Mitigating Biases of Large Language Models in Stance Detection with Counterfactual Augmented Calibration

Ang Li[1,3*], **Jingqian Zhao**[1*], **Bin Liang**[1,3,4], **Lin Gui**[5], **Hui Wang**[2], **Xi Zeng**[6], **Xingwei Liang**[1], **Kam-Fai Wong**[3,4], **and Ruifeng Xu**[1,2†]
[1] Harbin Institute of Technology, Shenzhen, China [2] Peng Cheng Laboratory, Shenzhen, China
[3] The Chinese University of Hong Kong, Hong Kong, China
[4] MoE Key Lab of High Confidence Software Technologies, CUHK, China
[5] King's College London, UK [6] The 30th Research Institute of China Electronics Technology Group Corporation, Chengdu, China
{angli,zhaojingqian}@stu.hit.edu.cn, xuruifeng@hit.edu.cn

Abstract

Stance detection is critical for understanding the underlying position or attitude expressed toward a topic. Large language models (LLMs) have demonstrated significant advancements across various natural language processing tasks including stance detection, however, their performance in stance detection is limited by biases and spurious correlations inherent due to their data-driven nature. Our statistical experiment reveals that LLMs are prone to generate biased stances due to sentiment-stance spurious correlations and preference towards certain individuals and topics. Furthermore, the results demonstrate a strong negative correlation between stance bias and stance detection performance, underscoring the importance of mitigating bias to enhance the utility of LLMs in stance detection. Therefore, in this paper, we propose a Counter**fact**ual **A**ugmented **Cal**ibration Network (FACTUAL), which a novel calibration network is devised to calibrate potential bias in the stance prediction of LLMs. Further, to address the challenge of effectively learning bias representations and the difficulty in the generalizability of debiasing, we construct counterfactual augmented data. This approach enhances the calibration network, facilitating the debiasing and out-of-domain generalization. Experimental results on in-target and zero-shot stance detection tasks show that the proposed FACTUAL can effectively mitigate biases of LLMs, achieving state-of-the-art results.

1 Introduction

Stance detection aims at automatically identifying the author's opinionated standpoint or attitude (e.g., *Favor*, *Against*, or *Neutral*) expressed in the content towards a specific target, topic, or proposition (Somasundaran and Wiebe, 2010; Mohammad et al., 2016). With the development of social media platforms, stance detection plays a pivotal role in analyzing public opinion on social media topics (Jang and Allan, 2018; Ghosh et al., 2019; Stefanov et al., 2020; Sun et al., 2018; Chen et al., 2021).

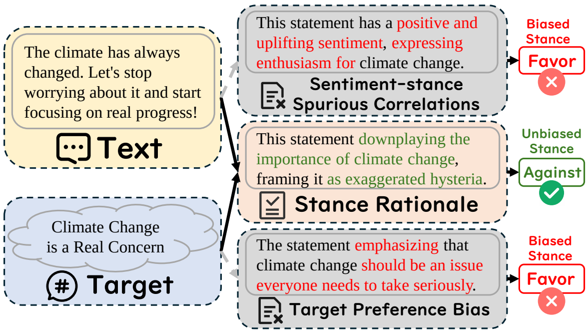

Figure 1: An example demonstrates two types of biases encountered by large language models in stance detection tasks (shown at the top and bottom) as well as unbiased stance rationale (shown in the middle).

Large Language Models (LLMs), such as ChatGPT[1], Bard[2], and LLaMA (Touvron et al., 2023), have demonstrated impressive language comprehension and task-handling capabilities by leveraging extensive corpus and knowledge. However, their data-driven nature makes them susceptible to biases and spurious correlations embedded in pretraining data. In stance detection, which requires interpreting the relationship between a sentence and a specific topic, clues from any isolated aspects could become spurious and lead to biased stances.

Our experiment identifies two primary biases in LLMs for stance detection: (1) sentiment-stance spurious correlations, where sentiment misleads stance judgment, and (2) target preference bias, where LLMs favor certain individuals or topics.

* The first two authors contribute equally to this work.
† Corresponding Author

[1] https://openai.com/blog/chatgpt/
[2] https://bard.google.com/

Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), pages 7075–7092
April 29 - May 4, 2025

Figure 1 illustrates examples of the two types of biases, as well as unbiased stance rationale. Furthermore, our results reveal a significant negative correlation between stance bias and stance detection performance, emphasizing the necessity of alleviating bias to improve the effectiveness of LLMs in stance detection.

Existing research of debiasing in stance detection largely centered on the creation of unbiased training samples and the retraining of stance detection models (Kaushal et al., 2021; Yuan et al., 2022b). However, there are two core limitations to the application of these debiasing methods in LLMs. **Limitation#1**, research (Luo et al., 2023) has shown that such retraining processes will undermine the generality of LLMs, potentially leading to catastrophic forgetting; not to mention that there are restrictions with certain closed-source LLMs like GPT-3.5-turbo, which can only be accessed with a restricted inference API, preventing access to internal model parameters. **Limitation#2**, existing approaches to constructing unbiased training samples typically entail the analysis of prevalent bias patterns, subsequently automating their construction based on these identified patterns, exemplified by substituting 'Men' with 'Women'. However, when dealing with stance detection tasks, our forthcoming analysis illuminates that these samples display varying bias propensities, attributable to divergences in sentiments and stance objectives. Consequently, utilizing conventional methods to create unbiased samples poses a significant challenge.

Therefore, to address the above two limitations, we propose to mitigate biases of LLMs in stance detection with a Counter**fact**ual A**u**gmented **Cal**ibration Network, coined as FACTUAL. We establish a trainable calibration network to approximate the inverse projection function of the bias label distribution within LLMs. This calibration network takes samples as input, including stance judgments and rationales from LLMs, and generates calibrated stance judgments. We construct counterfactual augmented data against the training data to rectify stance biases. The counterfactual samples are constructed from both causal and non-causal features, which can enhance the calibration network to yield unbiased stances and accomplish out-of-domain generalization. Through counterfactual augmented supervised training, the calibration network can capture biases present in specific samples, thereby performing debiasing. The main contributions of our work are summarized as follows:

1) We are the first to investigate the biases of LLMs on stance detection, categorizing the biases into two main types from the perspective of causality and proposing metrics to quantify these two types of biases.

2) We propose FACTUAL, a novel framework called the Counterfactual Augmented Calibration Network to mitigate biases of LLMs on stance detection.

3) A series of experiments demonstrate that our FACTUAL can effectively reduce the bias of LLMs in stance detection, improving the performance in both in-target and zero-shot stance detection tasks[3].

2 Related Work

Biases in Large Language Models Some studies (Gonçalves and Strubell, 2023) have examined the biases existing in Large Language Models (LLMs), these biases mainly include gender and religion (Salinas et al., 2023), politics (Jenny et al., 2023; He et al., 2023), and spurious correlations (Zhou et al., 2023). The associated debiasing efforts are centered around retraining the language model with debiased samples (Dong et al., 2023; Limisiewicz et al., 2023). Zheng et al. (2023) found that LLMs are vulnerable to option position changes in MCQs due to their inherent 'selection bias'. They perform debiasing by approximating the overall bias distribution. While based on our analysis in Section 3, the bias distribution varies significantly across different stance detection samples, so this method is not applicable.

Mitigating Biases in Stance Detection Currently, studies developed for mitigating biases in stance detection are oriented toward fine-tuned models. Kaushal et al. (2021) analyzed two biases existing in the current datasets: target-independent lexical choices and target-independent sentiment-stance correlations, and built an unbiased dataset. Yuan et al. (2022a) incorporated the stance reasoning process as task knowledge to retrain the model to reduce bias. Yuan et al. (2022b) constructed unbiased samples through counterfactual reasoning and performed adversarial bias learning. These methods involve retraining models and constructing unbiased training samples through special marks, which cannot be directly applied to LLMs.

[3]The code is available at https://github.com/Leon-Francis/FACTUAL.

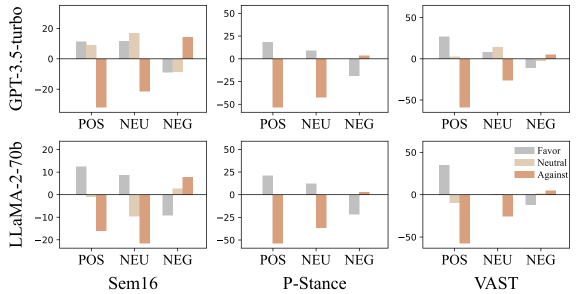

Figure 2: The recall score of each stance label on three sentiment subsets, normalizing by subtracting the overall recall scores of the corresponding stance labels across overall dataset, on Sem16, P-Stance, and VAST. POS for positive, NEU for neutral, NEG for negative.

