BITS Pilani at SemEval-2026 Task 9: Structured Supervised Fine-Tuning with DPO Refinement for Polarization Detection

Atharva Gupta[1] **Dhruv Kumar**[1] **Yash Sinha**[1]
[1]Birla Institute of Technology and Science, Pilani, India

{f20240519, dhruv.kumar, yash.sinha}@pilani.bits-pilani.ac.in

Abstract

The POLAR SemEval-2026 Shared Task aims to detect online polarization and focuses on the classification and identification of multilingual, multicultural, and multi-event polarization.

Accurate computational detection of online polarization is challenging due to nuanced rhetoric, implicit framing, and the high cost of human-in-the-loop annotation. Building on recent findings that contextual prompting enables large language models to function as strong polarization detectors, we present a two-stage approach for detecting polarization in social media text that combines structured supervised fine-tuning with Direct Preference Optimization (DPO) refinement.

We fine-tune Qwen 2.5-7B-Instruct with LoRA using an interpretable slot-filling template (target, claim type, manifestation checklist, and justification). We then apply DPO with automatically generated preference pairs to reduce costly false negatives. Our submitted system achieves 0.7664 Macro-F1 on the English test set. Post-submission experiments with Mistral-Nemo-Instruct-2407 and LLM-judge-filtered preference pairs further improve to 0.8162 Macro-F1 (not submitted to CodaBench), surpassing the organiser baseline of 0.7802.

1 Introduction

Large Language Models (LLMs) are rapidly being integrated across domains, from scientific research to customer service and policy analysis (Bommasani et al., 2022). Their capacity for large-scale language understanding has made them central to tasks such as semantic evaluation, stance detection, and content moderation (Devlin et al., 2019; Zhang et al., 2024; Pangtey et al., 2025).

In particular, LLMs have transformed online discourse analysis by enabling fine-grained interpretation of meaning, context, and intent at scale, surpassing earlier rule-based and shallow machine learning approaches (Franceschelli and Musolesi, 2025; Li et al., 2024). Online discourse increasingly reflects strong ideological divides, making polarization detection an important task for moderating digital communication (Loru et al., 2025).

In this paper, we study polarization detection in online discourse and describe a two-stage approach that combines structured supervised fine-tuning with preference-based refinement. Our goal is to improve Subtask 1 classification performance, particularly Macro-F1, while keeping the system efficient. Our submission targets Subtask 1 (Polarization Detection) in the English language only; all experiments, development evaluations, and the final task submission are conducted exclusively on the English portion of the POLAR dataset.

In summary, our contributions are:

- We formulate SemEval-2026 Task 9 (POLAR) Subtask 1 (Naseem et al., 2026a,b) as binary classification and augment predictions with a structured rationale (target, claim type, manifestation checklist, and justification) using a rigid slot-filling schema to reduce output variance.
- We apply Direct Preference Optimization (DPO) (Rafailov et al., 2024) with automatically constructed preference pairs to discourage overly conservative predictions and reduce false negatives.
- We show through ablation that structured reasoning generation and preference-based refinement are mutually reinforcing: the rationale enables more effective DPO pair construction, and post-submission experiments confirm this synergy scales with model capacity.

*Code available at https://github.com/atharva7-g/POLAR-SemEval-Submission

Proceedings of the The 20th International Workshop on Semantic Evaluation (2026), pages 3068–3079

- Post-submission experiments with Mistral-Nemo-Instruct-2407 and LLM-judge-filtered preference pairs improve Macro-F1 to 0.8162, surpassing the organiser baseline of 0.7802.

2 Related Work

Polarization and related abuse detection are widely studied research areas (Naseem et al., 2026b). Earlier shared-task work includes SemEval-2019 HatEval, which provides multilingual resources and baselines for related hate and abuse detection (Basile et al., 2019).

Methodologically, recent work explores LLM adaptation via prompting and parameter-efficient fine-tuning, finding that fine-tuning often outperforms in-context learning on polarization-adjacent tasks (Maggini et al., 2025). Complementary to standard cross-entropy fine-tuning, supervised contrastive objectives have been shown to improve robustness and generalization of pre-trained language model classifiers, especially in low-data settings (Gunel et al., 2021). Sucu et al. (2025) further demonstrate that the addition of contextual information substantially improves stance detection accuracy.

We extend this line of work by jointly enforcing a structured output schema and applying DPO refinement (Wang et al., 2024) with automatically generated preference pairs to reduce false negatives, combining the benefits of structured fine-tuning and preference-based optimization.

