Seals-NLP at SemEval-2026 Task 9: A Comparative Study of Transformer Architectures for Polarization Detection

Minh Smith and Cheryl Seals
Department of Computer Science and Software Engineering, Auburn University, AL, USA
{mzs0193, sealscd}@auburn.edu

Abstract

We describe the Seals-NLP system for SemEval-2026 Task 9 (POLAR) Subtask 1, binary polarization detection. Our study compares (i) fully fine-tuned encoder-only transformers, (ii) QLoRA-based fine-tuned open-weight LLMs, and (iii) zero-shot prompted LLMs. `ModernBERT-large` emerges as the most cost-effective option, matching or surpassing larger fine-tuned and zero-shot LLMs in macro-F_1 while requiring substantially less memory and lower latency. An error analysis by failure mode and polarization subtype reveals systematic over-triggering on political cue words and under-detection of sarcastic vilification and multifaceted attacks in the POLAR dataset across all models.

1 Introduction

Online polarization, or sharp opinion division accompanied by hostility, stereotyping, or intolerance, poses a growing threat to open and democratic discourse (Naseem et al., 2026b). On platforms such as X, Reddit and Facebook, polarized content can escalate into hate speech and broader fragmentation leading to social decline, motivating better understanding and automatic detection as a Natural Language Processing (NLP) task (Naseem et al., 2026b).

A key question today is which modeling paradigm best captures polarization in text. Encoder-only transformers such as BERT (Devlin et al., 2019) and RoBERTa (Liu et al., 2019) excel at classification, while generative LLMs offer deeper contextual and semantic insights. Existing comparisons often focus on a single family or rely on proprietary API-based models, limiting reproducibility (Zhang et al., 2025). We focus on Subtask 1 of SemEval-2026 Task 9 (POLAR) (Naseem et al., 2026a), a binary polarization detection task over English social media text.

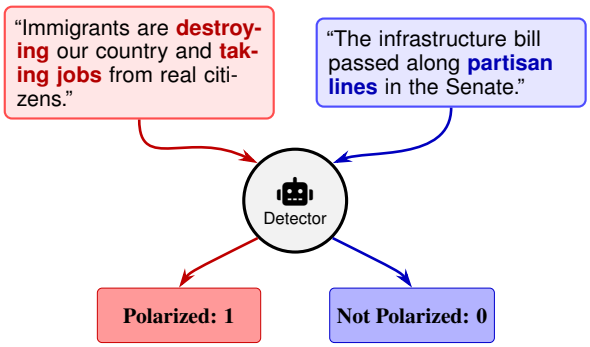

Figure 1: System pipeline for polarization detection. Social media posts are passed through a detector model that provides a label of polarized or not polarized.

Our contributions[1] are as follows:

a) A systematic comparison of encoder-only transformers, open-weight LLMs, and zero-shot LLMs on the POLAR dataset, using only open or openly licensed models.

b) A cost-effectiveness study combining macro-F_1 with storage footprint and latency.

c) An error-oriented analysis by polarization subtype and rhetorical strategy.

2 Background

Task definition POLAR is a shared task specifically devoted to online polarization (Naseem et al., 2026a,b). Subtask 1 is a classification task, as seen in Figure 1, where systems must classify texts as "Polarized" or "Not Polarized".

Related work Previous work in online polarization often relies on network-based methods such as retweet-graph partitioning and controversy scores (Conover et al., 2011; Garimella et al., 2018). Recent approaches combine language and network

[1]Code available at https://github.com/leonidasengineer-star/polar-semeval-seals-nlp.

Proceedings of the The 20th International Workshop on Semantic Evaluation (2026), pages 2394–2401
July 3-4, 2026

features, such as Retweet-BERT for user-level polarity (Jiang et al., 2023), or use sentence transformers to quantify echo chambers (Ghafouri, 2025). Adjacent tasks include stance detection (Mohammad et al., 2016), hate or offensive language detection (Zampieri et al., 2019), and sentiment analysis in low-resource settings (Raychawdhary et al., 2023), but polarization targets broader group-based antagonism that may be present without overt abuse.