3 Biases of LLMs in Stance Detection

3.1 Bias Measurement

Stance bias refers to the systematic errors where models tend to choose certain stances due to the influence of specific biases and stereotypes. Inspired by Zheng et al. (2023), the standard deviation of recalls (*RStd*) on stance labels is an excellent metric for quantitatively measuring systematic errors. The formula is as follows:

$$RStd = \sqrt{\frac{1}{K}\sum_{i=1}^{K}\left(\frac{TP_i}{P_i} - \frac{1}{K}\sum_{j=1}^{K}\frac{TP_j}{P_j}\right)^2} \quad (1)$$

Where K is the number of stance labels, TP_i is the number of true positive instances for stance label i, and P_i is the number of instances of stance label i. This measurement resists label imbalance and effectively reflects the model's bias tendency on samples (Refer to Appendix A for the validation).

3.2 Experimental Result

Through statistical analysis of the results from LLMs, we identified two significant types of bias: **Sentiment-stance Spurious Correlations** and **Target Preference Bias**.

3.2.1 Sentiment-Stance Spurious Correlations

Sentiment can influence stance judgment but is not the primary determinant. Overreliance on sentiment by the model suggests susceptibility to sentiment-stance spurious correlations, leading to biased stance assessments. To investigate stance bias across different sentiments, we first ascertain the sentiment label for each sample. In the Sem16 dataset, each sample has annotated sentiment labels, categorized as positive, neutral, or negative. For the P-Stance and VAST datasets, we utilize GPT-4 to annotate the sentiment labels. To gain a preliminary understanding of sentiment-stance spurious correlations, we first divide the dataset into three subsets based on sentiment categories. For each subset, we calculate the recall score for each stance label and normalize it by subtracting the overall recall score of the corresponding stance labels across the dataset, as shown in Figure 2. Observations indicate that LLMs tend to erroneously predict 'support' for positively-sentiment samples and 'against' for negatively-sentiment ones, indicating a deviation from expected patterns and highlighting an inherent stance bias. Hence, we identify the **S**entiment-stance **S**purious **C**orrelations (SSC) as a type of bias in LLMs on stance detection.

We calculate the average of the RStd across all sentiments as our quantification for sentiment-stance spurious correlations:

$$Bias\text{-}SSC = \frac{1}{|S|}\sum_{s\in S} RStd(X_s) \quad (2)$$

where X_s represents instances with sentiment label s, $|S|$ denotes the number of sentiment labels, which in our experiment, is 3.

We conducted experiments in various settings: *Task-Des* used task-related descriptions for stance judgment[4], *CoT-Demo* used the task description with 4-shot chain-of-thought demonstration, and *Debias-Instruct* used the task description indicating that sentiment was spurious cues for stance judgment. Refer to Appendix C for the detailed prompts. The results are shown in Table 1. We can observe that in most cases, there is a negative correlation between bias-SSC and stance detection performance. See further analysis in Appendix B. Moreover, prompt engineering methods proved ineffectual in mitigating this inherent bias.

3.2.2 Target Preference Bias

LLMs exhibit bias towards certain individuals or topics. This bias can interfere with their ability to judge stances based on the text, leading to biased stance judgments. We refer to this bias as target preference bias. To preliminarily observe the target preference bias of LLMs, we randomly sampled some targets from different datasets, calculated the recall scores for each stance label on each target subset, and normalized it by subtracting the overall recall score of the corresponding

[4]The results presented in Figures 2 and 3 are obtained based on *Task-Des*.

	Sem16		P-Stance		VAST	
	SSC↓	F1↑	SSC↓	F1↑	SSC↓	F1↑
LLaMA-2-70b-chat						
Task-Des	**17.80**	60.08	23.36	79.89	**16.87**	68.36
CoT-Demo	27.52	58.68	**22.81**	**80.77**	22.55	67.08
Debias-Instruct	19.24	**63.62**	24.86	78.85	19.63	**68.68**
GPT-3.5-Turbo-0125						
Task-Des	27.13	52.82	23.72	**81.62**	28.70	49.86
CoT-Demo	**18.08**	**67.59**	**22.75**	80.88	**16.32**	**69.90**
Debias-Instruct	23.75	51.77	23.48	81.48	30.53	48.68

Table 1: Bias-SSC and macro F1-score of stance detection on the Sem16, P-Stance and VAST dataset. Refer to Appendix D for detailed results on each sentiment.

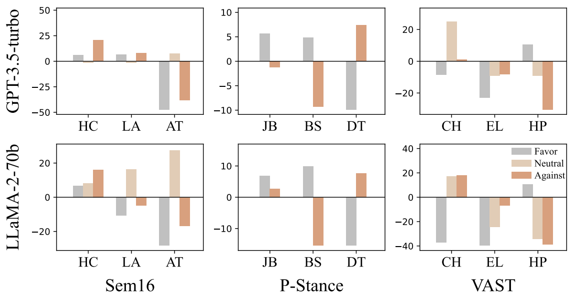

Figure 3: The recall score of each stance label on several target subsets, normalizing by subtracting the overall recall score of the corresponding stance labels across all targets, on Sem16, P-Stance, and VAST dataset. HC for Hillary Clinton, LA for Legalization of Abortion, AT for Atheism, JB for Joe Biden, BS for Bernie Sanders, DT for Donald Trump, CH for Christian, CL for Election, HP for Humanity Program.

stance labels across all targets, as shown in Figure 3. We observed that, on different targets, LLMs displayed markedly different tendencies in stance selection, which ultimately affected the correctness of stance judgment. Therefore, we identify the **T**arget **P**reference **B**ias (TPB) as a type of bias in LLMs on stance detection.

We calculate the average of the RStd of all targets as our quantification for target preference bias:

$$Bias\text{-}TPB = \frac{1}{|T|}\sum_{t \in T} RStd(X_t) \tag{3}$$

where X_t represents instances with stance target t, $|T|$ denotes the number of targets.

We conduct experiments based on *Task-Des*, *CoT-Demo*, and *Debias-Instruct* which emphasize the need to judge the stance based on the text and not to include the inherent attitude towards the target. Refer to Appendix C for the detailed prompts. The results are shown in Table 2. We can observe that in most cases, there is a negative correlation between bias-TPB and stance detection performance.

	Sem16		P-Stance		VAST	
	TPB↓	F1↑	TPB↓	F1↑	TPB↓	F1↑
LLaMA-2-70b-chat						
Task-Des	17.59	60.08	9.09	79.89	7.76	68.36
CoT-Demo	27.56	58.68	11.57	**80.77**	9.64	67.08
Debias-Instruct	**16.37**	**61.40**	**8.94**	78.70	**4.86**	**69.10**
GPT-3.5-Turbo-0125						
Task-Des	22.64	52.82	**5.43**	**81.62**	28.44	49.86
CoT-Demo	**13.47**	**67.59**	6.61	80.88	**8.40**	**69.90**
Debias-Instruct	21.87	53.33	5.79	81.59	26.77	51.66

Table 2: Bias-TPB and macro F1-score of stance detection on the Sem16, P-Stance and VAST dataset. Refer to Appendix D for detailed results on each target.

See further analysis in Appendix B, and prompt engineering fails to effectively mitigate bias-TPB.

4 Mitigating Bias with Calibration

Given $\{x_n, t_n\}_{n=1}^{N}$ as the labeled dataset, where x denotes the input text and t denotes the corresponding target, LLMs obtain the stance predictions y through the task instructions $\mathcal{I}$ for stance detection: $P_{obs}(y_i|\mathcal{I}; x_i, t_i)$. Inspired by Zheng et al. (2023), we believe that it can be deconstructed into the unbiased distribution $P_{unbiased}$ of the LLMs performing the stance detection task, and the bias distribution P_{bias} formed by confounding C_i:

$$P_{obs} = P_{unbiased}(y_i|x_i, t_i)P_{bias}(y_i|C_i) \tag{4}$$

The confounding factor C_i arises from the two types of biases analyzed earlier. It is important to note that it may be unaffected, influenced by a single bias, or impacted by both types of biases. Specifically, P_{obs} denotes the stance judgment text, which can be regarded as the outcome of word probability distribution following the argmax operation, derived from LLMs. We aim to estimate the unbiased stance distribution $P_{unbiased}$.

4.1 Calibration Network

By estimating the bias distribution based on the overall distribution of known samples (from the training set), we can obtain unbiased outputs by multiplying the observed distribution of LLMs by the inverse of the approximated bias distribution:

$$P_{unbiased} = P_{obs}(y_i'|\mathcal{I}; x_i, t_i)\widetilde{P}_{bias}(y_i''|C_i)^{-1} \tag{5}$$

where y_i' and y_i'' represent the label distribution output by LLMs and the label distribution affected by bias, $\widetilde{P}_{bias}$ represents the estimate of bias.

