Diagnosing Generalization in Open-Source LLMs for Stance Detection

Parush Gera
University of South Florida
parush@usf.edu

Tempestt Neal
University of South Florida
tjneal@usf.edu

Abstract

Stance detection identifies whether a text expresses support, opposition, or neutrality toward a target and is central to applications such as political analysis and misinformation monitoring. With the shift toward large language models (LLMs), stance classification increasingly relies on prompting and lightweight adaptation. Yet the generalization behavior of open-source LLMs across new targets and domains remains uneven. We conduct a large-scale diagnostic study of four open-source LLMs (3B–24B parameters), examining how model size, prompting strategies, and Low-Rank Adaptation (LoRA) interact across in-target, cross-target, and cross-domain settings. Across 912 experiments, three patterns emerge: (1) larger models improve prompting-based in-target performance, but this advantage diminishes after fine-tuning; (2) LoRA boosts in-target accuracy yet often harms cross-context transfer; (3) optimal prompting depends on model size. These results reveal a consistent tension between specialization and generalization, offering practical guidance for configuring LLM-based stance detection under transfer.

1 Introduction

Public discourse evolves rapidly: new political candidates emerge, public health crises unfold, and controversial policies shift overnight. Systems that analyze public opinion must therefore do more than classify familiar topics—they must adapt to new targets and domains as they arise. Stance detection, the task of identifying whether an author expresses support, opposition, or neutrality toward a target, has become central to this effort (Gera and Neal, 2025; Küçük and Can, 2020). Stance detection sits within a broader family of opinion-oriented NLP tasks—including aspect extraction and sentiment analysis, that share the underlying challenge of identifying what opinions are expressed toward which entities in text (Chaudhary and Neal, 2026).

To capture this challenge, prior work distinguishes between in-target stance detection (SD), where models are trained and tested on the same target; cross-target stance detection (SD_{CT}), which evaluates transfer from a seen target to an unseen one; and cross-domain stance detection (SD_{CD}), which requires generalization across targets in contextually dissimilar domains. Although SD_{CT} and SD_{CD} better reflect real-world deployment, they expose a persistent limitation: many models rely on lexical or domain-specific cues that fail to transfer, leading to substantial performance drops on unseen or out-of-domain data (Hardalov et al., 2021; Conforti et al., 2020).

To overcome these transfer limitations, prior work has explored increasingly sophisticated architectures, from early RNN- and CNN-based models to transformer encoders such as BERT and its variants (Xu et al., 2018; Liang et al., 2022; Ghosh et al., 2019; Wei and Mao, 2019; Zhang et al., 2017). Other approaches inject external knowledge—via Wikipedia passages or knowledge graphs—to supply richer target context (He et al., 2022; Zhang et al., 2024b). While these methods improve performance within specific settings, they do not eliminate the underlying fragility of cross-target and cross-domain transfer. Performance often degrades under substantial domain shift, and retraining separate models for each emerging topic is both costly and impractical given the rapid evolution of public discourse and the scarcity of large-scale labeled data (Gera and Neal, 2025; Küçük and Can, 2020).

The rise of generative large language models (LLMs) introduces a different paradigm for stance detection. Rather than training task-specific architectures, LLMs leverage large-scale pretraining and are adapted through prompting or lightweight fine-tuning, potentially enabling broader transfer across targets and domains. Prompting techniques such as Chain-of-Thought (CoT) further aim to elicit

*Proceedings of the 15th Joint Conference on Lexical and Computational Semantics (*SEM 2026)*, pages 317–330

structured reasoning from these models (Wei et al., 2022; Ma et al., 2025). While proprietary systems have demonstrated strong performance, access restrictions limit their use in academic research (Isaev et al.). This has driven growing interest in smaller, open-source LLMs, raising a critical question: *can these accessible models provide robust stance detection across evolving targets and domains?*

Recent work shows promise but remains fragmented. Existing studies have either focused on in-target settings only (Cruickshank and Ng, 2025; Ng et al., 2025), or prioritized dataset construction over systematic analysis of trade-offs between model scale, prompt complexity, and fine-tuning (Yuan et al., 2025). The effectiveness of zero-shot prompting across model sizes and domains is also underexplored (Schulhoff et al., 2024). Thus, unknowns remain regarding using open-source LLMs for stance detection, especially across contexts.

To address these gaps, we conduct a controlled diagnostic study of open-source LLMs across in-target (SD), cross-target (SD_{CT}), and cross-domain (SD_{CD}) settings. We evaluate four instruction-tuned models ranging from under 3B to 24B parameters across four widely used stance datasets spanning political, public health, and corporate domains. Our analysis systematically varies six prompting strategies—including zero-shot, few-shot, knowledge-infused, and Chain-of-Thought (CoT)—as well as parameter-efficient fine-tuning via Low-Rank Adaptation (LoRA) (Hu et al., 2022). In total, we perform 912 experiments to isolate how model size, prompting design, and adaptation interact under transfer. This study is guided by the following research questions:

- **RQ1:** How does model size shape generalization behavior across in-target, cross-target, and cross-domain stance detection settings?
- **RQ2:** How do different prompting strategies affect performance and transfer robustness across these settings?
- **RQ3:** How does parameter-efficient fine-tuning via LoRA alter the balance between in-target performance and cross-context generalization?

To our knowledge, this is the first study to systematically analyze how model size, prompt design, and parameter-efficient fine-tuning interact in open-source LLMs across both in-target and cross-context stance detection settings.[1]

2 Related Work

Stance detection gained broad visibility with SemEval 2016 Task 6 (Mohammad et al., 2016), which formalized in-target and cross-target evaluation and catalyzed a shift from feature-engineered pipelines to neural models. Subsequent work improved cross-target transfer using bidirectional conditional encoding (Augenstein et al., 2016), self-attention (Xu et al., 2018), adversarial learning (Hardalov et al., 2021), and topic modeling (Gómez-Suta et al., 2023). To mitigate cross-domain performance drops, researchers have employed contrastive learning with counterfactual data generation (Kim et al., 2025) and synthetic data for open-domain detection (Wagner et al., 2024). Another line of work improves generalization by injecting external knowledge through Wikipedia passages (He et al., 2022), knowledge graphs (Zhang et al., 2022), or retrieval-augmented LLMs (Zhu et al., 2025), though recent work cautions that imprecise retrieval can introduce bias (Nguyen and Kim, 2025). Despite these advances, generalization to new targets or domains remains difficult, with prior work consistently reporting substantial performance degradation (Gera and Neal, 2025; Alturayeif et al., 2023; Jamadi Khiabani and Zubiaga, 2025).