Early neural approaches to social-media classification widely adopted BERT and related variants as strong general-purpose baselines for sentiment and document classification (Devlin et al., 2019). Subsequent encoder-only architectures such as RoBERTa and DeBERTa-v3 improved over BERT by refining pretraining objectives and attention mechanisms, yielding richer contextual representations and stronger downstream performance on noisy text (Liu et al., 2019; He et al., 2023). More recent models like ModernBERT further modernize the bidirectional encoder with improved efficiency and extended context support, achieving favorable speed-accuracy trade-offs for classification and retrieval (Warner et al., 2025; Stepanov, 2025; Antoun et al., 2025; Gottesman et al., 2025).

Instruction-tuned LLMs are widely applied to downstream classification tasks via prompting and fine-tuning (Lou et al., 2024; Han et al., 2025; Zhang et al., 2026). Parameter-efficient fine-tuning (PEFT) adapts large pretrained models by training only a small number of additional parameters while keeping most weights frozen. Variants include LoRA (Hu et al., 2022), QLoRA (Dettmers et al., 2023), DoRA (Liu et al., 2024), PiSSA (Meng et al., 2024), SeLoRA (Cheng et al., 2025), and EVA (Paischer et al., 2024), each differing in how adapter weights are initialized or updated. Empirical studies show such mechanisms can match full fine-tuning (Nwaiwu, 2025).

3 System Overview

Table 1 summarises all nine evaluated models consisting of encoders, fine-tuned LLMs, and zero-shot LLMs. The *'-ft'* and *'-zs'* suffixes denote fine-tuned and zero-shot variants respectively.

Encoders We evaluate three encoder-only architectures as fully fine-tuned baselines. `BERT-base-uncased` serves as a classic BERT baseline for social-media classification (Devlin et al., 2019). `DeBERTa-v3-large` provides a stronger encoder with disentangled attention and improved pretraining (He et al., 2023). `ModernBERT-large` is a recent efficiency-oriented encoder with long-context support and optimized attention (Warner et al., 2025). We initialize a binary sequence-classification head and update all encoder and head weights jointly.

Model	Developer	Params
Encoders (fully fine-tuned)		
BERT-base-uncased	Microsoft	110M
DeBERTa-v3-large	Microsoft	304M
ModernBERT-large	Answer.AI	395M
Fine-tuned LLMs (NF4 QLoRA)		
Gemma-3-27b-it-ft	Google	27B
Qwen3-32b-ft	Alibaba	32B
EXAONE-4.0-32b-ft	LG AI	32B
Zero-shot LLMs (base)		
Gemma-3-27b-it-zs	Google	27B
Qwen3-32b-zs	Alibaba	32B
EXAONE-4.0-32b-zs	LG AI	32B

Table 1: Overview of all evaluated models. *Params* denote total parameter counts.

Fine-tuned LLMs We select open-weight decoder LLMs `Gemma-3-27b-it` (Gemma Team, 2025), `Qwen3-32b` (Qwen Team, 2025), and `EXAONE-4.0-32b` (LG AI Research, 2025) as base models for QLoRA-based fine-tuning. We attach LoRA adapters to attention projections of each quantized LLM and train only these adapters and a classification head, keeping base weights frozen. This preserves the capacity of multi-billion-parameter models while restricting gradient updates to a small subset of parameters. We adopt QLoRA to enable adaptation within a single-GPU memory budget.

Zero-shot LLMs For zero-shot experiments, we use the same base LLM models without additional training. We prompt each model with a short set of instructions that describes the polarization task then generates a response where the first occurrence of "0" or "1" in the output is mapped to the non-polarized or polarized labels. This ensures that zero-shot and fine-tuned LLMs provide similar evaluation conditions and are evaluated under the same binary decision rule.