However, based on the bias analysis in Section 3, we found that for stance detection, the samples with

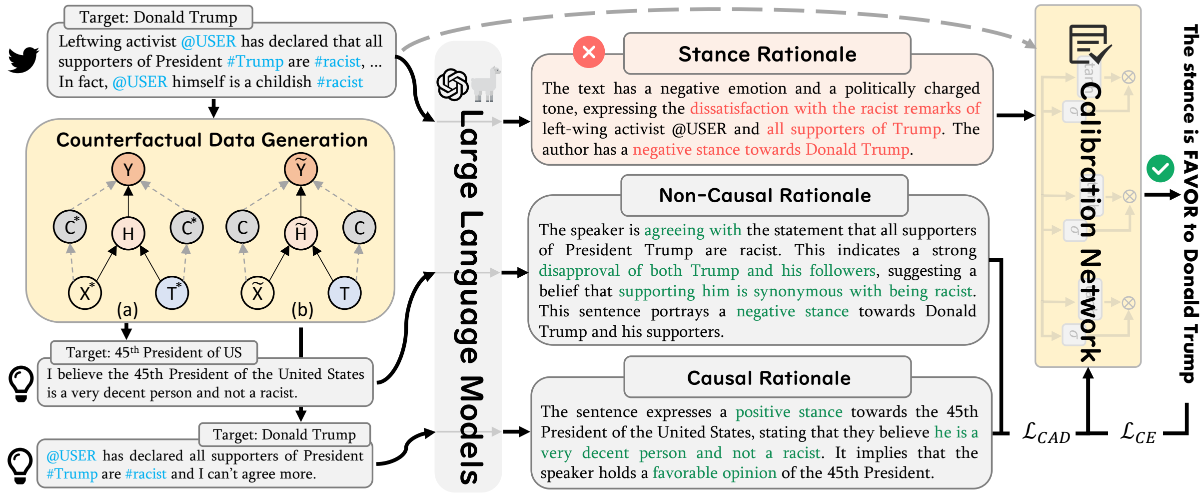

Figure 4: The overall architecture of our proposed FACTUAL. (a) and (b) in the counterfactual data generation represent two ways to generate counterfactual augmentation. X donates the text, T donates the target, H donates the features of the interaction of text and target, and Y donates the stance label. C represents confounding factors, which arise from the two types of biases previously analyzed and may distort the stance prediction. $*$ denotes the perturbation of non-causal features, and $\sim$ denotes the perturbation of causal features.

different sentiments and stance targets have completely different stance bias distributions. Therefore, we propose employing a network to capture the bias distribution specific to each sample, called the *calibration network* f_{Cal}. We use the network f_{Cal} to approximate the inverse projection function of the bias distribution:

$$f_{Cal} = P_{bias}(y_i''|C_i)^{-1} \tag{6}$$

By inputting the predicted stance distribution P_{obs} from LLMs, an approximating unbiased label can be obtained:

$$P_{unbiased}(\hat{y}_i) = f_{Cal}(P_{obs}(y_i'|\mathcal{I}; x_i, t_i)) \tag{7}$$

Specifically, as illustrated in Figure 4, we first use the *CoT-Demo* instruction (refer to Appendix C for the detail) to obtain the stance judgment and rationale from the LLMs (which correspond to the result of an argmax operation over the P_{obs}). Then, we input the sample, along with this stance judgment and rationale, into our *calibration network* (using RoBERTa-base in our setup) to obtain the debiased stance output. We train the *calibration network* using the cross-entropy loss function with ground truth label:

$$\mathcal{L}_{CE} = -\sum_{i=1}^{N} y_i \log(f_{Cal}(P_{obs}(y_i'|\mathcal{I}; x_i, t_i)) \tag{8}$$

4.2 Counterfactual Data Augmentation

One challenge in supervised training is the limited representation of bias within the overall training set, and the learned bias features are difficult to generalize. To fully leverage our analysis of the existing stance biases in LLMs (Section 3) and facilitate the *calibration network* in learning diverse bias patterns, we generate non-causal and causal **C**ounterfactual **A**ugmented **D**ata (CAD) based on the training data.

The objective of non-causal counterfactual data augmentation is to explicitly perturb bias-inducing features in the data, enabling the model to identify and mitigate biases. As illustrated in Figure 4 (a), we construct counterfactual samples by modifying non-causal features while preserving stance labels. Specifically, we construct an instruction that allows the LLMs to perturb the sentence and target. To address **sentiment-stance spurious correlations**, we perturb the text x_i by altering sentiment-related expressions while maintaining the stance. Similarly, to counteract **target preference bias**, we modify the target t_i while ensuring the stance remains unchanged. Refer to Figure 10 for the detailed prompts. This obtains the perturbed text x_i^* and perturbed target t_i^*. Since we only disturbed confounding, the stance label remains unaffected. We construct cross-entropy loss on non-causal counterfactual augmented data as follows:

$$\mathcal{L}_{CAD}^{n\text{-}cau} = -\sum_{i=1}^{N} y_i \log(f_{Cal}(P_{obs}(y_i'|\mathcal{I}; x_i^*, t_i^*)) \tag{9}$$

In contrast, causal counterfactual data augmentation directly manipulates causal features by re-

Dataset	Target	Favor	Against	Neutral
Sem16	HC	163	565	256
	FM	268	511	170
	LA	167	544	222
	A	124	464	145
	CC	335	26	203
	DT	148	299	260
P-Stance	Biden	3217	4079	-
	Sanders	3551	2774	-
	Trump	3663	4290	-
VAST	-	6952	7297	4296

Table 3: Statistics of SemEval-2016 Task6, P-Stance and VAST datasets.

versing the stance. This augmentation improves the model's capacity to focus on causal features, thereby reducing its reliance on spurious correlations that may introduce bias. As illustrated in Figure 4 (b), we make necessary alterations to text x_i to reverse the stance to target t_i, thereby only perturbing the causal features. Refer to Figure 11 for the detailed prompts. This obtains the perturbed text $\widetilde{x_i}$ expressing a reversed stance to target t_i. We construct cross-entropy loss on causal counterfactual augmented data as follows:

$$\mathcal{L}_{CAD}^{cau} = \sum_{i=1}^{N} y_i \log(f_{Cal}(P_{obs}(y_i'|\mathcal{I};\widetilde{x_i},t_i))) \quad (10)$$

4.3 Training Objective

The final training objective incorporates counterfactual augmented data and performs joint training:

$$\mathcal{L} = \mathcal{L}_{CE} + \mathcal{L}_{CAD}^{n\text{-}cau} + \mathcal{L}_{CAD}^{cau} \quad (11)$$

5 Experimental Setup

5.1 Datasets

We conduct experiments of in-target and zero-shot stance detection on three benchmark datasets: SemEval-2016 Task6 (Sem16) (Mohammad et al., 2016), P-Stance (Li et al., 2021) and Varied Stance Topics (VAST) (Allaway and McKeown, 2020). The statistic of datasets is shown in Table 3.

5.2 Implementation Details

For GPT-3.5-turbo, we utilize GPT-3.5-turbo-0125. For LLaMA2-70b, we utilize LLaMA2-70b-chat. We set the temperature to 1.0, top p to 1.0, max tokens to 1024, and fixed the decoding seed to ensure the reproducibility of our experiments. For our calibration network, we employ the RoBERTa-base model (Liu et al., 2019). For our counterfactual data augmentation, we employ GPT-3.5-turbo-0301 to generate counterfactual samples, guided by the instructions detailed in Appendix C. We use AdamW as an optimizer with a batch size of 32. Learning rate is set to 1e-5 and weight decay is set to 1e-3. All training was conducted on NVIDIA A100 40G GPUs. We report averaged scores of 5 runs to obtain statistically stable results.

5.3 Evaluation Metric

Across three datasets, we used the same evaluation metric established by their proposers, which was also adopted by most of the subsequent baselines. We adopt the macro-average of the F1-score as the evaluation metric. For Sem16 and P-Stance, we report $F1 = (F_{favor} + F_{against})/2$. For VAST, we report $F1 = (F_{favor} + F_{against} + F_{none})/3$. In the in-target stance detection setting, the model is trained and tested on the same set of targets. We follow the dataset splits provided by the original dataset publisher to ensure a fair comparison with related baselines. The zero-shot stance detection setting presents the model with unseen targets during testing, requiring it to generalize from known targets to infer stances toward new ones. For Sem16 and P-Stance, we take one target as the test set while splitting the remaining targets into training and validation sets in a 7:1 ratio. For VAST, which is inherently a zero-shot stance detection dataset, we use the original splits provided by the dataset publisher. These settings are consistent with the baselines to ensure fairness in comparison.