The advent of LLMs has shifted the paradigm from specialized architectures to prompting and adapting general-purpose foundation models (Cruickshank and Ng, 2025). Prompt-based approaches—including zero-shot prompting, where the model receives only task instructions without labeled examples, and few-shot prompting, where a small number of in-context labeled examples are provided—enable models to perform stance classification without updating model parameters. Chain-of-Thought (CoT) prompting encourages models to articulate reasoning through intermediate steps and has been shown to improve LLM performance in stance-related tasks (Gatto et al., 2023; Weinzierl and Harabagiu, 2024), while Chain-of-Stance decomposes prediction into sequential stance-relevant assertions (Ma et al., 2025). However, comprehensive evaluations reveal that performance remains highly sensitive to prompt phrasing and structure (Sclar et al., 2024; Kim et al., 2025; Zhuo et al., 2024). This body of work primarily evaluates prompting on in-target tasks and relies on closed-source models, leaving the effectiveness of differ-

[1] Code, prompts, and LoRA configurations are available at https://github.com/parushgera/llm_stance.

ent strategies on open-source LLMs across cross-context settings underexplored (Cruickshank and Ng, 2025).

While prompting has enabled strong zero- and few-shot stance classification without parameter updates (Gambini et al., 2024), its performance remains sensitive to prompt design. As a result, researchers have explored parameter-efficient fine-tuning (PEFT)—that is, adapting pretrained models by updating a small subset of parameters using labeled task data—as a more stable alternative. Among these, Low-Rank Adaptation (LoRA) (Hu et al., 2022) has emerged as a popular approach, injecting compact trainable matrices into the model architecture while keeping most weights fixed. However, the effectiveness of LoRA for stance detection remains unsettled: some studies suggest that carefully crafted zero-shot prompts can outperform LoRA-tuned models, particularly under cross-domain transfer (Sclar et al., 2024; Kim et al., 2025; Zhuo et al., 2024). To address these gaps, we systematically analyze how LoRA compares to advanced prompting strategies, how each approach generalizes to unseen targets and domains, and how model scale influences performance.

3 Methodology

3.1 Task Formulation

We frame stance detection as a constrained classification task: given a text sample x and a fixed label set $\mathcal{Y}$, predict a single label $y \in \mathcal{Y}$. Following prior work, $\mathcal{Y}$ typically consists of the three canonical stance categories, $\mathcal{Y} = \{favor,\ against,\ None\}$. For datasets that do not include a neutral category (e.g., P-Stance), we adopt the reduced binary label set $\mathcal{Y} = \{favor,\ against\}$, while maintaining consistency across experiments by aligning predictions to the labels available in each dataset. We consider 13 targets and two domains, as described in Table 1, evaluated across three settings:

- *In-Target Stance Detection (SD):* A stance detection model is trained and tested on data from the same target or domain.
- *Cross-Target Stance Detection (SD_{CT}):* A stance detection model is trained on a source target (Tr_{src}) and evaluated on a different destination target (Te_{dest}). Consistent with prior work, we adopt cross-target label pairs, specifically { Donald Trump (dt) ↔ Hillary Clinton (hc), Feminist Movement (fm) ↔ Legalization of Abortion (la) } (Xu et al., 2018; Liang et al., 2022). These reciprocal pairs have become the standard benchmark setup for SD_{CT}.
- *Cross-Domain Stance Detection (SD_{CD}):* A stance detection model is trained on a source domain (δ_{src}) and evaluated on a different destination domain (δ_{dest}). Following prior work, we adopt the reciprocal cross-domain label pair { Entertainment (ent) ↔ Healthcare (hlt) }, which has been highlighted as a particularly challenging benchmark for SD_{CD} (Conforti et al., 2020).

3.2 Datasets

We use four publicly available datasets, which are summarized in Table 1:

- **SemEval-2016 Task 6** (Mohammad et al., 2016): Tweets labeled *favor*, *against*, or *none* toward six sociopolitical targets.
- **P-Stance** (Li et al., 2021): Tweets labeled *favor* or *against* toward three U.S. political figures.
- **COVID-19-Stance** (Glandt et al., 2021): Tweets labeled *favor*, *against*, or *none* on COVID-19-related topics.
- **Will They Won't They (WT-WT)** (Conforti et al., 2020): Tweets about mergers and acquisitions in the healthcare and entertainment domains, labeled *support*, *refute*, *comment*, or *unrelated*.

Dataset	Target or Domain	Train	Test
COVID-19	face_masks (face)	1,507	200
	fauci	1,664	200
	school_closures (school)	990	200
	stay_at_home_orders (stay)	1,172	200
	Total	**5,333**	**800**
P-Stance	Bernie Sanders (bernie)	5,690	635
	Donald Trump (dtp)	7,176	777
	Joe Biden (joe)	6,551	745
	Total	**19,417**	**2,157**
SemEval-2016	Atheism (at)	513	220
	Climate Change is a Concern (cc)	395	169
	Donald Trump (dt)	530	177
	Feminist Movement (fm)	664	285
	Hillary Clinton (hc)	689	295
	Legalization of Abortion (la)	653	280
	Total	**3,444**	**1,426**
Will They Won't They	Healthcare (ent)*	22,101	7,367
	Entertainment (hlt)*	11,141	3,714
	Total	**33,242**	**11,081**

Table 1: Dataset statistics with train/test splits. Parentheses indicate target abbreviations; * denotes cross-domain settings.

Both reciprocal cross-target pairs (dt ↔ hc and fm ↔ la) are drawn from the SemEval-2016 dataset, which uses a uniform label space {favor,

against, none} across all targets, so no cross-dataset label alignment is required for SD_{CT}. We adopt these specific pairs in line with prior cross-target stance detection studies (Xu et al., 2018; Liang et al., 2022), which have established them as the standard benchmark setup. Using the same pairs ensures a fair and directly comparable evaluation against existing work. We also note that 'dt' and 'dtp' refer to distinct targets in different datasets (SemEval-2016 and P-Stance, respectively) and only 'dt' is used in the cross-target evaluation.

3.3 Models

We evaluate four publicly available, instruction-tuned language models spanning small to mid-scale parameter sizes. Unlike proprietary large-scale systems, these models are computationally accessible and suitable for academic and resource-constrained deployment. This allows us to systematically investigate performance scaling and practical trade-offs for stance detection under real-world resource constraints. The specific models are: **Phi-3-mini-128k-instruct (Phi)**, Microsoft's 3.8B model (Zha et al., 2024); **Mistral-7B-Instruct-v0.3 (M7B)**, Mistral AI's 7B model (Jiang et al., 2023); **Llama-3-8B-Instruct (L3-8B)**, Meta's 8B model (Grattafiori et al., 2024); and **Mistral-Small-Instruct-2409 (M24B)**, Mistral AI's ~24B model (AI, 2024).

3.4 Experimental Design

In total, we conducted 912 experiments:

- **In-Target / In-Domain (Vanilla SD)**: 13 targets and 2 domains were evaluated using six prompting strategies across four models, each tested in both base and LoRA-tuned conditions, for a total of $15 \times 6 \times 4 \times 2 = 720$ experiments.
- **SD_{CT}**: Two reciprocal target pairs: (*hc* $\leftrightarrow$ *dt*, *fm* $\leftrightarrow$ *la*), yielding four transfer directions. With four prompts, all four models, and both base and LoRA-tuned conditions, this yielded $4 \times 4 \times 4 \times 2 = 128$ experiments.
- **SD_{CD}**: One reciprocal domain pair: (*ent* $\leftrightarrow$ *hlt*), yielding two transfer directions. With four prompts, four models, and both base and LoRA-tuned conditions, this yielded $2 \times 4 \times 4 \times 2 = 64$ experiments.