4 Experimental Setup

Data We use the English portion of the POLAR dataset for Subtask 1 (Naseem et al., 2026b,a), which contains 3,222 training instances, 160 development instances, and 1,452 test instances. The

official splits are imbalanced, with roughly one-third polarized examples across all partitions. The corpus includes social-media texts from X, Reddit, blogs, news comments, and regional forums, covering events such as elections, conflicts, protests, and migration. We adopt the organizers' train/dev splits and do not use any additional labeled or unlabeled data.

Preprocessing We apply each model's native tokenizer with no additional normalization, preserving casing, misspellings, and hashtags, truncating to model-specific maximum lengths.

Implementation We use the Hugging Face `transformers` and `datasets` libraries with PyTorch backends for all fine-tuning and inference.

Training Configuration All experiments run on a single `NVIDIA RTX 5090` (32 GB `VRAM`, 128 GB RAM). Encoders are fully fine-tuned in `bfloat16` with `AdamW`-style optimizers and early stopping on macro-F_1 with full settings provided in Appendix A.1. QLoRA LLMs load base weights in 4-bit NF4 with double quantization; LoRA adapters and a binary classification head are trained in `bfloat16` with a cosine schedule. Zero-shot LLMs use the same instruction-tuned checkpoints with 4-bit NF4 quantization; binary labels are derived from generated tokens using the prompt in Appendix A.2.

Metrics and Evaluation For each model, we compute precision, recall, and macro-F_1 over the official binary labels, and record average per-instance latency. Scores are computed on the official English dev split. Precision, recall, and macro-F_1 are computed by first obtaining the class-wise precision, recall, and F_1 and then averaging over the two classes.

For a given class c, precision and recall are:

$$\text{Precision}_c = \frac{\text{TP}_c}{\text{TP}_c + \text{FP}_c} \tag{1}$$

$$\text{Recall}_c = \frac{\text{TP}_c}{\text{TP}_c + \text{FN}_c} \tag{2}$$

The class-wise F_1-score is:

$$F_{1c} = 2 \cdot \frac{\text{Precision}_c \cdot \text{Recall}_c}{\text{Precision}_c + \text{Recall}_c} \tag{3}$$

macro-F_1 is then the unweighted mean over all classes:

$$F_{1\text{-macro}} = \frac{1}{C} \sum_{c=1}^{C} F_{1c}. \tag{4}$$

We use macro-F_1 as the primary selection, reporting, and evaluation metric throughout.

5 Results and Discussion

5.1 Main Results

Overall scores Table 2 summarizes performance across all architectures for Subtask 1. Among encoders, `ModernBERT-large` attains the strongest macro-F_1, only slightly outperforming `BERT-base-uncased` despite using a more modern architecture and longer context window. `DeBERTa-v3-large` underperforms markedly in this setup, suggesting sensitivity to optimization on the POLAR dataset and the difficulty of polarizing speech.

Model	Precision	Recall	Macro-F1
Encoders (fully fine-tuned)			
BERT-base-uncased	**0.7897**	0.7746	0.7802
DeBERTa-v3-large	0.3156	0.5000	0.3870
ModernBERT-large	0.7786	**0.7873**	**0.7821**
Fine-tuned LLMs (NF4 QLoRA)			
Gemma-3-27b-it-ft	0.4815	0.4895	0.3716
Qwen3-32b-ft	**0.6138**	**0.5602**	0.4338
EXAONE-4.0-32b-ft	0.4948	0.4971	**0.4673**
Zero-shot LLMs (base)			
Gemma-3-27b-it-zs	**0.7783**	**0.7673**	**0.7060**
Qwen3-32b-zs	0.6235	0.6114	0.6139
EXAONE-4.0-32b-zs	0.6667	0.5608	0.4050

Table 2: Performance across all evaluated groups. Best scores per group are bolded.