5.4 Comparison Models

The fine-tuned model baselines include vanilla RoBERTa (Liu et al., 2019), domain pre-trained model: BERTweet (Nguyen et al., 2020), prompt tuning method KPT (Shin et al., 2020), joint contrastive learning framework: JointCL (Liang et al., 2022), incorporating ConceptGraph knowledge model: KEprompt (Huang et al., 2023), incorporating Wikipedia knowledge model: TarBK-BERT (Zhu et al., 2022) and WS-BERT (He et al., 2022), incorporating knowledge from LLMs: KASD-BERT (Li et al., 2023). For large language models, we compare baselines include Task-Des (Zhang et al., 2022), CoT-Demo (Zhang et al., 2023b), the self-consistent chain-of-thought: CoT-SC (Wang et al., 2023), incorporating Wikipedia knowledge for retrieval-augmented generation: KASD-ChatGPT and KASD-LLaMA-2 (Li et al., 2023), fine tuning LLaMA-2-7b using LoRA with training set: LLaMA-2-7b-FT, utilizing collaborative role-infused LLM-based agents: COLA (Lan

	Sem16(%)						P-Stance(%)			
	HC	FM	LA	A	CC	Avg	Biden	Sanders	Trump	Avg
Fine-tuning Based Methods										
Roberta	55.97	68.19	67.60	65.40	43.08	58.71	84.29	79.56	82.70	82.18
BERTweet	62.31	64.20	64.14	68.12	41.30	57.99	78.09	81.02	82.48	80.53
KPT	71.30	63.30	63.50	-	-	-	80.40	77.10	80.20	79.23
KEprompt	77.10$^\sharp$	68.30$^\sharp$	70.30$^\sharp$	-	-	-	84.40$^\sharp$	-	83.20$^\sharp$	-
WS-BERT-Dual	75.26†	66.02†	70.42†	71.57†	57.31†	68.12†	83.50$^\flat$	79.00$^\flat$	85.80$^\flat$	82.77$^\flat$
KASD-BERT	77.60†	70.38†	72.29†	72.32†	61.47†	70.81†	85.66†	80.39†	85.35†	83.80†
LLaMA-2 Based Methods										
LLaMA-2-Task-Des	75.96	66.60	61.68	53.40	73.56	66.24	84.31	77.29	78.08	79.89
LLaMA-2-CoT-Demo	74.84	71.45	62.67	57.58	73.26	67.96	85.03	79.77	77.52	80.77
KASD-LLaMA-2	77.89†	67.29†	52.00†	35.78†	47.12†	56.02†	79.59†	71.32†	67.89†	72.93†
LLaMA-2-7b-FT	**81.86**	71.58	65.56	68.74	75.59	72.67	85.79	81.25	**87.47**	84.84
FACTUAL$_{\text{LLaMA-2}}$ (Ours)	80.44	**73.46***	**67.18**	**71.85**	**76.19**	**73.82***	**86.34**	**83.06***	85.58	**84.99**
- w/o CAD	78.00	70.82	65.57	71.40	72.24	71.61	85.35	82.00	85.51	84.29
GPT-3.5-Turbo Based Methods										
GPT-3.5-Turbo-Task-Des	73.33	66.81	67.22	25.18	72.54	61.02	83.20	80.02	81.66	81.62
GPT-3.5-Turbo-CoT-Demo	81.58	73.42	68.28	64.96	78.35	73.32	83.07	77.98	81.59	80.88
KASD-ChatGPT	80.92†	70.37†	63.26†	61.92†	62.72†	67.84†	84.59†	79.96†	**85.06**†	83.20†
FACTUAL$_{\text{GPT-3.5}}$ (Ours)	**83.38***	**78.46***	**69.36***	**69.56***	**80.05***	**76.16***	**86.03***	**81.60***	84.95	**84.20***
- w/o CAD	82.38	73.80	63.65	69.21	62.93	70.39	85.40	81.36	85.00	83.92

Table 4: In-target stance detection experiment results on Sem16 and P-Stance datasets. The results with $\sharp$ are retrieved from (Huang et al., 2023), $\flat$ from (He et al., 2022), $\dagger$ from (Li et al., 2023). The best scores over the same type are in bold. Results with $\star$ indicate significance of our FACTUAL over the same type baseline models at $p < 0.05$.

et al., 2023) and utilizing logically consistent chain-of-thought: LC-CoT (Zhang et al., 2023a)[5].

6 Experimental Results

6.1 In-Target Stance Detection

We perform experiments on Sem16 and P-Stance for in-target stance detection. The results are presented in Table 4. It shows that our FACTUAL outperforms all baselines based on different large language models. We can observe that *FACTUAL - w/o CAD*, which is without the counterfactual data enhancement, the calibration network trained exclusively on the training set data can still improve stance detection performance. Moreover, the application of counterfactual data enhancement furthers the model's performance. When compared to the *LLaMA-2-7b-FT* method, which fine-tunes LLaMA-2-7b, our method attains superior accuracy in stance detection with significantly reduced computation resources.

6.2 Zero-Shot Stance Detection

We conduct experiments on Sem16, P-Stance, and VAST for zero-shot stance detection. The results are shown in Table 5. It shows that our FACTUAL outperforms all baselines including both fine-tuned models and large language models. This indicates that our FACTUAL has strong generalization capabilities and can perform well on unseen targets. We can observe that *FACTUAL - w/o CAD* exhibits subpar performance. This can be attributed to the constraints posed by the exclusive reliance on fine-tuning within the limited training dataset, making it an uphill task to generalize the model's debiasing capability. Conversely, employing our counterfactual data enhancement bolsters the out-of-domain generalization prowess of the model considerably, yielding impressive results in zero-shot performance. Compared to the *LLaMA-2-7b-FT*, which performs poorly on zero-shot tasks, FACTUAL demonstrates strong generalization capabilities.

6.3 Mitigating Biases Effect Analysis

We conduct experiments to evaluate the Bias-SSC and Bias-TPB of LLMs and further assess the impact of our bias mitigation efforts. The results are shown in Table 6, which indicate that our FACTUAL can effectively alleviate Bias-SSC and Bias-TPB for both GPT-3.5-turbo and LLaMA2-70b, thus validating its effectiveness in mitigating biases. The inclusion of counterfactual data augmentation can effectively improve its debiasing ability, indicating the importance of our counterfactual data augmentation. Our findings also highlight that the integration of counterfactual data augmentation enhances the debiasing capacity of the model, thereby em-

[5] Since JointCL, TarBK-BERT, COLA, and LC-CoT were only proposed in zero-shot stance detection scenarios, our comparisons with them are conducted solely within the corresponding experimental settings.

	Sem16(%)							P-Stance(%)				VAST(%)
	DT	HC	FM	LA	A	CC	Avg	Biden	Sanders	Trump	Avg	All
Fine-tuning Based Methods												
Roberta	32.12	43.45	40.38	38.79	26.80	18.70	33.37	76.29	72.07	67.56	71.97	73.18
BERTweet	26.88	44.82	21.97	31.91	30.49	12.48	28.09	73.13	68.22	67.66	69.67	71.10
JointCL	50.50$^\natural$	54.80$^\natural$	53.80$^\natural$	49.50$^\natural$	54.50$^\natural$	39.70$^\natural$	50.47$^\natural$	-	-	-	-	72.30
TarBK-BERT	50.80$^\sharp$	55.10$^\sharp$	53.80$^\sharp$	48.70$^\sharp$	56.20$^\sharp$	39.50$^\sharp$	50.68$^\sharp$	75.49	70.45	65.80	70.58$^\sharp$	73.60$^\sharp$
KASD-BERT	54.74†	64.78†	57.13†	51.63†	55.97†	40.11†	54.06†	79.04†	75.09†	70.84†	74.99†	76.82†
LLaMA-2 Based Methods												
LLaMA-2-Task-Des	66.03	73.79	71.03	66.00	**60.44**	61.91	66.53	82.81	78.00	**78.87**	79.89	68.54
LLaMA-2-CoT-Demo	58.56	72.09	73.83	66.10	57.58	62.47	65.11	83.97	79.26	77.96	80.40	67.28
KASD-LLaMA-2	-	**77.70**†	65.57†	57.07†	39.55†	50.72†	-	75.28†	74.09†	69.27†	72.88†	43.42†
LLaMA-2-7b-FT	63.99	55.49	59.46	33.18	46.37	58.24	52.79	83.93	77.00	74.35	78.43	77.80
FACTUAL$_{\text{LLaMA-2}}$ (Ours)	**66.96**	77.19	**74.71**	**72.49***	58.29	**67.71***	**69.56***	**84.04**	**81.22***	77.57	**80.94**	**79.62***
- w/o CAD	61.99	69.22	62.77	60.39	40.83	63.69	59.81	83.09	78.21	76.74	79.35	76.61
GPT-3.5-Turbo Based Methods												
GPT-3.5-Turbo-Task-Des	61.72	72.70	71.71	67.89	28.87	59.36	60.38	84.08	80.38	82.38	82.28	50.21
GPT-3.5-Turbo-CoT-Demo	64.16	78.69	73.22	72.84	65.15	**75.20**	71.54	84.08	80.12	82.24	82.15	70.14
KASD-ChatGPT	64.23†	80.32†	70.41†	62.71†	63.95†	55.83†	66.24†	83.60†	79.66†	84.31†	82.52†	67.03†
COLA	71.20‡	75.90‡	69.10‡	**71.00**‡	62.30‡	64.00‡	68.92‡	-	-	-	-	73.40‡
LC-CoT	71.70$^\flat$	**82.90**$^\flat$	70.40$^\flat$	63.20$^\flat$	-	-	-	-	-	-	-	72.50$^\flat$
FACTUAL$_{\text{GPT-3.5}}$ (Ours)	**72.80***	80.26	**75.76***	68.77*	**66.54***	71.00	**72.52***	**85.14**	**81.05***	**85.08**	**83.76***	**79.98***
- w/o CAD	63.28	72.65	60.88	62.07	41.65	67.80	61.39	84.26	77.80	75.26	79.11	77.50

Table 5: Zero-shot stance detection experiment results on Sem16, P-Stance and VAST dataset. The results with $\natural$ are retrieved from (Liang et al., 2022), $\sharp$ from (Zhu et al., 2022), $\dagger$ from (Li et al., 2023), $\ddagger$ from (Lan et al., 2023), $\flat$ from (Zhang et al., 2023a). The best scores over the same type are in bold. Results with $\star$ denote the significance tests of our FACTUAL over the same type baseline models at p-value < 0.05.