Each experiment corresponds to its individual evaluation run on the test split of the respective target or domain, producing one macro-F_1 score. For few-shot prompting, in-context examples were drawn exclusively from the training split. We report macro-F_1, which equally weights all stance classes. All experiments were conducted on NVIDIA H100 GPUs; fine-tuning the 24B model required two GPUs, while all other runs used one.

3.4.1 Prompting Strategies

We evaluated six prompting strategies: **(1) Zero-Shot (P^{ZS}):** a minimal instruction defining "stance" and the label set, asking for a single label; **(2) Few-Shot (P^{FS}):** augmented with k=5 labeled examples per stance label prior to the query instance; **(3) Knowledge-Infused (P^{KI}):** a target/domain information block prepended; **(4) Chain-of-Thought (P^{CoT}):** instructions to perform step-by-step internal reasoning and emit only the final label; **(5) CoT + Knowledge (P^{CoT+KI}):** combines background knowledge with internal reasoning; **(6) CoT + Knowledge + Few-Shot ($P^{CoT+KI+FS}$):** combines knowledge, k=5 labeled examples, and step-by-step reasoning.

For few-shot prompting, the $k = 5$ in-context examples per stance label were sampled randomly from the source training split and held fixed across runs for a given target to ensure reproducibility. For knowledge-infused prompting, the target/domain information block consisted of a brief, factual description of the target (e.g., who the political figure is, what the policy entails, or what the merger involved), and was held constant across all models. Prompt examples are provided in the Appendix(Listings 1–6); the full prompts used in our experiments expand the example block with $k = 5$ examples per label as described.

All prompts followed each model's tag-based instruction format, beginning with a preamble directing the model to act as an expert stance detector and noting that the task was for research only given the inclusion of sensitive content (e.g., political figures, elections, abusive language). Prompts enumerated valid stance labels and required outputs in the format `{label: stance_name}`. All six strategies were applied in SD; in SD_{CT} and SD_{CD}, only four (CoT+Knowledge, Knowledge-Infused, Few-Shot, and CoT+Knowledge+Few-Shot) were used, as these explicitly transfer knowledge or exemplars from source to destination, with inference restricted to destination examples.

3.4.2 LoRA Fine-Tuning

We evaluate models in their base (instruction-tuned) form and after PEFT using LoRA (Hu et al., 2022). LoRA inserts small trainable matrices into selected

weight updates while keeping the original model parameters frozen, enabling task-specific adaptation at substantially lower cost than full fine-tuning. Fine-tuning was performed separately for each of the 13 targets and two domains using their respective training splits. For evaluation, SD used the target-specific adapter, whereas SD_{CT} and SD_{CD} applied the adapter trained on the source target or domain and evaluated it on the corresponding destination data.

We include LoRA: (1) to quantify performance gains over zero- and few-shot prompting, and (2) to examine how access to labeled stance data affects cross-target and cross-domain generalization. Hyperparameters are summarized in Table 2. All runs used a LoRA rank $r = 8$, scaling factor $\alpha = 16$, bias=none, FP16 precision, and a warmup ratio of 0.1. Table 2 further details LoRA dropout, per-device batch size, gradient accumulation steps, number of epochs, and learning rate for each dataset group.

Dataset Group	LoRA dropout	Per-device batch	Grad-accum	Epochs	LR
SemEval/Covid	0.10	8	2	5	5×10^{-4}
P-Stance	0.05	16	2	3	3×10^{-4}
WT-WT	0.05	16	2	2	3×10^{-4}

Table 2: LoRA settings and training schedules. Global settings: r=8, α=16, bias=none; FP16, 0.1 warmup.

We adopt a single LoRA configuration ($r = 8$, $\alpha = 16$) across all models and datasets to enable controlled comparison. Varying configurations alongside model size, prompting strategy, and transfer setting would confound the diagnostic analysis. The chosen rank follows recent scaling analyses of efficient fine-tuning (Zhang et al., 2024a) and represents a widely used default; the trade-offs we report should therefore be read as characterizing this standard configuration rather than the full LoRA design space.

4 Results

4.1 RQ1: Effect of Model Size

In this section, we address **RQ1**: *How does model size shape generalization behavior across in-target, cross-target, and cross-domain stance detection settings?* Figure 1 summarizes the results, where each violin aggregates macro-F_1 scores across all prompting strategies and targets/domains for a given model size category: larger models generally outperform smaller ones under prompting, but this advantage diminishes and in some generalization settings, even reverses after task-specific adaptation.

In-Target Stance Detection Without fine-tuning, performance shows a strong scaling effect. As reported in Table 3, average macro-F_1 scores under $\mathbf{P^{ZS}}$ and across all strategies (**Base**) increased consistently with model size. The M24B model reached 0.6991, a 21% gain over Phi's 0.5782 across all prompting strategies. Larger models also yielded more stable performance, as reflected in smaller standard deviations across targets (**Base**). These findings indicate that when domain-matched training data are available, scaling model size provides reliable improvements in both effectiveness and robustness. After LoRA fine-tuning, however, this scale advantage largely disappears: a tuned 7B model (Mistral-7B) performs on par with the tuned 24B model, indicating that the main benefit of scale in SD lies in higher initial baselines, an advantage that parameter-efficient fine-tuning enables smaller models to close.

		In-Target		
Model Size	**Model**	$\mathbf{P^{ZS}}$	**Base**	**LoRA**
Small (3B)	Phi	0.4328 ± 0.1790	0.5782 ± 0.1381	0.8292 ± 0.0505
Medium (7B)	Mistral 7B	0.4528 ± 0.1978	0.6249 ± 0.1164	0.8576 ± 0.0678
Medium (8B)	L3-8B	0.4383 ± 0.1780	0.6330 ± 0.1110	0.8065 ± 0.0632
Large (24B)	M24B	0.4791 ± 0.2034	0.6991 ± 0.1058	0.8567 ± 0.0622
		Cross-Target/Domain		
Model Size	**Model**	$\mathbf{P^{KI}}$	**Base**	**LoRA**
Small (3B)	Phi	0.4574 ± 0.0765	0.4806 ± 0.0827	0.3312 ± 0.1467
Medium (7B)	M7B	0.3085 ± 0.1208	0.4692 ± 0.0791	0.3085 ± 0.1562
Medium (8B)	L3-8B	0.4527 ± 0.1768	0.4874 ± 0.1615	0.3553 ± 0.1522
Large (24B)	M24B	0.2626 ± 0.1540	0.4864 ± 0.1413	0.3048 ± 0.1584

Table 3: Model parameter size analysis. $\mathbf{P^{ZS}}$/$\mathbf{P^{KI}}$ report macro-F_1 ($\pm$ stdev) over 13 targets, 2 domains, and the respective prompt; **Base** reports averages across all prompts on non-fine-tuned models; **LoRA** reports averages across all prompts on LoRA fine-tuned models.