Cross-paradigm analysis When comparing paradigms, the best encoder, `ModernBERT-large`, surpasses the best fine-tuned LLM by a wide margin despite being far smaller. Fine-tuned LLMs underperform their zero-shot counterparts on POLAR, suggesting that aggressive quantization and limited data hurt adaptation for this task. The sole exception is `EXAONE-4.0-32b-ft`, whose fine-tuned variant outperforms its zero-shot configuration, but still lags behind encoder baselines. This may indicate that EXAONE-4.0's pretraining or instruction-tuning distribution is more amenable to binary classification adaptation under QLoRA. Under our constraints, the benefits of QLoRA do not compensate for the loss in performance, suggesting that these factors may outweigh the capacity advantage of larger models on this task.

Leaderboard Table 3 shows the official Subtask 1 English leaderboard results, where our fine-tuned `ModernBERT-large` achieved a macro-F_1

of 0.7832, placing 32nd out of 44 submissions (Naseem et al., 2026a).

Rank	Team	Macro-F1
1	UTokyo Tsuruoka Lab	0.8252
2	danielkhir	0.8189
3	PSK	0.8177
4	NYCU-NLP	0.8172
5	yunkuang0329	0.8153
	⋮	
30	joshualee2	0.7846
31	TranTranUIT	0.7845
32	**Seals-NLP (ours)**	**0.7832**
33	YEZE	0.7806
34	wangkongqiang	0.7805
–	POLAR Baseline	0.7802

Table 3: Selected ranks showing the Official Subtask 1 English leaderboard.

5.2 Cost-Effectiveness Analysis

Table 4 summarizes the deployment trade-offs in terms of model size and single-sequence latency. Among encoders, `ModernBERT-large` offers an ideal balance: it achieves the highest macro-F_1 in its family while remaining under 8 GB and responding in roughly 20 ms per input. This makes it well-suited for high-throughput social-media monitoring and useful for consumers with lighter hardware requirements. `BERT-base-uncased` is even smaller and faster, sacrificing only a small amount of accuracy for a model of its age. `DeBERTa-v3-large` is both slower and less effective in our setup and appears to suffer in binary classification with such nuanced context.

Model	Size	Latency / seq
Encoders (fully fine-tuned)		
BERT-base-uncased	0.44 GB	5.1 ms
DeBERTa-v3-large	1.2 GB	28.8 ms
ModernBERT-large	1.6 GB	20.3 ms
Fine-tuned LLMs (NF4 QLoRA)		
Gemma-3-27b-it-ft	54.9 GB	19.9 s
Qwen3-32b-ft	65.5 GB	11.4 s
EXAONE-4.0-32b-ft	63.1 GB	9.4 s
Zero-shot LLMs (base)		
Gemma-3-27b-it-zs	54.9 GB	45.5 s
Qwen3-32b-zs	65.5 GB	40.6 s
EXAONE-4.0-32b-zs	63.1 GB	51.2 s

Table 4: Inference efficiency of the models. *Size* denotes the total file size of the model weights. *Latency / seq* is average wall-clock time per input sequence.

Fine-tuned LLMs sit in a very different regime and require an order of magnitude more storage capacity, even though quantization can reduce their size by half or more while largely preserving accuracy and stability. They are also orders of magnitude slower and, on POLAR, show no noticeable improvement, although quantization does improve speed.

Zero-shot LLMs inherit the same storage and latency profile but perform better than their fine-tuned counterparts, suggesting that, for this binary polarization task, large LLMs may not be a practical deployment option. Notably, zero-shot inference is slower than fine-tuned despite better accuracy, as fine-tuning's classification head allows earlier termination while zero-shot must generate a full response. Part of the measured zero-shot latency is attributable to the generative formulation because the model must generate a response before label extraction can occur. LLMs, overall, appear better reserved for their intended task of generation or low-throughput scenarios where latency and hardware cost are less constrained.