	Sem16		P-Stance		VAST	
	SSC↓	TPB↓	SSC↓	TPB↓	SSC↓	TPB↓
LLaMA-2 Based Methods						
Task-Des	17.80	17.59	23.36	9.09	23.87	17.76
CoT-Demo	27.52	27.56	22.81	11.57	22.55	9.64
CoT-SC	33.67	27.18	29.85	6.34	31.98	23.70
LLaMA-2-7b-FT	22.44	25.09	18.14	5.07	22.36	6.84
KASD-LLaMA-2	18.74	10.90	18.74	4.43	20.51	18.00
FACTUAL$_{\text{LLaMA-2}}$	**9.61***	**5.52***	**17.15**	**2.81***	**19.89**	**5.42**
- w/o CAD	15.43	12.07	19.15	5.81	21.89	11.42
GPT-3.5-Turbo Based Methods						
Task-Des	27.13	22.64	23.72	5.43	28.70	28.44
CoT-Demo	18.08	13.47	22.75	6.61	16.32	18.40
CoT-SC	21.42	16.03	26.30	8.06	22.58	21.63
KASD-ChatGPT	19.49	20.17	20.90	13.94	20.54	15.56
FACTUAL$_{\text{GPT-3.5}}$	**11.31***	**7.74***	**16.00***	**3.38**	**13.25***	**7.03***
- w/o CAD	12.92	11.83	18.00	4.38	17.27	12.04

Table 6: Results of Bias-SSC and Bias-TPB in in-target stance detection on Sem16, P-Stance, and zero-shot stance detection on VAST. The best scores are in bold. Results with $\star$ denote the significance tests of our FACTUAL over the same type baseline models at p-value < 0.05.

phasizing the significance of this augmentation in our methodology.

6.4 Ablation Study

We conduct ablation studies to examine the impact of different components in our FACTUAL: (1) "**w/o** Calibration" denotes without the calibration network, letting the LLMs directly output stance

	Sem16			VAST		
	Avg↑	SSC↓	TPB↓	All↑	SSC↓	TPB↓
FACTUAL$_{\text{LLaMA-2}}$	**73.82***	**9.61**	**5.52***	**79.62**	**19.89***	5.42
w/o Calibration	67.96	27.52	27.56	67.28	22.55	9.64
w/o CAD	71.61	15.43	12.07	76.61	21.89	11.42
- w/o non-causal	71.29	13.12	13.25	79.38	24.04	7.12
- w/o causal	69.39	10.66	14.82	77.45	21.69	**3.97**
FACTUAL$_{\text{GPT-3.5}}$	**76.16***	11.31	**7.74***	**79.98**	**13.25***	7.03
w/o Calibration	73.32	18.08	13.47	70.14	16.32	18.40
w/o CAD	70.39	12.92	11.83	77.50	17.27	12.04
- w/o non-causal	74.08	14.36	11.80	79.38	26.48	9.32
- w/o causal	71.71	**10.22**	10.21	77.80	22.25	**6.81**

Table 7: Experimental results of ablation study of in-target stance detection on the Sem16, and zero-shot stance detection on VAST. Results on P-Stance are shown in Table 10. The best scores are in bold. Results with $\star$ indicate the significance tests of our FACTUAL over the ablation experiments at p-value < 0.05.

labels. (2) "**w/o** CAD" denotes without the counterfactual augmented data when training the calibration network. (3) "**w/o** non-causal" denotes without the non-causal counterfactual augmented data when training the calibration network. (4) "**w/o** causal" denotes without the causal counterfactual augmented data when training the calibration network.

The results are presented in Table 7. Note that despite utilizing the same stance reasoning, a lack of calibration can result in sub-optimal results and notable biases. Thus validating the effectiveness of our gate calibration network. Analysis in Appendix E demonstrate that our method exhibits robustness across different prompt templates.

Additionally, removing non-causal counterfactual data significantly increases bias, highlighting its crucial role in bias mitigation. Conversely, eliminating causal counterfactual data markedly reduces performance, underscoring its substantial impact on the accuracy and generalizability of the calibration network.

7 Conclusion

In this paper, we categorize the biases of LLMs in stance detection into two types from the perspective of causality and propose metrics to quantify these biases. Then, we propose a Counter**fact**ual A**u**gmented C**al**ibration Network, coined as `FACTUAL`. In which, a trainable calibration network and counterfactual data augmentation are explored to mitigate the biases of LLMs in stance detection. Experimental results on in-target and zero-shot stance detection show that our `FACTUAL` can effectively reduce the bias of LLMs in stance detection and contribute to improved performance.

Acknowledgements

This work was partially supported by the National Natural Science Foundation of China 62176076, Natural Science Foundation of Guang Dong 2023A1515012922, the Shenzhen Foundational Research Funding JCYJ20220818102415032, the Major Key Project of PCL2021A06, CIPSC-SMP-ZHIPU Large Model Cross-Disciplinary Fund ZPCG20241119405 and Hong Kong RGC GRF No. 14206324.

Limitations

Our framework involves using GPT-3.5 to generate counterfactual augmented data. As we discussed in Appendix F, these samples may contain errors, but overall are beneficial to the training of our calibration network. The methods of constructing counterfactual augmented data using manual annotation or other methods remain to be explored. While our study primarily focuses on stance biases—particularly sentiment-stance spurious correlations and target preference bias—our approach can be extended to mitigate other forms of bias by designing appropriate counterfactual samples.

Ethics Statement

The datasets used in this paper are sourced from open-access datasets. The VAST dataset provides complete text data in open access. In compliance with the privacy agreement of Twitter for academic usage, the Sem16 and P-Stance were accessed using the official Twitter API[6] through the Tweet IDs to fetch complete text data. We removed the information on user privacy from the data. In these datasets, we analyze the biases and stereotypes in stance detection for some sensitive targets (e.g., belief, politics, etc.). We DO NOT critique any biases and stereotypes. We focus on analyzing their impacts on stance detection and mitigating these impacts. We used the counterfactual augmented data obtained from the GPT-3.5-Turbo API service from OpenAI. We followed their term and policies. Some examples in our paper may include a stance or tendency. It should be clarified that they are randomly sampled from the dataset for better studying the dataset and task, and do not represent any personal viewpoints.

References

Emily Allaway and Kathleen R. 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 2020, Online, November 16-20, 2020*, pages 8913–8931. Association for Computational Linguistics.

Pengyuan Chen, Kai Ye, and Xiaohui Cui. 2021. Integrating n-gram features into pre-trained model: A novel ensemble model for multi-target stance detection. In *Artificial Neural Networks and Machine Learning - ICANN 2021 - 30th International Conference on Artificial Neural Networks, Bratislava, Slovakia, September 14-17, 2021, Proceedings, Part III*, volume 12893 of *Lecture Notes in Computer Science*, pages 269–279. Springer.

Xiangjue Dong, Ziwei Zhu, Zhuoer Wang, Maria Teleki, and James Caverlee. 2023. Co^{2}pt: Mitigating bias in pre-trained language models through counterfactual contrastive prompt tuning. *CoRR*, abs/2310.12490.

Shalmoli Ghosh, Prajwal Singhania, Siddharth Singh, Koustav Rudra, and Saptarshi Ghosh. 2019. Stance detection in web and social media: A comparative study. In *Experimental IR Meets Multilinguality, Multimodality, and Interaction - 10th International Conference of the CLEF Association, CLEF 2019,*

[6]https://developer.twitter.com/en/docs/twitter-api

Lugano, Switzerland, September 9-12, 2019, Proceedings, volume 11696 of *Lecture Notes in Computer Science*, pages 75–87. Springer.

Gustavo Gonçalves and Emma Strubell. 2023. Understanding the effect of model compression on social bias in large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pages 2663–2675. Association for Computational Linguistics.

Zihao He, Siyi Guo, Ashwin Rao, and Kristina Lerman. 2023. Inducing political bias allows language models anticipate partisan reactions to controversies. *CoRR*, abs/2311.09687.

Zihao He, Negar Mokhberian, and Kristina Lerman. 2022. Infusing knowledge from wikipedia to enhance stance detection. In *Proceedings of the 12th Workshop on Computational Approaches to Subjectivity, Sentiment & Social Media Analysis, WASSA@ACL 2022, Dublin, Ireland, May 26, 2022*, pages 71–77. Association for Computational Linguistics.

Hu Huang, Bowen Zhang, Yangyang Li, Baoquan Zhang, Yuxi Sun, Chuyao Luo, and Cheng Peng. 2023. Knowledge-enhanced prompt-tuning for stance detection. *ACM Transactions on Asian and Low-Resource Language Information Processing*.

Myungha Jang and James Allan. 2018. Explaining controversy on social media via stance summarization. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval, SIGIR 2018, Ann Arbor, MI, USA, July 08-12, 2018*, pages 1221–1224. ACM.