Cross-Target/Cross-Domain Stance Detection In SD_{CT} and SD_{CD}, model size shows weaker and inconsistent relationships with performance. Under $\mathbf{P^{KI}}$—the closest cross-target equivalent to zero-shot—larger models performed notably worse, while averaging across all strategies showed no consistent size advantage. No clear scaling law emerges: the medium 8B model (Llama-3-8B) achieved the highest overall score (**Base**), suggesting a balance between capacity and resistance to spurious pretraining patterns. LoRA fine-tuning consistently harmed generalization across all sizes, most severely in larger models: the relationship between size and performance reversed, consistent

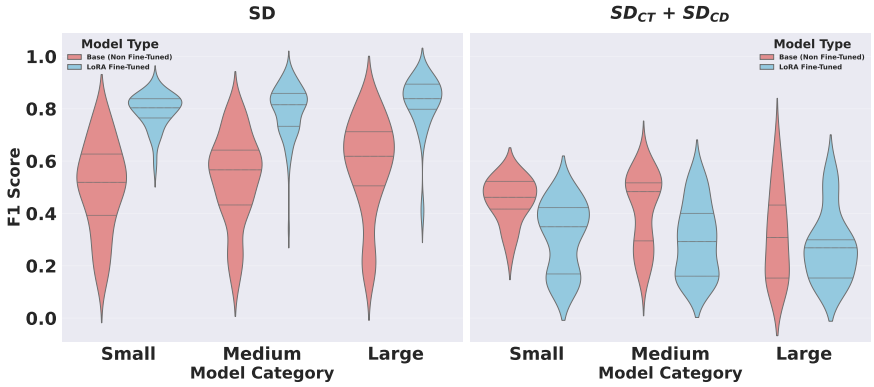

Figure 1: Distribution of macro-F_1 scores with (blue) and without (red) LoRA fine-tuning for small (Phi), medium (M7B, L3-8B), and large models (M24B). Each violin aggregates scores across all prompting strategies and targets/domains.

with the interpretation indicating that greater capacity increases susceptibility to source-domain overfitting. In short, more parameters do not guarantee better cross-context performance, and can accelerate degradation after fine-tuning.

4.2 RQ2: Prompt Effectiveness

This section analyzes the results to address **RQ2**: *How do different prompting strategies affect performance and transfer robustness across these settings?* Figure 2 contrasts base and fine-tuned F_1 distributions across six prompting strategies, where each violin aggregates scores across all models and targets/domains for a given strategy. For non-fine-tuned models, zero-shot and CoT approaches underperform in SD, whereas these same strategies perform best once fine-tuned. In cross-target and cross-domain settings, few-shot prompting is most advantageous, yet no fine-tuned model surpasses its base counterpart.

In-Target Stance Detection As shown in Table 4, the simplest strategies—P^{ZS} and P^{CoT}—perform worst (macro-F_1 of 0.4508 and 0.4458), suggesting that reasoning instructions alone provide no benefit over a minimal prompt. Adding background knowledge (P^{KI}, 0.5667) or labeled examples (P^{FS}, 0.5884) substantially improves performance, while combining all elements in $P^{CoT+KI+FS}$ achieves the highest score of 0.6279. This indicates that in-target performance scales with the amount of relevant context in the prompt. However, the optimal strategy also depends on model scale (Table 5): the smaller model performed best with P^{CoT+KI}, medium models with P^{FS}, and the larger model with $P^{CoT+KI+FS}$—suggesting that larger models are better able to synthesize multiple forms of guidance simultaneously. Overall, while minimal prompts can underperform, contextual enrichment substantially improves outcomes, with the most effective strategy depending on model scale.

	In-Target		*Cross-Target/Domain*	
Prompt	**Base**	**LoRA**	**Base**	**LoRA**
P^{CoT}	0.4458	0.8071	—	—
P^{ZS}	0.4508	0.8159	—	—
P^{CoT+KI}	0.5479	0.8003	0.3743	0.2897
P^{KI}	0.5667	0.8020	0.3703	0.2868
P^{FS}	0.5884	0.7845	0.4526	0.3083
$P^{CoT+KI+FS}$	0.6279	0.7862	0.3940	0.3011

Table 4: Prompt-specific average macro-F_1 scores.

Cross-Target/Cross-Domain Stance Detection In generalization tasks (SD_{CT} and SD_{CD}), the relative value of contextual cues shifts. As shown in Table 4, Few-Shot prompting achieves the highest mean score (0.4526), indicating that concrete task demonstrations transfer better than abstract background knowledge. However, the optimal strategy depends on model scale and—contrary to the in-target setting—becomes simpler as models grow larger (Table 5): the small model performs best with knowledge infusion, medium models with CoT, and the largest model with Few-Shot. This suggests that larger models may be hampered by additional scaffolding, generalizing more effectively with minimal guidance.

4.3 RQ3: LoRA Effectiveness

This section analyzes the results to address **RQ3**: *How does parameter-efficient fine-tuning via LoRA alter the balance between in-target performance*

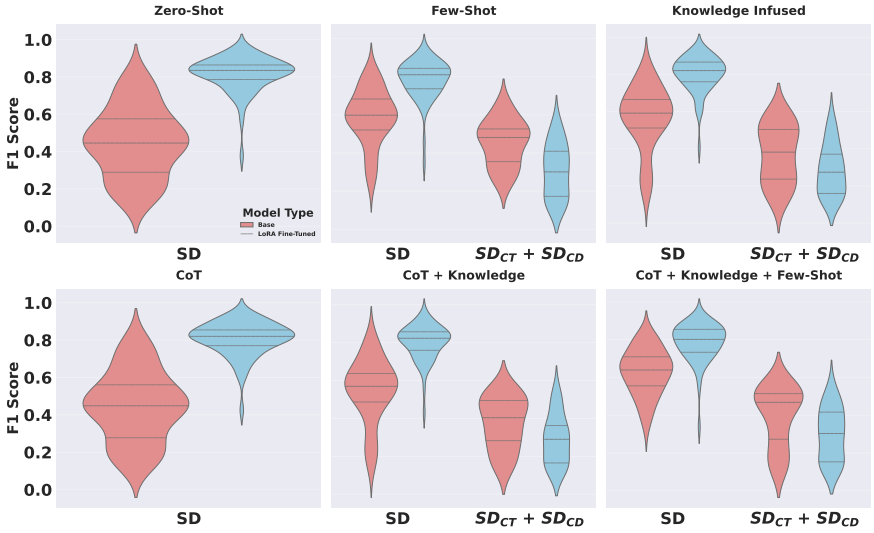

Figure 2: Distribution of macro-F_1 scores for base (red) and LoRA fine-tuned (blue) models across the six prompting strategies. Each strategy is evaluated in both in-target and cross-target settings, where applicable. Each violin aggregates scores across all models and targets/domains.