5.3 Error Analysis

Difficult examples Table 5 shows samples that many models failed to predict correctly. These failures show over-triggering on cue words and references, even when the post may lack clear antagonism or overt hostility. Short or innocuous statements about parties, elections, or border policy are frequently labeled as polarized, whereas genuinely polarized text that relies on mockery or indirect vilification is more likely to be missed. By contrast, subtypes such as religious targeting show lower error rates, suggesting models handle more explicit or conventionally recognized forms of antagonism more reliably. Together, these patterns suggest that systems rely heavily on surface-level markers of controversy rather than robustly modeling intent, animus, and stance.

Sublabel failures Table 6 summarizes how these mistakes are distributed over polarization subtypes. Political text accounts for the largest share of incorrect predictions, followed by content annotated with extreme language, vilification and invalidation, indicating systems struggle both with borderline political commentary and with multi-faceted, highly negative attacks. Subtypes such as dehumanization, lack of empathy and racial or ethnic targeting also accumulate substantial errors, showing that explicitly harmful patterns can be hard to capture. These sublabel failures appear across all models and likely challenge many human readers as well, suggesting that future work should prioritize more nuanced architectures with richer multi-label repre-

#	Failure mechanism	Labels	Example text	Models
Non-polarizing examples misclassified as polarized				
1	Over-triggering on border rhetoric	Not polarizing	"Legal immigrants . . . with **open borders** our country will not exist for very long."	8/9
2	Over-triggering on partisan cues	Not polarizing	"DOES ANYONE HAVE A LIST OF JUST THE **BLUE STATES**??"	7/9
3	Over-triggering on anti-Trump framing	Not polarizing	"Can **Trumpism** be defeated? Absolutely. Heres how"	7/9
4	Over-triggering on ideological labels	Not polarizing	"BLUESKY **Socialist** Media with **Communist** Guidelines."	7/9
5	Over-triggering on party identity	Not polarizing	"**Democrats** are not **liberals** actually."	7/9
Polarizing examples missed by many models				
1	Under-detection of sarcastic vilification	Political; Vilification; Extreme; Invalidation	"Zelensky drops dime on the **Brandon crime family**."	6/9
2	Under-detection of media attack	Political; Stereotype; Vilification; Invalidation	"Every **MSNBC** article reads like they had **no idea** this would happen"	6/9
3	Under-detection of sexualized vilification	Political; Gender; Dehumanization; Extreme	"Tomorrow Mike Pence is in love with **bussy**"	5/9
4	Under-detection of partisan dehumanization	Political; Stereotype; Dehumanization; Invalidation	"**red states** take more **handouts** than the **blue states**."	5/9
5	Under-detection of accusatory attack	Political; Extreme; Invalidation	"Then why did you **withhold military aid** to them??"	5/9

Table 5: Illustrative examples of the top 5 non-polarizing and top 5 polarizing failures. *Failure mechanisms* were inferred from qualitative inspection of highlighted words or phrases. *Models* indicate how many systems misclassified each example.

sentations of polarization.

Subtype	Incorrect	Unique	Models
Political	186	58	8 / 9
Extreme Language	130	41	8 / 9
Vilification	123	39	8 / 9
Invalidation	99	29	7 / 9
Stereotype	74	24	7 / 9
Lack of Empathy	59	18	6 / 9
Dehumanization	57	19	7 / 9
Racial/Ethnic	37	14	5 / 9
Other	20	6	6 / 9
Religious	13	5	3 / 9
Gender/Sexual	12	3	7 / 9

Table 6: Sublabel failure analysis. *Incorrect* counts the total number of misclassified predictions across all models for examples annotated with each subtype. *Unique* counts the number of distinct misclassified examples. *Models* indicates how many of the nine systems contributed errors for that subtype.