David F. Jenny, Yann Billeter, Mrinmaya Sachan, Bernhard Schölkopf, and Zhijing Jin. 2023. Navigating the ocean of biases: Political bias attribution in language models via causal structures. *CoRR*, abs/2311.08605.

Ayush Kaushal, Avirup Saha, and Niloy Ganguly. 2021. twt-wt: A dataset to assert the role of target entities for detecting stance of tweets. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2021, Online, June 6-11, 2021*, pages 3879–3889. Association for Computational Linguistics.

Xiaochong Lan, Chen Gao, Depeng Jin, and Yong Li. 2023. Stance detection with collaborative role-infused llm-based agents. *CoRR*, abs/2310.10467.

Ang Li, Bin Liang, Jingqian Zhao, Bowen Zhang, Min Yang, and Ruifeng Xu. 2023. Stance detection on social media with background knowledge. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pages 15703–15717. 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, Online Event, August 1-6, 2021*, volume ACL/IJCNLP 2021 of *Findings of ACL*, pages 2355–2365. Association for Computational Linguistics.

Bin Liang, Qinglin Zhu, Xiang Li, Min Yang, Lin Gui, Yulan He, and Ruifeng Xu. 2022. Jointcl: A joint contrastive learning framework for zero-shot stance detection. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 81–91. Association for Computational Linguistics.

Tomasz Limisiewicz, David Marecek, and Tomás Musil. 2023. Debiasing algorithm through model adaptation. *CoRR*, abs/2310.18913.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692.

Yun Luo, Zhen Yang, Fandong Meng, Yafu Li, Jie Zhou, and Yue Zhang. 2023. An empirical study of catastrophic forgetting in large language models during continual fine-tuning. *CoRR*, abs/2308.08747.

Saif M. 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@NAACL-HLT 2016, San Diego, CA, USA, June 16-17, 2016*, pages 31–41. The Association for Computer Linguistics.

Dat Quoc Nguyen, Thanh Vu, and Anh Tuan Nguyen. 2020. Bertweet: A pre-trained language model for english tweets. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, EMNLP 2020 - Demos, Online, November 16-20, 2020*, pages 9–14. Association for Computational Linguistics.

Abel Salinas, Louis Penafiel, Robert McCormack, and Fred Morstatter. 2023. "im not racist but...": Discovering bias in the internal knowledge of large language models. *CoRR*, abs/2310.08780.

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 2020, Online, November 16-20, 2020*, pages 4222–4235. Association for Computational Linguistics.

Swapna Somasundaran and Janyce Wiebe. 2010. Recognizing stances in ideological on-line debates. In *Proceedings of the NAACL HLT 2010 Workshop on*

Computational Approaches to Analysis and Generation of Emotion in Text, pages 116–124, Los Angeles, CA. Association for Computational Linguistics.

Peter Stefanov, Kareem Darwish, Atanas Atanasov, and Preslav Nakov. 2020. Predicting the topical stance and political leaning of media using tweets. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 527–537. Association for Computational Linguistics.

Qingying Sun, Zhongqing Wang, Qiaoming Zhu, and Guodong Zhou. 2018. Stance detection with hierarchical attention network. In *Proceedings of the 27th International Conference on Computational Linguistics, COLING 2018, Santa Fe, New Mexico, USA, August 20-26, 2018*, pages 2399–2409. Association for Computational Linguistics.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurélien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. Llama: Open and efficient foundation language models. *CoRR*, abs/2302.13971.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V. Le, Ed H. Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2023. Self-consistency improves chain of thought reasoning in language models. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.

Jianhua Yuan, Yanyan Zhao, Yanyue Lu, and Bing Qin. 2022a. SSR: utilizing simplified stance reasoning process for robust stance detection. In *Proceedings of the 29th International Conference on Computational Linguistics, COLING 2022, Gyeongju, Republic of Korea, October 12-17, 2022*, pages 6846–6858. International Committee on Computational Linguistics.

Jianhua Yuan, Yanyan Zhao, and Bing Qin. 2022b. Debiasing stance detection models with counterfactual reasoning and adversarial bias learning. *CoRR*, abs/2212.10392.

Bowen Zhang, Daijun Ding, and Liwen Jing. 2022. How would stance detection techniques evolve after the launch of chatgpt? *CoRR*, abs/2212.14548.

Bowen Zhang, Daijun Ding, Liwen Jing, and Hu Huang. 2023a. A logically consistent chain-of-thought approach for stance detection. *CoRR*, abs/2312.16054.

Bowen Zhang, Xianghua Fu, Daijun Ding, Hu Huang, Yangyang Li, and Liwen Jing. 2023b. Investigating chain-of-thought with chatgpt for stance detection on social media. *CoRR*, abs/2304.03087.

Chujie Zheng, Hao Zhou, Fandong Meng, Jie Zhou, and Minlie Huang. 2023. Large language models are not robust multiple choice selectors. *CoRR*, abs/2309.03882.

Yuhang Zhou, Paiheng Xu, Xiaoyu Liu, Bang An, Wei Ai, and Furong Huang. 2023. Explore spurious correlations at the concept level in language models for text classification. *CoRR*, abs/2311.08648.

Qinglin Zhu, Bin Liang, Jingyi Sun, Jiachen Du, Lanjun Zhou, and Ruifeng Xu. 2022. Enhancing zero-shot stance detection via targeted background knowledge. In *SIGIR '22: The 45th International ACM SIGIR Conference on Research and Development in Information Retrieval, Madrid, Spain, July 11 - 15, 2022*, pages 2070–2075. ACM.

A Bias Measurement Analysis

The standard deviation of recalls (RStd) can largely mitigate the impact of label imbalance in the dataset and effectively assess model bias. This metric is commonly used to evaluate model bias and, as highlighted in relevant research (Zheng et al., 2023), "This measurement is intuitive that greater recall imbalance indicates more pronounced selection bias and is not as susceptible to label imbalance as the counting-based measurement."

To illustrate the robustness of this metric against label imbalance, we provide an example from our experiments with the Sem16 dataset. The ground truth data distribution is shown in Table 8. We observed a certain degree of imbalance in label distribution. Based on this, we sampled 200 instances in which ground truth is favor, none, and against, respectively on each sentiment. The sampled instances exhibited no imbalance. We calculated Bias-SSC on these sampled instances, obtaining a result of **27.34**, compared to **27.13** on the original dataset. Thus, we believe that the actual distribution of sentiment, target, and stance labels, does not significantly affect our measurement of stance detection bias in LLMs.

(%)	Positive	Neutral	Negative
Favor	9.96	1.99	12.79
None	7.52	2.96	15.32
Against	13.82	1.46	34.19

Table 8: The ground truth data distribution of the Sem16 dataset.

B Bias Influence

Bias represents a macro-level error pattern. In our experiments, the macroscopic results indicate that larger bias-SSC and bias-TPB tend to lead to poorer stance detection results. To prove this, we conducted a comprehensive analysis involving the calculation of Pearson correlation coefficients between bias-SSC and stance detection F1

Bias Type	Correlation Coefficient	p-value
bias-SSC	-0.4047	3.9e-05
bias-TPB	-0.5323	2.0e-08

Table 9: Results of Pearson correlation coefficients and p-value between bias-SSC and stance detection F1 scores, as well as bias-TPB and stance detection F1 scores.

scores, as well as bias-TPB and stance detection F1 scores, across 97 groups derived from Tables 1-2, 6, 7, 10, 11-15 (with bias-SSC and bias-TPB in Tables 6, 7, 10 and their corresponding F1 values in Tables 4, 5, and bias-SSC and F1 or bias-TPB and F1 values in Tables 11-15). The results are shown in Table 9, which indicate significant negative correlations for bias-SSC and stance detection F1, and for bias-TPB and stance detection F1.

C Prompts Setting

We present the prompt templates used in Section 3.2 and Section 4.2.

Specifically, Figure 5 shows the prompt template we use with GPT-4 to obtain sentiment labels. Figures 6, 7, and 8 show the prompt templates corresponding to the Task-Des, CoT-Demo, and Debias-Instruct prompt settings in Section 3.2.1, respectively. Similarly, Figures 6, 7, and 9 display the prompt templates corresponding to the Task-Des, CoT-Demo, and Debias-Instruct prompt settings in Section 3.2.2, respectively.

Figures 10 and 11 illustrate the prompt templates used to obtain non-causal and causal counterfactual augmented data, respectively, as discussed in Section 4.2. Figure 10 presents the constructed instruction for acquiring non-causal counterfactual augmented data, while Figure 11 shows the instruction for obtaining causal counterfactual augmented data.