Setting	Size	Best Prompt	Base F_1	LoRA F_1
In-Target	Small	$P^{\text{CoT+KI}}$	0.456	0.523
In-Target	Medium	P^{FS}	0.612	0.645
In-Target	Large	$P^{\text{CoT+KI+FS}}$	0.678	0.701
Cross-Target/Domain	Small	P^{KI}	0.234	0.267
Cross-Target/Domain	Medium	P^{CoT}	0.345	0.356
Cross-Target/Domain	Large	P^{FS}	0.423	0.431

Table 5: Macro-F_1 averages for the optimal prompt strategy by model size.

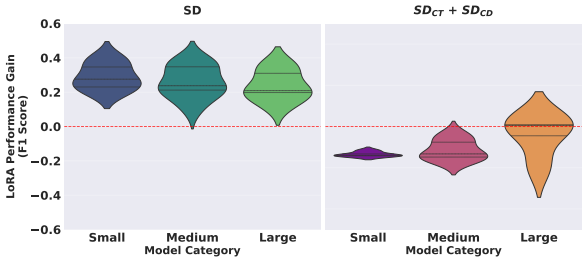

Figure 3: Distribution of LoRA performance gains (LoRA macro-F_1 minus Base macro-F_1) for small (Phi), medium (M7B, L3-8B), and large models (M24B). Each violin aggregates gains across all prompting strategies and targets/domains. The dashed red line indicates zero gain.

and cross-context generalization? As depicted in Figures 1, 2, and 3—where each violin in Figure 3 shows per-experiment LoRA gains (LoRA F_1 minus Base F_1) across all prompting strategies and targets/domains, LoRA fine-tuning consistently improves in-target performance while degrading cross-context transfer: $+48.6\%$ average macro-F_1 improvement in SD versus -25.5% degradation in SD$_{CT}$/SD$_{CD}$ (Table 6).

In-Target Stance Detection Considering macro-F_1 averages for all models and prompt strategies, LoRA fine-tuning yields performance gains ranging from $+40.8\%$ to $+57.5\%$, though the magnitude of improvement appears target-dependent. As shown in Table 7, LoRA's impact is inversely related to the base model's initial performance. For difficult targets with low baselines, such as *atheism* and *school_closures*, LoRA boosts macro-F_1 by $+185.8\%$ and $+152.8\%$, respectively. In contrast, for targets where base models already perform well (e.g., *Joe Biden* and *Bernie Sanders*), the gains are modest ($+8.6\%$ and $+9.5\%$). These findings suggest that LoRA is especially effective on challenging targets where prompting-based methods struggle. This raises an important direction for future work: systematically characterizing what makes certain targets "easy" or "hard" for stance detection models, which does not appear to correlate directly with obvious factors such as training set size (e.g., the WT-WT dataset, despite being the largest, still showed substantial improvement with LoRA). Notably, these same trends hold when considering macro-F_1 scores using the optimal prompting strategy by model size identified in Table 5.

Cross-Target/Cross-Domain Stance Detection LoRA fine-tuning on source-domain data consistently impairs models' ability to generalize to new targets and domains. As shown in Table 6, all models experience performance declines, though the

largest model (M24B) is most resilient (-13.0% vs. -30.2% for Phi-3B). This suggests that while larger models are still vulnerable to overfitting, their scale offers some protection against catastrophic forgetting. The negative effect holds across nearly all *source* $\rightarrow$ *destination* transfer pairs, with the steepest drops observed in cross-domain cases (*entertainment* $\rightarrow$ *healthcare* and *healthcare* $\rightarrow$ *entertainment*).

Model	Setting	Base	LoRA	Δ (%)	Best Prompt: Base	Best Prompt: LoRA
Phi (3B)	SD	0.5032	**0.7926**	+57.5	0.4328	0.8220
	SD_{CT}/SD_{CD}	**0.4484**	0.3131	-30.2	0.4296	0.3118
M7B	SD	0.5345	**0.8198**	+53.4	0.4506	0.8353
	SD_{CT}/SD_{CD}	**0.3878**	0.2884	-25.6	0.3432	0.2857
L3-8B	SD	0.5314	**0.7645**	+43.9	0.4330	0.7796
	SD_{CT}/SD_{CD}	**0.4387**	0.3091	-29.5	0.4400	0.2960
M24B	SD	0.5826	**0.8205**	+40.8	0.4176	0.8152
	SD_{CT}/SD_{CD}	**0.3164**	0.2753	-13.0	0.2626	0.2792
Mean (SD)		0.5379	**0.7993**	+48.6	0.4335	0.8130
Mean (SD_{CT}/SD_{CD})		**0.3978**	0.2965	-25.5	0.3688	0.2931

Table 6: Average macro-F_1 scores comparison between base and LoRA fine-tuned models across all models and prompt types, as well as for the optimal prompting strategies identified in Table 5.

Although we report SD_{CT} and SD_{CD} together, Table 7 reveals that the two settings degrade with different severity. Cross-target pairs within the same dataset (fm↔la, hc↔dt) show moderate drops, while cross-domain pairs across the entertainment and healthcare partitions of WT-WT degrade most sharply (-35.3% and -56.8%). This indicates that domain shift, rather than target shift alone, is the more significant driver of LoRA-induced generalization loss, and that aggregate SD_{CT}/SD_{CD} averages should be interpreted with this distinction in mind.

5 Discussion

A consistent pattern emerges across experiments: methods that improve in-target performance tend to reduce transfer robustness. While open-source LLMs achieve strong in-target detection after LoRA fine-tuning, their cross-context performance remains fragile—a tension evident across model scales, prompting strategies, and fine-tuning alike.

Diagnosing the Specialization–Generalization Trade-off. In in-target settings, prompting performance scales strongly with parameter count, yet LoRA fine-tuning largely neutralizes this scaling advantage: a tuned 7B model performs comparably to a tuned 24B model. Under transfer, however, scale does not reliably improve robustness. After fine-tuning, larger models often show greater performance degradation, consistent with increased specialization to source distributions. These findings suggest that lightweight parameter adaptation prioritizes distributional fit over invariance, constraining the model's ability to generalize across targets and domains.

Target	Base	LoRA	Δ (%)
school	0.3064	**0.7746**	+152.8
at	0.2291	**0.6546**	+185.8
ent	0.3615	**0.7870**	+117.7
face	0.5449	**0.8870**	+62.8
hlt	0.4792	**0.7967**	+66.3
fauci	0.5306	**0.8232**	+55.1
cc	0.4959	**0.7758**	+56.5
stay	0.6101	**0.8535**	+39.9
fm	0.5742	**0.7953**	+38.5
la	0.5089	**0.7257**	+42.6
dtp	0.6921	**0.9022**	+30.4
hc	0.6292	**0.8253**	+31.2
dt	0.5595	**0.7022**	+25.5
bernie	0.7425	**0.8133**	+9.5
joe	0.8045	**0.8737**	+8.6
fm $\rightarrow$ la	0.4866	**0.4940**	+1.5
la $\rightarrow$ fm	**0.3960**	0.3060	-22.7
hc $\rightarrow$ dt	**0.4669**	0.3588	-23.2
dt $\rightarrow$ hc	**0.4855**	0.3290	-32.3
hlt* $\rightarrow$ ent*	**0.2454**	0.1587	-35.3
ent* $\rightarrow$ hlt*	**0.3064**	0.1323	-56.8

Table 7: Performance comparison between base and LoRA fine-tuned models for all experiments (*source* $\rightarrow$ *destination* indicates cross-target/cross-domain scenarios). Values show mean of macro-F_1 across all models and prompt types.