3 Method

3.1 Problem setup

We address Subtask 1 (Polarization Detection) of SemEval-2026 Task 9 (POLAR) (Naseem et al., 2026b), formulated as binary classification of social media posts as polarized (y=1) or not (y=0). In addition to the label, we generate a structured rationale: a target group, claim type, and a 6-category manifestation checklist (Stereotype, Vilification, Dehumanization, Extreme Language, Lack of Empathy, Invalidation) adopted from the organizers' Subtask 3 scheme (POLAR Task Organizers, 2026), plus a free-form justification.

3.2 Approach

We treat polarization detection as a generative task in which the model outputs both the label and a structured rationale. Our approach consists of a two-stage training pipeline: (i) structured supervised fine-tuning (SFT) to obtain a format-following polarization detector, and (ii) Direct Preference Optimization (DPO) refinement to improve decision quality, with a focus on reducing costly false negatives, while preserving the same output schema.

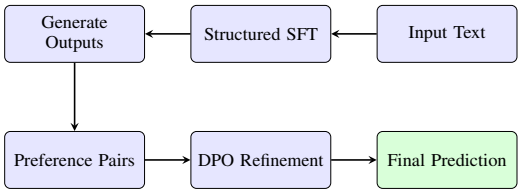

Figure 1: Overview of our two-stage pipeline for Subtask 1 polarization detection: structured supervised fine-tuning (SFT) first generates schema-consistent outputs, and DPO refinement then learns from preference pairs to produce the final prediction.

Stage 1: Structured supervised fine-tuning (SFT). We fine-tune an instruction-tuned LLM to emit a single structured record that includes the binary label and a rationale, enforcing a fixed output schema for consistency.

Stage 2: Preference-based refinement with DPO. Starting from the SFT checkpoint, we apply Direct Preference Optimization (DPO) (Rafailov et al., 2024) to refine the decision boundary under the same output schema.

Preference pair construction. We build pairs by contrasting model outputs that (i) correctly classified content versus (ii) plausible but overly conservative outputs that wrongly label the instance. We employ categories obtained from the Subtask 3 manifestation indicators to prioritize outputs that recognize explicit vilification, dehumanization, or extreme language.

4 Experiments

4.1 Experimental setup

Dataset We evaluate our approach on the POLAR @ SemEval-2026 dataset released by the task organizers (Naseem et al., 2026b; POLAR Task Organizers, 2026). Although the full dataset spans 22 languages, this work focuses exclusively on the English subset; the non-English portions of the dataset were not processed using the method described here, and our final task submission covered English only. The *train* and *validation* splits were released before evaluation. We use these official train and validation splits as-is and report the En-

Split	Total	Polarized	Non-polarized
Train	3,222	1,175 (36.5%)	2,047 (63.5%)
Dev	160	59 (36.9%)	101 (63.1%)
Test	1,452	533 (36.7%)	919 (63.3%)
Total	4,834	1,767 (36.6%)	3,067 (63.4%)

Table 1: English dataset statistics and label distribution across official splits.

glish subset statistics in Table 1.

Baseline fine-tuning pipeline. We establish a simple fine-tuning baseline by splitting the data into train/validation/test (80/10/10), fine-tuning a classifier with LoRA adapters, and evaluating with macro precision, recall, and F1.

4.1.1 Stage 1: Structured Supervised Fine-Tuning

Base Model We initialize with the Qwen 2.5-7B-Instruct model (Team, 2024), a decoder-only large language model. Structured SFT training data was generated using Gemma 3 27B (Gemma Team et al., 2025) served via Ollama, using the slot-filling prompt template described in Appendix A.6.

Parameter-Efficient Fine-Tuning We fine-tune with LoRA (Hu et al., 2021) on Qwen/Qwen2.5-7B-Instruct using attention projections only (full hyperparameters in Appendix Table 11).

To address class imbalance, we also test with a weighted loss upweighting polarized examples with class weights [1.0, 1.5].

Generation Protocol Each prompt follows the format below; the model generates a structured rationale followed by a binary label, which is extracted via regex.

```
Input: {text}
Reasoning:
```

4.1.2 Stage 2: Preference-based refinement with DPO

We apply DPO to refine the SFT model by learning from preference pairs that distinguish higher-quality reasoning and correct labels from weaker or incorrect outputs. We specifically try to address SFT's tendency to produce false negatives.