D Experimental Result of LLMs Bias

Table 10 shows the experimental results of the ablation study on the P-Stance dataset, which are consistent with our previous observations. We present the complete experimental results in Section 3. Tables 11, 12, and 13 show the RStd and the macro F1-Score of samples with different sentiment on the Sem16, P-Stance, and VAST datasets. Tables 14 and 15 present the RStd and the macro F1-Score of samples with different stance target on the Sem16, P-Stance, and VAST datasets. We can observe that in most cases, a larger stance bias leads to poorer

	P-Stance		
	Avg↑	SSC↓	TPB↓
$\text{FACTUAL}_{\text{LLaMA-2}}$	**73.82***	**9.61**	**5.52***
w/o Calibration	67.96	27.52	27.56
w/o CAD	71.61	15.43	12.07
- w/o non-causal	71.29	13.12	13.25
- w/o causal	69.39	10.66	14.82
$\text{FACTUAL}_{\text{GPT-3.5}}$	**76.16***	11.31	**7.74***
w/o Calibration	73.32	18.08	13.47
w/o CAD	70.39	12.92	11.83
- w/o non-causal	74.08	14.36	11.80
- w/o causal	71.71	**10.22**	10.21

Table 10: Experimental results of ablation study of in-target stance detection on the P-Stance dataset. The best scores are in bold. Results with $\star$ indicate the significance tests of our FACTUAL over the ablation experiments at p-value < 0.05.

stance detection results. In Tables 11, 12, and 13, samples with positive and negative emotions exhibited larger Rstd, indicating that sentiment influenced the stance judgment of LLMs as a bias pattern. In Tables 14 and 15, on some controversial debate topics such as the "Legalization of Abortion", the "Feminist Movement", and specific individuals like "Donald Trump" and "Hillary Clinton", larger Rstd indicates that LLMs demonstrated a relatively large target preference bias.

E Prompt Robustness

We conducted experiments, and demonstrating that our method is robust across different prompt templates. The results are shown in Table 16, and the prompts used are listed in Table 17. According to experimental results, the variance in direct stance inference through prompts (w/o Calibration) is 3.5454 for LLaMA-2 and 10.5929 for GPT-3.5-Turbo. In contrast, the variance for stance judgments output by our FACTUAL is 0.3766 for LLaMA-2 and 1.1055 for GPT-3.5-Turbo. We believe this robustness stems from the fact that our calibration network uses LLM-generated rationales to analyze the stance of samples.

F Human Evaluation of Counterfactual Augmented Data

We randomly select 500 samples and use human evaluation (with three experienced researchers who are not involved in this work and have worked on natural language processing for over 3 years) to measure the counterfactual data generated by GPT-3.5-turbo. The primary consideration focuses on the qualitative assessment of the generated sam-

```
[sentence]: {sentence}
What is the sentiment of [sentence]?
Only answer with "positive", "negative" or "neutral".
```

Figure 5: Prompt template of sentiment labels annotation by GPT-4. Fill the blue text with the corresponding text from the sample.

```
Stance detection is to determine the attitude or tendency towards a certain
target through a given sentence, including favor, against and neutral.

{sentence}

Question: What is the attitude of the sentence toward "{target}"? Please
select the correct answer from "favor", "against" and "neutral".
Answer this question with JSON format:
```json
{{
    "stance": "favor" | "against" | "neutral",
}}
```
```

Figure 6: Prompt template with Task-Des setting. We first outline the stance detection task, then instruct the LLMs to determine the stance based on the sentence in relation to the target. Fill the blue text with the corresponding text and target from the sample.

ples, necessitating evaluators to confirm the accuracy of both the grammar and the affirmed stance. The secondary consideration pertains to achieving generating objectives, necessitating evaluators to confirm if the samples were generated as guideline instructions. Evaluators respond to these considerations with a binary "yes" or "no". Subsequently, we calculate the average ratio of affirmative responses from three evaluators for each query. The results in Table 18 show that the generated samples are of high quality, contributing substantially to our calibration network training.

G Case Study

We conduct a case study on Sem16, P-Stance and VAST datasets, to analyze the biases of LLMs in the stance detection task and the practical effectiveness of our calibration network. The results are show in Table 19, 20 and 21. The correct analysis patterns of LLMs are marked in blue, while biased analysis patterns are marked in red. We can observe that for some samples with strong sentiment expressions, such as the examples in Table 20, LLMs are influenced by sentiment Spurious cues and result in biased stance judgments. For some controversial debate topics, such as the examples in Table 19, LLMs generate hallucinations due to their preferences, leading to biased stance judgments.

```
Stance detection is to determine the attitude or tendency towards a certain
target through a given sentence, including favor, against and neutral.
**Please read the following examples carefully and use them as references to
judge the attitude of the sentence towards the target.**

[in-context examples]

Your sentence:

{sentence}

Question: What is the attitude of the sentence toward "{target}"? Please
select the correct answer from "favor", "against" and "neutral".
Answer this question with JSON format:
```json
{{
    "answer": "your answer",
    "stance": "favor" | "against" | "neutral"
}}
```
```

Figure 7: Prompt template with CoT-Demo setting. We randomly select 4 samples from the training set, provide the ground truth stance labels, and guide GPT-4 to generate chain-of-thought rationales as examples for this prompt. Fill the green text with constructed examples, and fill the blue text with the corresponding text and target from the sample.

	Sem16(%)					
	Positive		Neutral		Negative	
	RStd↓	F1↑	RStd↓	F1↑	RStd↓	F1↑
LLaMA-2-70b-chat						
Task-Des	**22.20**	**59.73**	**23.65**	58.11	7.56	65.19
CoT-Demo	26.15	58.24	27.88	55.18	28.54	62.04
Debias-Instruct	26.86	55.04	25.51	**58.20**	**5.34**	**68.61**
GPT-3.5-Turbo-0125						
Task-Des	29.37	48.37	25.22	58.78	26.80	55.44
CoT-Demo	**16.02**	**65.77**	25.47	**61.69**	**12.74**	**70.69**
Debias-Instruct	30.52	46.09	**16.31**	60.07	24.43	54.91

Table 11: RStd of sentiment labels and macro F1-score of stance detection on Sem16 dataset. The best scores are in bold.

```
Stance detection is to determine the attitude or tendency towards a certain
target through a given sentence, including favor, against and neutral.
**Note that the sentiment of the sentence is not necessarily consistent with
the author's attitude on the target, and avoid directly using emotion as the
only basis for judging the attitude.**

{sentence}

Question: What is the attitude of the sentence toward "{target}"? Please
select the correct answer from "favor", "against" and "neutral".
Answer this question with JSON format:
```json
{{
    "stance": "favor" | "against" | "neutral",
}}
```
```

Figure 8: Prompt template with SSC Debias-Instruct setting. We add explicit debiasing instructions following the task description. Fill the blue text with the corresponding text and target from the sample.

```
Stance detection is to determine the attitude or tendency towards a certain
target through a given sentence, including favor, against and neutral. **Be
careful to only judge the author's attitude on the target based on the
content in the sentence, and do not include your inherent attitude towards
the target.**

{sentence}

Question: What is the attitude of the sentence toward "{target}"? Please
select the correct answer from "favor", "against" and "neutral".
Answer this question with JSON format:
```json
{{
    "stance": "favor" | "against" | "neutral",
}}
```
```

Figure 9: Prompt template with TPB Debias-Instruct setting. We add explicit debiasing instructions following the task description. Fill the blue text with the corresponding text and target from the sample.

```
[sentence]: {sentence}
[target]: {target}

The [sentence] expresses a {stance} stance to the [target]. Please rephrase
the [sentence] using different words and emotions, and rewrite the [target]
using different words while preserving the same meaning and stance as the
original.
```

Figure 10: Prompt template that allows the LLMs to rephrase the original sentence with different words and sentiments and express the target while ensuring that the semantics and the stance of the perturbed sample towards the target remain unchanged. Fill the blue text with the corresponding text, target, and stance label from the sample.

```
[sentence]: {sentence}
[target]: {target}

The [sentence] expresses a {stance} attitude to the [target]. Please make
minimal changes to the [sentence] to express a reverse attitude to the
[target].
```

Figure 11: Prompt template that makes necessary modifications to reverse the applicability of the label. Fill the blue text with the corresponding text, target, and stance label from the sample.

	P-Stance(%)					
	Positive		Neutral		Negative	
	RStd↓	F1↑	RStd↓	F1↑	RStd↓	F1↑
LLaMA-2-70b-chat						
Task-Des	33.32	**68.42**	19.98	62.68	**16.77**	72.97
CoT-Demo	**32.62**	67.17	**15.10**	**65.00**	20.70	**73.61**
Debias-Instruct	34.66	66.49	22.81	59.14	17.10	71.82
GPT-3.5-Turbo-0125						
Task-Des	33.79	**68.61**	23.48	62.32	**13.88**	**75.66**
CoT-Demo	**32.39**	66.99	**19.25**	**65.09**	16.60	74.44
Debias-Instruct	33.56	68.00	20.05	63.19	16.83	75.07

Table 12: RStd of sentiment labels and macro F1-score of stance detection on P-Stance dataset. The best scores are in bold.

	VAST(%)					
	Positive		Neutral		Negative	
	RStd↓	F1↑	RStd↓	F1↑	RStd↓	F1↑
LLaMA-2-70b-chat						
Task-Des	**30.19**	**61.47**	**5.79**	60.85	14.63	66.97
CoT-Demo	36.20	59.60	18.93	**67.87**	**12.51**	63.87
Debias-Instruct	34.60	56.63	10.82	59.49	13.48	**67.61**
GPT-3.5-Turbo-0125						
Task-Des	36.33	44.58	19.75	52.72	30.03	46.74
CoT-Demo	**26.01**	**69.05**	**8.78**	**66.73**	**14.17**	**67.38**
Debias-Instruct	38.40	40.96	23.51	49.01	29.66	46.19

Table 13: RStd of sentiment labels and macro F1-score of stance detection on VAST dataset. The best scores are in bold.