Prompting strategies exhibit a similar pattern. Enriched prompts combining examples, knowledge, and reasoning steps improve in-target performance, particularly for smaller models. Yet under transfer, simpler few-shot prompting frequently yields more stable results. Together, these results indicate that both parameter adaptation and prompt scaffolding may encourage task-specific alignment at the expense of abstraction.

Importantly, these trade-offs are not isolated to a single model or dataset. Their consistency across scales and domains suggests that they reflect broader properties of adaptation in open-source LLMs rather than idiosyncrasies of individual sys-

tems.

Asymmetry Within Transfer Pairs. Performance changes are notably asymmetric within reciprocal pairs (Table 7). For example, fm→la shows a marginal $+1.5\%$ change after LoRA, while la→fm drops by -22.7%; the cross-domain pair degrades far more steeply in one direction (ent→hlt: -56.8%) than the other (hlt→ent: -35.3%). This asymmetry indicates that the direction of transfer matters within a pair: training on one target and evaluating on its reciprocal does not produce the same outcome as the reverse. The pattern is not explained by training set size—the fm and la training splits are nearly identical (664 vs. 653), and for the cross-domain pair the larger source (ent, 22,101) actually transfers worse than the smaller one (hlt, 11,141). Notably, similar directional asymmetries are visible in prior cross-target results on these same benchmark pairs across BiLSTM, BiCond, CrossNet, and BERT-based models (Augenstein et al., 2016; Xu et al., 2018; Liang et al., 2022), yet they are typically reported in tables without being analyzed as a phenomenon in their own right. Source-side factors beyond data volume—such as label distribution, lexical diversity, or topic-specific cues—may contribute, but a systematic characterization of which properties most influence transfer outcomes remains an open question for the field.

Practical Implications for Deployment. From a deployment perspective, these findings imply that configuration choices should depend on task requirements. When labeled data is available and high in-target accuracy is critical, LoRA fine-tuning offers substantial gains. However, for applications involving rapidly emerging topics or domain shifts, zero- or few-shot prompting may provide more stable cross-context performance. Simply increasing model size is not a guaranteed solution and introduces significant computational costs.

Prompt Engineering Considerations. Working with open-source models surfaced practical challenges that shaped the experimental pipeline. Smaller models required structured output formats (e.g., `{label:FAVOR}`) to ensure consistent parsing; less constrained prompts often produced ambiguous multi-label responses, such as discussing several labels before settling on one (*"While some might see this as AGAINST, the text is clearly in FAVOR"*), which broke automated parsing. Addressing model refusals on sensitive topics required research disclaimers to bypass default safety filters—acceptable in controlled experimentation but problematic in high-stakes domains, where forced compliance may yield confident but unreliable predictions. Models also occasionally defaulted to *none* despite it not being offered, seemingly reflecting a tendency toward neutrality on contentious subjects and while forced-choice prompting mitigated this, it introduced artificial certainty at the cost of expressive nuance. These observations highlight that deploying open-source LLMs for stance detection is as much about prompt level engineering as it is about model selection, and that small implementation choices can materially affect downstream interpretation.

6 Conclusion & Limitations

This study systematically evaluates scale, prompting, and parameter-efficient adaptation in open-source LLMs for stance detection, revealing a persistent tension: methods that improve in-target performance do not consistently enhance cross-context robustness. Effective deployment thus requires aligning model size, adaptation strategy, and anticipated distribution shift.

Our analysis is limited along several dimensions. First, we evaluate four open-source models up to 24B parameters; the trends we observe may not extrapolate to substantially larger or proprietary systems. Second, we adopt a single LoRA configuration to enable controlled comparison, and a wider exploration of ranks, target modules, or alternative PEFT methods (e.g., adapters, prefix tuning) could reveal whether the cross-context degradation we observe is configuration-specific or a more general phenomenon. Third, our datasets are of the English-language and span only a few domains, leaving open whether the specialization–generalization trade-off holds in multilingual or specialized settings (e.g., legal, financial, biomedical text). Finally, our findings are anchored to stance detection; whether comparable trade-offs arise in other classification tasks involving subjective or context-dependent judgments remains an open empirical question. Future work could explore alternative adaptation schemes, multi-task training to mitigate over-specialization, and transfer failures under broader domain and language settings.

7 Ethics Statement

This work evaluates open-source large language models for stance detection using publicly available, previously released datasets. All datasets employed in our experiments (SemEval 2016, P-Stance, COVID-19 Stance, and WT-WT) are publicly accessible and contain social media or publicly posted content. We do not introduce new data collection involving human subjects.

Stance detection has potential applications in areas such as political analysis, public opinion monitoring, and misinformation research. At the same time, automated stance classification systems may be misused for surveillance, targeted persuasion, or manipulation of public discourse. Our study does not deploy models in real-world settings but instead focuses on controlled evaluation to better understand generalization behavior and trade-offs under distribution shift. We emphasize that performance gains in in-target settings do not necessarily imply robustness to new or evolving topics.

Working with open-source LLMs also surfaced practical considerations, including model refusals on sensitive topics and the need for structured output constraints. While research disclaimers were used in controlled experiments to mitigate refusals, such techniques should be applied cautiously in high-stakes domains. Overconfident predictions or forced-choice classifications may oversimplify nuanced political or health-related discussions.

Finally, our analysis is limited to English-language datasets and open-source models up to 24B parameters. Broader evaluations across languages, domains, and demographic contexts are necessary to assess potential biases and fairness implications. We encourage responsible deployment practices and further study of generalization, bias, and misuse risks in stance detection systems.

References

Mistral AI. 2024. Mistral-small-instruct-2409 model card. https://huggingface.co/mistralai/Mistral-Small-Instruct-2409.

Nora Alturayeif, Hamzah Luqman, and Moataz Ahmed. 2023. A systematic review of machine learning techniques for stance detection and its applications. *Neural Comput. Appl.*, 35(7):5113–5144.

Isabelle Augenstein, Tim Rocktäschel, Andreas Vlachos, and Kalina Bontcheva. 2016. Stance detection with bidirectional conditional encoding.

Meghna Chaudhary and Tempestt Neal. 2026. Implicit aspect extraction: A systematic review. *ACM Comput. Surv.*, 58(7).

Costanza Conforti, Jakob Berndt, Mohammad Taher Pilehvar, Chryssi Giannitsarou, Flavio Toxvaerd, and Nigel Collier. 2020. Will-they-won't-they: A very large dataset for stance detection on Twitter. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1715–1724, Online. Association for Computational Linguistics.

Iain J. Cruickshank and Lynnette Hui Xian Ng. 2025. Prompting and fine-tuning open-sourced large language models for stance classification. *ACM Trans. Intell. Syst. Technol.* Just Accepted.