DPO optimizes the model to prefer chosen responses over rejected ones via a contrastive loss, implicitly learning a reward signal without requiring a separate reward model (Rafailov et al., 2024). Compared to other RLHF techniques such as GRPO (Shao et al., 2024), DPO is simpler to implement, more stable, and computationally lightweight (Rafailov et al., 2024).

Method	English Dev F1
Zero-shot baseline	0.7105
SLMs	0.7149
SFT	0.738
SFT + DPO	0.7893

Table 2: F1 scores on the English development set. SLMs = Small Language Models baseline (DistilBERT fine-tuned for sequence classification). Weighted loss during SFT did not improve performance.

Preference Pair Creation. For each input, we sample multiple SFT completions at varied decoding temperatures to increase output diversity, then assign each completion one of three labels. When an input yields heterogeneous labels, we construct preference pairs by ranking completions according to the labels above:

$$\text{CORRECT} \succ \text{FP} \succ \text{FN},$$

treating higher-ranked completions as *chosen* and lower-ranked ones as *rejected*. We rank false negatives below false positives because missed polarized content poses greater harm in moderation contexts than over-flagging: an undetected polarizing post may spread unchecked, whereas a false positive can be reviewed and reversed. This ordering prioritizes recall recovery: SFT alone captures only half of polarized instances, making false negatives the dominant failure mode to correct.

After the release of the leaderboards, we also test a new pair-generation variant that uses two prompt settings per input rather than a single prompt: one that encourages a prediction of 1 and another that encourages a prediction of 0. The resulting outputs are then processed with the same ranking rule to determine *chosen* and *rejected* responses.

Training configuration. We report full DPO hyperparameters in Appendix Table 12.

5 Results

Tables 2 and 4 report results on the **English development set** using the 80/10/10 split described above.

Tables 2 and 4 summarizes our English development results. As a lightweight baseline, we fine-tune DistilBERT (Sanh et al., 2019) for binary sequence classification (referred to as SLMs in Table 2); this serves as a reference point for the cost of supervised adaptation without instruction tuning or structured rationale generation. The zero-shot

System	Rank (out of 60)	F1
Highest-ranked system	1	0.8252
POLAR baseline	47	0.7802
Our system	52	0.7664

Table 3: Unofficial English Subtask 1 leaderboard comparison (60 teams).

Metric	SFT	SFT + DPO
Accuracy	0.7812	0.8000
Precision	0.8333	0.7077
Recall	0.5085	0.7797
F1 (Binary)	0.6316	0.7419
F1 (Macro)	0.7380	0.7893
F1 (Micro)	0.7812	0.8000

Table 4: English development set metrics comparing SFT vs. SFT + DPO.

baseline starts at 0.7105 F1 on the dev set. The DistilBERT baseline reaches 0.7149, marginally above zero-shot. SFT further improves performance to 0.738, while DPO yields the strongest dev performance at 0.7893.

On the English test set provided by the organizers, the SFT + DPO model reaches 0.7664 F1, indicating that preference refinement improves generalization beyond the development set.

Weighted-loss supervised fine-tuning (SFT) did not outperform standard SFT in our experiments, suggesting that class imbalance was not the primary bottleneck. We therefore report unweighted SFT as the primary baseline.

Table 3 presents the ranking comparison between our system, the POLAR baseline, and the highest-ranked system.

Table 4 highlights that DPO substantially increases recall (0.5085 $\rightarrow$ 0.7797), which reduces false negatives, but this comes with a drop in precision (0.8333 $\rightarrow$ 0.7077). We can explain this pattern by considering that DPO shifts the decision boundary toward preferring polarized outputs in borderline cases, increasing sensitivity at the cost of more false positives. The corresponding gains in F1 (binary and macro) indicate improved sensitivity to polarized content.

A qualitative example showing DPO correcting a false negative is provided in Appendix Table 10.

6 Experiments Post CodaBench Submission

After the release of the official rankings and leaderboard on GitHub, we trained both SFT and SFT + DPO on the full training set. For the DPO runs, we also experimented with LLM-as-a-judge filtering of preference pairs using DeepSeek-R1 (DeepSeek-AI, 2025), following the same filtering procedure described in Section 6.3. Full details in A.7.

Tables 7 and 6 report results on the **official English test set** (n=1,452), using models trained on the full official training split. When trained on all 3,222 examples from the official English training split, SFT achieved an F1 score of 0.7712 after 3 epochs and 0.7795 after 10 epochs on the test set. Applying DPO to the best-performing SFT model resulted in an F1 score of 0.7889.