	Sem16(%)									
	HC		FM		LA		A		CC	
	RStd↓	F1↑	RStd↓	F1↑	RStd↓	F1↑	RStd↓	F1↑	RStd↓	F1↑
LLaMA-2-70b-chat										
Task-Des	11.59	73.64	24.41	58.33	**6.84**	59.36	**15.12**	48.94	29.98	60.15
CoT-Demo	28.36	64.71	35.98	52.96	19.27	57.10	19.91	**55.87**	34.26	62.78
Debias-Instruct	**8.73**	**76.50**	**14.75**	**63.39**	9.77	**58.64**	25.76	39.32	**22.83**	**69.16**
GPT-3.5-Turbo-0125										
Task-Des	28.68	61.97	24.21	57.47	26.98	56.75	5.81	27.54	27.52	60.36
CoT-Demo	**20.99**	**74.12**	**15.16**	**65.61**	**11.12**	**64.74**	6.75	**57.89**	**13.30**	**75.57**
Debias-Instruct	28.07	63.07	27.81	54.72	24.08	57.11	**5.43**	29.14	23.94	62.63

Table 14: RStd of targets and macro F1-score of stance detection on Sem16 dataset. The best scores are in bold.

	P-Stance(%)						VAST(%)	
	JB		BS		DT		ALL	
	RStd↓	F1↑	RStd↓	F1↑	RStd↓	F1↑	RStd↓	F1↑
LLaMA-2-70b-chat								
Task-Des	2.69	84.31	8.79	77.29	**15.79**	78.08	7.76	68.36
CoT-Demo	7.91	**85.03**	**7.62**	**79.77**	19.19	77.52	9.64	67.08
Debias-Instruct	**1.58**	82.63	9.29	75.10	15.96	**78.36**	**4.86**	**69.10**
GPT-3.5-Turbo-0125								
Task-Des	**0.53**	**83.20**	4.82	**80.02**	**10.94**	81.66	28.44	49.86
CoT-Demo	3.22	83.07	**4.01**	77.98	12.59	81.59	**8.40**	**69.90**
Debias-Instruct	0.63	82.91	5.36	79.01	11.39	**82.85**	26.77	51.66

Table 15: RStd of targets and macro F1-score of stance detection on P-Stance and VAST dataset. The best scores are in bold.

	Sem16 (LLaMA-2-70b-chat)			Sem16 (GPT-3.5-Turbo-0125)		
	Avg	Bias-SSC	Bias-TPB	Avg	Bias-SSC	Bias-TPB
FACTUAL-Prompt-1	**73.82**	**9.61**	**5.52**	**76.16**	11.31	**7.74**
w/o Calibrition	67.96	27.52	27.56	73.32	18.08	13.47
w/o CAD	71.61	15.43	12.07	70.39	12.92	11.83
- w/o CAD-non-causal	71.29	13.12	13.25	74.08	14.36	11.80
- w/o CAD-causal	69.39	10.66	14.82	71.71	**10.22**	10.21
FACTUAL-Prompt-2	**72.48**	**9.99**	**7.69**	**74.73**	**10.86**	**8.77**
w/o Calibrition	66.97	26.92	23.75	69.50	22.22	15.88
w/o CAD	70.51	19.26	13.58	70.49	19.89	12.40
- w/o CAD-non-causal	71.00	11.27	13.27	72.01	15.51	10.05
- w/o CAD-causal	69.89	10.03	11.57	70.81	11.58	10.56
FACTUAL-Prompt-3	**73.74**	9.19	**5.60**	**73.59**	10.11	**7.77**
w/o Calibrition	67.37	19.45	18.14	65.35	26.35	25.88
w/o CAD	69.99	12.83	13.15	67.86	16.84	18.24
- w/o CAD-non-causal	72.65	9.47	15.46	71.51	13.70	16.03
- w/o CAD-causal	69.90	**8.11**	10.27	69.60	**9.53**	13.16

Table 16: Experimental results of three different prompt templates on in-target stance detection on the Sem16 dataset. The best scores are in bold.

Task Description	Stance detection is to determine the attitude or tendency towards a certain target through a given sentence, including favor, against, and neutral. **Please read the following examples carefully and use them as references to judge the attitude of the sentence towards the target.**
Prompt #1	[Task Description]\n \n[in-context examples]\n Your sentence: sentence\n Question: What is the attitude of the sentence toward "target"? Please select the correct answer from "favor", "against" and "neutral". Answer this question with JSON format: “‘json "answer": "your answer", "stance": "favor" | "against" | "neutral" “‘
Prompt #2	[Task Description]\n\n[in-context examples]\n Give you [sentence] and [target], please judge the attitude of [sentence] toward [target]. Select the correct answer from favor, against and neutral.\n [sentence]: sentence\n [target]: target\n Only select the correct answer from "favor", "against" and "neutral".\n Answer this question with JSON format: “‘json "answer": "your answer", "stance": "favor" | "against" | "neutral" “‘
Prompt #3	[Task Description]\n\n[in-context examples]\n What is the attitude of [sentence] toward [target]?\n A.against\n B.favor\n C.neutral\n[sentence]: sentence\n[target]: target\n Answer this question with JSON format: “‘json "answer": "your answer", "stance": "A.against" | "B.favor" | "C.neutral" “‘

Table 17: Three different prompts used in the experiment in Table 16.

	CAD-non-causal		CAD-causal	
	Quality	Achievement	Quality	Achievement
Human Eval	96.67%	91.33%	95.33%	88.83%

Table 18: Human evaluation on Sem16, P-Stance, and VAST datasets. Quality stands for qualitative assessment of the generated samples and Achievement stands for the achievement of generating objectives.

Text (Target: Legalization of Abortion)	Label	LLMs Pred	LLMs Stance Rationale	Calibration
A prochoice advocate but circumcise ur baby? Fucking hypocrite! #circumcision #humanrights	Against	Favor	The sentence criticizes the hypocrisy of being a prochoice advocate but supporting circumcision, indicating a negative attitude towards circumcising babies. Since the author is prochoice, it can be inferred that they are in favor of the legalization of abortion.	Against
Text (Target: Feminist Movement)	**Label**	**LLMs Pred**	**LLMs Stance Rationale**	**Calibration**
Equality is the police burying a domestic violence accusation against a female sports star, too #wedidit #usa	Against	Favor	The sentence mentions burying a domestic violence accusation against a female sports star, which implies a negative view towards this unequal treatment. Therefore, the attitude of the sentence towards the Feminist Movement is likely favorable.	Against

Table 19: Case study on the Sem16 dataset.

Text (Target: Donald Trump)	Label	LLMs Pred	LLMs Stance Rationale	Calibration
You might as well put a sock in it. Voters see these coordinated stupid attacks are like your #Kavanaugh symphony. #Dems can't win with hollow attempts to bash #Trump.	Favor	Against	The sentence uses derogatory terms like 'stupid attacks' and 'hollow attempts to bash #Trump,' indicating a critical perspective. Therefore, the attitude of the sentence toward 'Donald Trump' is against.	Favor
Text (Target: Joe Biden)	Label	LLMs Pred	LLMs Stance Rationale	Calibration
Saying that is as absurd as saying breast cancer screening increases breast cancer. Is there any Republican who isn't as dumb as dirt?	Favor	Against	The sentence implies negative assumptions about the intelligence of Republicans. The mention of Joe Biden in the context of criticizing Republicans suggests a critical or skeptical attitude towards Joe Biden.	Favor

Table 20: Case study on the P-Stance dataset.

Text (Target: Homosexual)	Label	LLMs Pred	LLMs Stance Rationale	Calibration
Yet that same Bible gives us Christians a particular anthropology, and reveals to us the meaning of sex and marriage within the cosmic order. And herein lies the problem. If you look to a "holy" book written long, long ago and betraying its human origins on every page for guidance on issues of cosmology, biology, sexuality and other human and natural sciences, you will inevitably be in constant conflict with social progress and, indeed, reality. There is no secular opposition to homosexuality."	Favor	Against	The sentence criticizing the use of a 'holy' book for guidance on issues related to sexuality and natural sciences. It suggests that relying on such sources will lead to conflict with social progress and reality. Additionally, the statement asserts that there is no secular opposition to homosexuality, implying a negative view towards those who oppose it based on religious beliefs. Overall, the tone of the sentence is against homosexuality, indicating a critical stance.	Favor
Text (Target: Gun Control)	Label	LLMs Pred	LLMs Stance Rationale	Calibration
Military and police, yes. They're sworn and trained to uphold the country and the laws of the country. The rest is an ego trip.	Favor	Against	The sentence expresses a negative attitude towards gun control in the military and police, stating that they are sworn and trained to uphold the country and its laws. However, it is critical of the use of guns by others, referring to it as an ego trip. Therefore, the attitude of the sentence towards gun control is against	Favor

Table 21: Case study on the VAST dataset.