6 Limitations

Our study has several limitations. Findings may not generalize to other languages or cultural contexts, and the POLAR dataset's political skew may inflate errors in that subtype. Evaluation relies solely on the development set, with no variance across runs. Zero-shot evaluation uses only one prompt and does not explore prompt sensitivity, few-shot variants, or chain-of-thought formulations. The underperformance of fine-tuned LLMs relative to zero-shot remains unexplained, as we did not ablate quantization level or training data size. Latency figures are not fully controlled, since generation is a confounding factor, and failure mechanisms were inferred from qualitative inspection alone.

7 Conclusion

In this work, we presented the Seals-NLP system for POLAR English polarization detection, conducting a comparison of encoder-only transformers, fine-tuned large language models and zero-shot LLMs. Our experiments show that while LLMs can achieve comparable macro-F_1, encoder models, particularly `ModernBERT-large`, deliver the best balance between accuracy, efficiency, and computational cost. These findings suggest that polarization detection, as a focused classification problem, still favors compact encoder architectures over large-scale generative models. Future work will include ablations to isolate component contributions and evaluation of emerging architectures such as Mamba-based and mixture-of-experts models.

References

Wissam Antoun, Benoît Sagot, and Djamé Seddah. 2025. ModernBERT or DeBERTaV3? Examining architecture and data influence on transformer encoder models performance. In *Proceedings of the 14th International Joint Conference on Natural Language Processing and the 4th Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics (IJCNLP-AACL)*, pages 3061–3074, Mumbai, India. The Asian Federation of Natural Language Processing and The Association for Computational Linguistics.

Jiashun Cheng, Aochuan Chen, Nuo Chen, Ziqi Gao, Yuhan Li, Jia Li, and Fugee Tsung. 2025. Revisiting LoRA through the lens of parameter redundancy: Spectral encoding helps. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 2701–2718, Vienna, Austria. Association for Computational Linguistics.

Michael D Conover, Jacob Ratkiewicz, Matthew Francisco, Bruno Gonçalves, Filippo Menczer, and Alessandro Flammini. 2011. Political polarization on twitter. In *Proceedings of the 5th International AAAI Conference on Weblogs and Social Media (ICWSM)*, pages 89–96, Barcelona, Spain. AAAI Press.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. QLoRA: Efficient finetuning of quantized large language models. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 36, pages 10088–10115. Curran Associates, Inc.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Kiran Garimella, Gianmarco De Francisci Morales, Aristides Gionis, and Michael Mathioudakis. 2018. Quantifying controversy on social media. *ACM Transactions on Social Computing*, 1(1):3:1–3:27.

Gemma Team. 2025. Gemma 3 technical report. *Preprint*, arXiv:2503.19786.

Vahid Ghafouri. 2025. *NLP-Driven Approaches to Measuring Online Polarization and Radicalization*. Ph.D. thesis, IMDEA Networks Institute / Universidad Carlos III de Madrid, Madrid, Spain.

Daniela Gottesman, Alon Gilae-Dotan, Ido Cohen, Yoav Gur-Arieh, Marius Mosbach, Ori Yoran, and Mor Geva. 2025. LMEnt: A suite for analyzing knowledge in language models from pretraining data to representations. *Preprint*, arXiv:2509.03405.

Xudong Han, Junjie Yang, Tianyang Wang, Ziqian Bi, Xinyuan Song, Junfeng Hao, and Junhao Song. 2025. Towards alignment-centric paradigm: A survey of instruction tuning in large language models. *Preprint*, arXiv:2508.17184.

Pengcheng He, Jianfeng Gao, and Weizhu Chen. 2023. DeBERTaV3: Improving DeBERTa using ELECTRA-style pre-training with gradient-disentangled embedding sharing. In *Proceedings of the 11th International Conference on Learning Representations (ICLR)*, Kigali, Rwanda. OpenReview.net.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Lu Wang, Shean Li, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. In *Proceedings of the 10th International Conference on Learning Representations (ICLR)*, Virtual. OpenReview.net.