Margherita Gambini, Caterina Senette, Tiziano Fagni, and Maurizio Tesconi. 2024. Evaluating large language models for user stance detection on x (twitter). *Machine Learning*, 113(10):7243–7266.

Joseph Gatto, Omar Sharif, and Sarah M. Preum. 2023. Chain-of-thought embeddings for stance detection on social media. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 4154–4161, Singapore. Association for Computational Linguistics.

Parush Gera and Tempestt Neal. 2025. Deep learning in stance detection: A survey. *ACM Comput. Surv.* Just Accepted.

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*, pages 75–87, Cham. Springer International Publishing.

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.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, and et al. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Manuela Gómez-Suta, Julián Echeverry-Correa, and José A. Soto-Mejía. 2023. Stance detection in tweets: A topic modeling approach supporting explainability. *Expert Systems with Applications*, 214:119046.

Momchil Hardalov, Arnav Arora, Preslav Nakov, and Isabelle Augenstein. 2021. Cross-domain label-adaptive stance detection. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 9011–9028, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

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*, pages 71–77, Dublin, Ireland. Association for Computational Linguistics.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, and 1 others. 2022. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3.

Mikhail Isaev, Nic McDonald, and Richard Vuduc. Scaling infrastructure to support multi-trillion parameter llm training. In *Architecture and System Support for Transformer Models (ASSYST@ ISCA 2023)*.

Parisa Jamadi Khiabani and Arkaitz Zubiaga. 2025. Cross-target stance detection: A survey of techniques, datasets, and challenges. *Expert Systems with Applications*, 283:127790.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, and et al. 2023. Mistral 7b. *Preprint*, arXiv:2310.06825.

Nayoung Kim, David Mosallanezhad, Lu Cheng, Michelle V. Mancenido, and Huan Liu. 2025. Robust stance detection: Understanding public perceptions in social media. In *Social Networks Analysis and Mining*, pages 21–37, Cham. Springer Nature Switzerland.

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

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.

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)*, pages 81–91, Dublin, Ireland. Association for Computational Linguistics.

Junxia Ma, Changjiang Wang, Hanwen Xing, Dongming Zhao, and Yazhou Zhang. 2025. Chain of stance: Stance detection with large language models. In *Natural Language Processing and Chinese Computing*, pages 82–94, Singapore. Springer Nature Singapore.

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.

Lynnette Hui Xian Ng, Iain J Cruickshank, and Roy Lee. 2025. Examining the influence of political bias on large language model performance in stance classification. *Proceedings of the International AAAI Conference on Web and Social Media*, 19(1):1315–1328.

Quang Minh Nguyen and Taegyoon Kim. 2025. Is external information useful for stance detection with LLMs? In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 14798–14807, Vienna, Austria. Association for Computational Linguistics.

Sander Schulhoff, Michael Ilie, Nishant Balepur, Konstantine Kahadze, Amanda Liu, Chenglei Si, Yinheng Li, Aayush Gupta, HyoJung Han, Sevien Schulhoff, and 1 others. 2024. The prompt report: a systematic survey of prompt engineering techniques. *arXiv preprint arXiv:2406.06608*.

Melanie Sclar, Yejin Choi, Yulia Tsvetkov, and Alane Suhr. 2024. Quantifying language models' sensitivity to spurious features in prompt design or: How i learned to start worrying about prompt formatting. In *The Twelfth International Conference on Learning Representations*.

Stefan Sylvius Wagner, Maike Behrendt, Marc Ziegele, and Stefan Harmeling. 2024. The power of llm-generated synthetic data for stance detection in online political discussions. *arXiv preprint arXiv:2406.12480*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed Chi, Quoc V Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, volume 35, pages 24824–24837. Curran Associates, Inc.

Penghui Wei and Wenji Mao. 2019. Modeling transferable topics for cross-target stance detection. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR'19, page 1173–1176, New York, NY, USA. Association for Computing Machinery.

Maxwell Weinzierl and Sanda Harabagiu. 2024. Tree-of-counterfactual prompting for zero-shot stance detection. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 861–880, Bangkok, Thailand. Association for Computational Linguistics.

Chang Xu, Cécile Paris, Surya Nepal, and Ross Sparks. 2018. Cross-target stance classification with self-attention networks. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 778–783, Melbourne, Australia. Association for Computational Linguistics.

Jiaqing Yuan, Ruijie Xi, and Munindar P. Singh. 2025. A benchmark for cross-domain argumentative stance classification on social media. *Proceedings of the International AAAI Conference on Web and Social Media*, 19(1):2182–2196.

Sheng Zha, Vinh Q. Tran, Misha Laskin, Zhen Qin, Yuxiao Hu, and et al. 2024. Phi-3 technical report: A highly capable language model locally deployable on your device. *Preprint*, arXiv:2404.14219.

Biao Zhang, Zhongtao Liu, Colin Cherry, and Orhan Firat. 2024a. When scaling meets LLM finetuning: The effect of data, model and finetuning method. In *The Twelfth International Conference on Learning Representations*.

Hao Zhang, Yizhou Li, Tuanfei Zhu, and Chuang Li. 2024b. Commonsense-based adversarial learning framework for zero-shot stance detection. *Neurocomputing*, 563:126943.

Xinliang Frederick Zhang, Nick Beauchamp, and Lu Wang. 2022. Generative entity-to-entity stance detection with knowledge graph augmentation. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 9950–9969, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Yuan Zhang, Regina Barzilay, and Tommi Jaakkola. 2017. Aspect-augmented Adversarial Networks for Domain Adaptation. *Transactions of the Association for Computational Linguistics*, 5:515–528.

Zhengyuan Zhu, Zeyu Zhang, Haiqi Zhang, and Chengkai Li. 2025. RATSD: Retrieval augmented truthfulness stance detection from social media posts toward factual claims. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 3366–3381, Albuquerque, New Mexico. Association for Computational Linguistics.

Jingming Zhuo, Songyang Zhang, Xinyu Fang, Haodong Duan, Dahua Lin, and Kai Chen. 2024. ProSA: Assessing and understanding the prompt sensitivity of LLMs. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 1950–1976, Miami, Florida, USA. Association for Computational Linguistics.

8 Appendix

8.1 Training Time

Table 8 reports per-target dataset sizes and LoRA fine-tuning time in minutes measured on a single NVIDIA H100 GPU for L3-8B, M24B, M7B, and Phi.

Target	Size	L3-8B	M24B	M7B	Phi
hlt	29468	102.11	**267.71**	95.69	84.16
ent	14855	51.66	**135.74**	48.52	42.48
dt	8660	78.46	**203.13**	71.39	64.42
dtp	8660	45.25	**118.08**	42.29	37.17
joe	7296	38.18	**99.61**	35.66	31.33
bernie	6325	33.17	**86.42**	30.94	27.20
fauci	1864	17.06	**43.87**	15.47	13.95
face	1707	15.64	**40.18**	14.16	12.78
stay	1372	12.61	**32.31**	11.40	10.29
school	1190	10.96	**28.07**	9.89	8.94
hc	984	9.08	**23.17**	8.16	7.39
fm	949	8.77	**22.37**	7.87	7.13
la	933	8.63	**22.00**	7.74	7.01
at	733	6.82	**17.32**	6.10	5.53
cc	564	5.30	**13.36**	4.72	4.28

Table 8: Per–target dataset size and corresponding LoRA adapter training time (in minutes) for each model. Bold indicates the largest value per row.

