16 precision on an NVIDIA A100 GPU with 40GB. The fine-tuning of `Llama2-7b` on the SemEval-2016 dataset took approximately 20 minutes, while `Llama2-13b` took around 30 minutes.

B Prompting Technique

In our fine-tuning process, structured prompts were essential in creating the training and test datasets for the LLMs. The prompts are designed to offer context, guidelines, and the exact task the model is expected to accomplish. In this section, we provide a detailed overview of the prompts utilized for each dataset while fine-tuning ChatGPT.

B.1 ChatGPT Fine-tuning Prompts

B.1.1 SemEval-2016 Template

For the SemEval-2016 dataset, the following structured prompt was utilized:

> **### Instruction:**
>
> Analyze the tweet below in the following context: [topic]. Consider the text, subtext, regional and cultural references, and any implicit meanings to determine the stance expressed in the tweet towards the target. The possible stances are:
>
> - FAVOR: The tweet has a positive or supportive attitude towards the target, either explicitly or implicitly.
> - AGAINST: The tweet opposes or criticizes the target, either explicitly or implicitly.
> - NONE: The tweet is neutral or doesn't have a stance towards the target.
>
> **Tweet:** [tweet]
>
> **### Question:**
>
> What is the stance expressed in the tweet towards the target "[target]"?
>
> Choose one of the following options: FAVOR, AGAINST, NONE.
>
> **### Answer:**

For this prompt structure, placeholders are utilized: `[tweet]`, `[target]`, and `[topic]`.

- **`[tweet]`**: Represents the actual tweet being analyzed.
- **`[target]`**: Denotes what or whom the tweet's stance is directed at, whether directly or indirectly.
- **`[topic]`**: Offers a brief description of the `[target]`. Specifically for the SemEval-2016 dataset, this description was crafted by us to facilitate the understanding of the tweet's context.

When fine-tuning, these placeholders are substituted with real data, making it easier for the model to understand the context and identify the stance.

B.1.2 P-Stance Template

For the P-Stance dataset, the prompt tailored specifically for political domain analysis was:

> **### Instruction:**
>
> Analyze the following tweet, which is in the political domain, deeply. Consider

any subtext, regional and cultural references, or implicit meanings to determine the tweet's stance towards the target. The possible stances are:

- FAVOR: The tweet has a positive or supportive attitude towards the target, either explicitly or implicitly.
- AGAINST: The tweet opposes or criticizes the target, either explicitly or implicitly.

Tweet: [tweet]

Question:

What is the stance of the tweet above towards the target "[target]"?

Select from FAVOR or AGAINST.

Answer:

The placeholders `[tweet]` and `[target]` are used in a similar manner as explained for the SemEval-2016 template above.

B.2 Llama 2 Fine-Tuning Prompts

B.2.1 SemEval-2016 Llama 2 Template

This prompt template focuses on detecting the stance in tweets using a structured instruction to guide the model:

[INST] «SYS»
You are a helpful, respectful, and honest assistant for stance detection for a given target. Always answer from the possible options given below as helpfully as possible. Stance detection is the process of determining whether the author of a tweet is in support of or against a given target. The target may not always be explicitly mentioned in the text, and the tweet's stance can be conveyed implicitly through subtext, regional and cultural references, or other implicit meanings.
The possible stances are:

- support: The tweet has a positive or supportive attitude towards the target, either explicitly or implicitly.
- against: The tweet opposes or criticizes the target, either explicitly or implicitly.
- none: The tweet is neutral or doesn't have a stance towards the target.

</SYS>
Tweet: `[tweet]`
Stance towards the target `[target]`:[/INST]

For this prompt structure, placeholders are utilized: `[tweet]` and `[target]`.

- `[tweet]`: Represents the actual tweet being analyzed.
- `[target]`: Denotes what or whom the tweet's stance is directed at.

B.2.2 P-Stance Llama 2 Template

This prompt template is specifically designed for analyzing tweets related to the US presidential candidates:

[INST] «SYS»
You are a helpful, respectful, and honest assistant for stance detection for presidential candidates for the USA election. Always answer from the possible options given below as helpfully as possible. Stance detection is the process of determining whether the author of a tweet is in favor of or against a given target. The target may not always be explicitly mentioned in the text, and the tweet's stance can be conveyed implicitly through subtext, regional and cultural references, or other implicit meanings.
The possible stances are:

- support: The tweet has a positive or supportive attitude towards the target, either explicitly or implicitly.
- against: The tweet opposes or criticizes the target, either explicitly or implicitly.

</SYS>
Tweet: `[tweet]`
Stance towards the target `[target]`:[/INST]

The placeholders `[tweet]` and `[target]` are used in a similar manner as explained for the SemEval-2016 template above.

Note on Terminology: In the Llama 2 templates, we decided to use the term "support" instead of "favor". This decision was made based on token analysis for Llama 2, revealing that the model had a specific token for "support" but not for "favor". As a result, for the sake of efficiency, "support" was used in our prompt.

C Stance Detention Results

The summarized comparison for all targets in the SemEval-2016 Dataset is depicted in Table 3. This table encapsulates the strengths and potential areas of improvement for each model across different targets. Observing the data, `ChatGPT-ft` generally exhibits superior performance across the majority of the targets. Notably, for the Climate Change and Feminist Movement targets, this model distinctly leads, signifying its robustness in these domains. However, the competition tightens for the Hillary Clinton target, where the `Llama-2-13b-ft` model slightly surpasses both the `ChatGPT-ft` and `Llama-2-7b-ft`. This reveals that even though large language models like `ChatGPT-ft` generally excel, they can be outperformed in specific domains or targets by other variants. Furthermore, the performance of `Llama-2-7b-ft` is particularly intriguing, given that it achieves higher scores than its more sizable counterpart, `Llama-2-13b-ft`, in some targets like Atheism and Donald Trump. This variance reiterates the importance of model fine-tuning and adaptation for specific tasks, as mere model size does not guarantee consistent supremacy across all domains.

Model	**A**	**CC**	**DT**	**FM**	**HC**	**LA**
ChatGPT-ft	**81.3**	**86.2**	70.4	**79.7**	83.4	**72.6**
llama2-7b-ft	78.9	69.8	**72.0**	73.3	84.2	71.2
llama2-13b-ft	76.9	80.4	70.9	76.0	**84.8**	72.5

Table 3: F_{avg} scores among fine-tuned models for each target in SemEval-2016 Dataset.