Julie Jiang, Xiang Ren, and Emilio Ferrara. 2023. Retweet-BERT: Political leaning detection using language features and information diffusion on social networks. In *Proceedings of the 17th International AAAI Conference on Web and Social Media (ICWSM)*, pages 459–469, Limassol, Cyprus.

LG AI Research. 2025. EXAONE 4.0: Unified large language models integrating non-reasoning and reasoning capabilities. *Preprint*, arXiv:2507.11407.

Shih-Yang Liu, Chien-Yi Wang, Hongxu Yin, Pavlo Molchanov, Yu-Chiang Frank Wang, Kwang-Ting Cheng, and Min-Hung Chen. 2024. DoRA: Weight-decomposed low-rank adaptation. In *Proceedings of the 41st International Conference on Machine Learning (ICML)*, volume 235 of *Proceedings of Machine Learning Research*, pages 32100–32121. PMLR.

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. *Preprint*, arXiv:1907.11692.

Renze Lou, Kai Zhang, and Wenpeng Yin. 2024. Large language model instruction following: A survey of progresses and challenges. *Computational Linguistics*, 50(3):1053–1095.

Fanxu Meng, Zhaohui Wang, and Muhan Zhang. 2024. PiSSA: Principal singular values and singular vectors adaptation of large language models. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 37.

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.

Usman Naseem, Robert Geislinger, Juan Ren, Sarah Kohail, Rudy Garrido Veliz, P Sam Sahil, Yiran Zhang, Marco Antonio Stranisci, Idris Abdulmumin, Özge Alacam, Cengiz Acarürk, Aisha Jabr, Saba Anwar, Abinew Ali Ayele, Elena Tutubalina, Aung Kyaw Htet, Xintong Wang, Surendrabikram Thapa, Tanmoy Chakraborty, and 15 others. 2026a. SemEval-2026 Task 9: Detecting Multilingual, Multicultural and Multievent Online Polarization. In *Proceedings of the 20th International Workshop on Semantic Evaluation (SemEval-2026)*, San Diego, CA, USA. Association for Computational Linguistics.

Usman Naseem, Robert Geislinger, Juan Ren, Sarah Kohail, Rudy Garrido Veliz, P Sam Sahil, Yiran Zhang, Marco Antonio Stranisci, Idris Abdulmumin, Özge Alacam, Cengiz Acarürk, Aisha Jabr, Saba Anwar, Abinew Ali Ayele, Simona Frenda, Alessandra Teresa Cignarella, Elena Tutubalina, Oleg Rogov, Aung Kyaw Htet, and 24 others. 2026b. POLAR: A benchmark for multilingual, multicultural, and multi-event online polarization. *Preprint*, arXiv:2505.20624.

Steve Nwaiwu. 2025. Parameter-efficient fine-tuning for low-resource text classification: A comparative study of LoRA, IA^3, and ReFT. *Frontiers in Big Data*, 8:1677331.

Fabian Paischer, Lukas Hauzenberger, Thomas Schmied, Benedikt Alkin, Marc Peter Deisenroth, and Sepp Hochreiter. 2024. One initialization to rule them all: Fine-tuning via explained variance adaptation. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 37.

Qwen Team. 2025. Qwen3 technical report. *Preprint*, arXiv:2505.09388.

Nilanjana Raychawdhary, Amit Das, Gerry Dozier, and Cheryl D. Seals. 2023. Seals_Lab at SemEval-2023 task 12: Sentiment analysis for low-resource African languages, Hausa and Igbo. In *Proceedings of the 17th International Workshop on Semantic Evaluation (SemEval-2023)*, pages 1508–1517, Toronto, Canada. Association for Computational Linguistics.

Ihor Stepanov. 2025. FlashDeBERTa: Flash attention implementation of DeBERTa disentangled attention. `https://github.com/Knowledgator/FlashDeBERTa`. GitHub repository.