8.2 Prompting Strategies

This study evaluates six prompting strategies, demonstrated below using the Llama-3 model. All prompts utilize *favor*, *against*, or *none* as labels, and the specific label text is dynamically updated based on the dataset.

1. **Zero-Shot (P^{ZS}).** A minimal instruction that defines "stance" and the label set, then asks for a single label for the quoted tweet.

Listing 1: Zero-Shot Prompt

```
<s>[INST]
Analyze the following tweet and determine the author's stance.
A "stance" refers to the author's clear position, whether they are in favor of, against, or none to a target implied by the tweet.
A Target can be an entity, organization, policy, person, etc.
The stance must be one of the following:

- favor
- against
- none

Your output should be in the format: {label: stance_label}

**Now, analyze the following tweet:**
Tweet: "This new policy is a game-changer for our city!"

Stance:
[/INST]
```

2. **Few-Shot (P^{FS}).** Prompt augmented with $k = 5$ labeled examples, per stance label, (each formatted as `Tweet: "..."` followed by `Stance: {label: ...}`) prior to the query instance.

Listing 2: Few-Shot Prompt

```
<s>[INST]
Analyze the following tweet and determine the author's stance.
A "stance" refers to the author's clear position, whether they are in favor of, against, or none to a target implied by the tweet.
A Target can be an entity, organization, policy, person, etc.

Here are a few examples to guide you:
Tweet: "I can't believe they're spending our money on this."
Stance: {label: against}

Tweet: "Finally, some positive changes around here!"
Stance: {label: favor}

The stance must be one of the following:
- favor
- against
- none

Your output should be in the format: {label: stance_label}

Now, analyze the following tweet.
Tweet: "This new policy is a game-changer for our city!"

Stance:
[/INST]
```

3. **Knowledge-Infused (P^{KI}).** Supplies a target/-domain information block prepended.

Listing 3: Knowledge-Infused Prompt

```
<s>[INST]
Given the following **Contextual Information**, analyze the tweet to determine the author's stance.
A "stance" refers to the author's clear position, whether they are in favor of, against, or none to a target implied by the tweet.
A Target can be an entity, organization, policy, person, etc.
The Contextual Information should be used to help understand the tweet's nuances and references.

**Target/Domain Information:**
The 'City Revitalization Act' is a new policy aimed at improving public spaces and supporting local businesses.

The stance must be one of the following:
- favor
- against
- none

Your output should be in the format: {label: stance_label}

**Now, analyze the following tweet:**
Tweet: "This new policy is a game-changer for our city!"

Stance:
[/INST]
```

4. **Chain-of-Thought (P^{CoT}).** Instructions to perform step-by-step internal reasoning (identify target → analyze cues → synthesize → decide) and emit only the final label.

Listing 4: Chain-of-Thought (CoT) Prompt

```
<s>[INST]
Analyze the following tweet and determine the author's stance.
A "stance" refers to the author's clear position, whether they are in favor of, against, or none to a target (topic, entity, policy etc.) implied by the tweet.

Before you decide on the stance, **think step-by-step through the reasoning process internally.** Your internal thinking should cover:
1. Identify the specific target or subject...
2. Analyze the tweet's language for sentiment, keywords, and contextual cues...
3. Synthesize the analysis to determine the author's clear position...
4. Justify your final stance based on the analysis.

After completing your internal reasoning, state only the final stance.
The stance must be one of the following:
- favor
- against
- none

Your output should be in the format: {label: stance_label}

**Now, analyze the following tweet:**
Tweet: "This new policy is a game-changer for our city!"

Stance:
[/INST]
```

5. **CoT + Knowledge-Infused (P^{CoT+KI}).** Combines a target/domain information block that supplies brief background relevant to the text/-target with requested internal reasoning using this information.

Listing 5: CoT + Knowledge Prompt

```
<s>[INST]
Given the following **Target/Domain Information**, analyze the tweet
```

```
    ↪ to determine the author's
    ↪ stance.
A "stance" refers to the author's clear
    ↪ position...

**Target/Domain Information:**
The 'City Revitalization Act' is a new
    ↪ policy aimed at improving public
    ↪ spaces and supporting local
    ↪ businesses.

Before you decide on the stance, **
    ↪ think step-by-step through the
    ↪ reasoning process internally.**
    ↪ Your internal thinking should
    ↪ cover:
1. Identify the specific target or
    ↪ subject...
2. Analyze the tweet's language...,
    ↪ using the provided **Target/
    ↪ Domain Information** for better
    ↪ understanding.
3. Synthesize the analysis to determine
    ↪ the author's clear position...
4. Justify your final stance based on
    ↪ the analysis and the provided
    ↪ knowledge.

After completing your internal
    ↪ reasoning, state only the final
    ↪ stance.
The stance must be one of the following:
    ↪
- favor
- against
- none

Your output should be in the format: {
    ↪ label: stance_label}

**Now, analyze the following tweet:**
Tweet: "This new policy is a game-
    ↪ changer for our city!"

Stance:
[/INST]
```

6. **CoT + Knowledge + Few-Shot ($P^{\text{CoT+KI+FS}}$).** Combines a target/domain information block, $k = 5$ labeled examples preceding the query, and requested internal step-by-step reasoning.

Listing 6: CoT + Knowledge + Few-Shot Prompt

```
<s>[INST]
Given the following **Target/Domain
    ↪ Information** and **Examples**,
    ↪ analyze the tweet to determine the
    ↪ author's stance.
A "stance" refers to the author's clear
    ↪ position...

**Target/Domain Information:**
The 'City Revitalization Act' is a new
    ↪ policy aimed at improving public
    ↪ spaces and supporting local
    ↪ businesses.

**Here are a few examples to guide you:**
Tweet: "I can't believe they're spending our
    ↪ money on this."
Stance: {label: against}

Tweet: "Finally, some positive changes
    ↪ around here!"
Stance: {label: favor}

Before you decide on the stance, **think
    ↪ step-by-step through the reasoning
    ↪ process internally.** Your internal
    ↪ thinking should cover:
1. Identify the specific target or subject...
    ↪
2. Analyze the tweet's language..., using
    ↪ the provided **Target/Domain
    ↪ Information** and **Examples** for
    ↪ better understanding.
3. Synthesize the analysis to determine the
    ↪ author's clear position...
4. Justify your final stance based on the
    ↪ analysis, the provided knowledge, and
    ↪ patterns observed in the examples.

After completing your internal reasoning,
    ↪ state only the final stance.
The final stance must be one of the
    ↪ following:
- favor
- against
- none

Your output should be in the format: {label:
    ↪ stance_label}

**Now, analyze the following tweet:**
Tweet: "This new policy is a game-changer
    ↪ for our city!"

Stance:
[/INST]
```