Benjamin Warner, Antoine Chaffin, Benjamin Clavié, Orion Weller, Oskar Hallström, Said Taghadouini, Alexis Gallagher, Raja Biswas, Faisal Ladhak, Tom Aarsen, Nathan Cooper, Griffin Thomas Adams, Jeremy Howard, and Iacopo Poli. 2025. Smarter, better, faster, longer: A modern bidirectional encoder for fast, memory efficient, and long context finetuning and inference. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2526–2547, Vienna, Austria. Association for Computational Linguistics.

Marcos Zampieri, Shervin Malmasi, Preslav Nakov, Sara Rosenthal, Noura Farra, and Ritesh Kumar. 2019. SemEval-2019 Task 6: Identifying and categorizing offensive language in social media (OffensEval). In *Proceedings of the 13th International Workshop on Semantic Evaluation (SemEval-2019)*, pages 75–86, Minneapolis, Minnesota, USA. Association for Computational Linguistics.

Junyan Zhang, Yiming Huang, Shuliang Liu, Yubo Gao, and Xuming Hu. 2025. Do BERT-like bidirectional models still perform better on text classification in the era of LLMs? In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 18980–18989. Association for Computational Linguistics.

Shengyu Zhang, Linfeng Dong, Xiaoya Li, Sen Zhang, Xiaofei Sun, Shuhe Wang, Jiwei Li, Runyi Hu, Tianwei Zhang, Fei Wu, and Guoyin Wang. 2026. Instruction tuning for large language models: A survey. *ACM Computing Surveys*, 58(7):169:1–169:36.

A Appendix

A.1 Hyperparameters

Table A.1 reports the complete training configuration for all model families. Encoder and QLoRA LLM settings were selected via light manual tuning on the development set.

Setting	Value
Encoders (full fine-tuning)	
Learning rate	2e-4
Epochs	6
Batch size	32
Grad. accum.	2
Warmup steps	10
Weight decay	0.01
Precision	`bfloat16`
Fine-tuned LLMs (NF4 QLoRA)	
Learning rate	2e-4
Epochs	6
Batch size	2
Grad. accum.	8
Warmup steps	100
Weight decay	0.01
LR schedule	cosine
LoRA r / α	8 / 16
LoRA dropout	0.05
Target modules	`q_proj`, `v_proj`
Quantization	NF4 + `bfloat16` adapters

Table A.1: Hyperparameter configurations for training.

A.2 Zero-Shot Prompt

Table A.2 shows the instruction template used for all zero-shot LLM experiments. The placeholder `[input text]` is replaced with the raw post at inference time. We deliberately kept the prompt minimal to avoid inadvertently encoding task-specific heuristics that might inflate zero-shot performance

or introduce prompt-induced bias. The binary response format was chosen to simplify label extraction: we map the first occurrence of `0` or `1` in the model output to the not-polarized or polarized class respectively, discarding any additional generated text.

```
Classify this text as polarized (1) or not
(0). Answer with only the digit.
Text: [input text]
```

Table A.2: Zero-shot prompt template used for all LLM zero-shot experiments. The prompt was held constant across all three LLMs and all inputs, with no few-shot examples or chain-of-thought instructions.

A.3 Models

Table A.3 lists the Hugging Face checkpoints used in all experiments. All models are publicly available and were loaded directly from the Hugging Face Hub without modification to their base weights prior to fine-tuning or zero-shot inference.

Model	**Hugging Face Repo**
BERT-base-uncased	`bert-base-uncased`
DeBERTa-v3-large	`microsoft/deberta-v3-large`
ModernBERT-large	`answerdotai/ModernBERT-large`
Gemma-3-27b-it	`google/gemma-3-27b-it`
Qwen3-32b	`Qwen/Qwen3-32B`
EXAONE-4.0-32b	`LGAI-EXAONE/EXAONE-4.0-32B`

Table A.3: Hugging Face model repositories for all evaluated models. All checkpoints were accessed via the `transformers` library and are freely available for research use.