Training data was re-validated by Claude 3.5 Sonnet (hereafter *Rejudged Sonnet* dataset), and DPO preference pairs were filtered using DeepSeek-R1 (DeepSeek-AI, 2025) as an LLM judge, yielding a balanced 62:38 FP:FN ratio.

Rejudged Sonnet dataset. The Rejudged Sonnet dataset is derived from the official English training split (3,222 examples) by re-validating every label with Claude 3.5 Sonnet as an LLM judge. Reasoning for each example was generated by GPT OSS 120B Nitro; Claude 3.5 Sonnet then evaluated the generated reasoning and revised labels where the reasoning did not support the original annotation. Each label was then also manually reviewed. Of the 3,178 training examples with an exact text match in the rejudged set, 196 (6.2%) received a revised label, finally producing a net increase in the proportion of polarized examples.

Model	**mF1**	**Acc.**	**P(1)**	**R(1)**
Mistral-Nemo	0.7963	0.8030	0.7928	0.8121
Qwen2.5-7B	0.7835	0.7899	0.7811	0.8006

Table 5: Label-only SFT performance on the English test set (n = 1,452) trained on the original unmodified labels without the Rejudged Sonnet dataset.

6.1 DPO preference pair data

Our DPO dataset is built automatically from SFT generations and is designed to preserve the same structured response format used during supervised fine-tuning. For details on how these pairs are constructed, please see Section 4.1.2.

The following analysis was conducted after the task ranking was released and was not used to generate the leaderboard predictions.

Using the two-prompt method at varied temperatures, we generate 721 candidate pairs, filtered to 330 by deduplication and length ratio (Table 6). More pairs reduce FNs but degrade overall F1 due

Mode	F1	FNs	FPs
SFT	0.7795	158	137
SFT with DPO (330 pairs)	0.7889	132	155
SFT with DPO (721 pairs)	0.7637	64	274

Table 6: Effect of DPO preference pair count on F1, false negatives (FNs), and false positives (FPs). SFT baseline is the 10-epoch full-training-set model; development-set results appear in Tables 2 and 4.

Condition	Acc.	P(1)	R(1)	mF1
Label-only SFT	0.792	0.777	0.7884	0.781
Label-only + DPO	0.720	0.618	0.625	0.699
Reasoning SFT	0.793	0.745	0.662	0.771
Reasoning + DPO	0.802	0.732	0.704	0.789

Table 7: Structured rationale ablation on the English test set. P(1) and R(1) = precision and recall for the polarized class; mF1 = Macro-F1. Best in bold.

to noise, confirming that LLM-as-a-judge quality control (Yu et al., 2025) is needed before training.

Preference pairs for post-submission DPO training on Mistral-Nemo were constructed exclusively from inputs where the Mistral-Nemo SFT checkpoint produced incorrect outputs. For each such input, the two-prompt method was applied using Llama-3.3-70B (Grattafiori et al., 2024), yielding a pool of candidate responses. These candidates were ranked according to the ordering CORRECT $\succ$ FP $\succ$ FN.

The resulting pairs were then evaluated by DeepSeek-R1 (DeepSeek-AI, 2025), acting as an LLM-based judge, to remove low-confidence or inconsistent comparisons. This filtering step produced a final dataset of 299 preference pairs, with a 62:38 ratio of FP to FN cases. Full construction details and prompts are provided in Appendix A.7.

6.2 Structured Rationale Ablation

To assess whether the structured rationale contributes to classification performance, we compare four conditions in a controlled ablation on the English test set (n = 1,452): label-only SFT, label-only SFT followed by our reasoning DPO setup (label-only+DPO), reasoning SFT, and reasoning SFT+DPO. All SFT conditions use 3 epochs, the same base model, LoRA configuration, and training data.

Table 7 shows that label-only SFT outperforms reasoning SFT in isolation (Macro-F1: 0.7811 vs. 0.771), consistent with findings that SFT on full chain-of-thought sequences dilutes gradient signal on the final label token (Shi et al., 2025). However, label-only SFT+DPO collapses to 0.699, while reasoning SFT+DPO achieves the best result (0.789), showing that the rationale's value is in enabling DPO rather than improving the SFT decision boundary. These ablations use Qwen2.5-7B-Instruct; post-submission results (Section 6.3) show that reasoning data does benefit the larger Mistral-Nemo model.

System	mF1
Qwen2.5-7B Label-Only SFT (Rejudged)	0.7811
Mistral-Nemo Label-Only SFT (Rejudged)	0.8019
Mistral-Nemo SFT (Rejudged)	0.8097
Mistral-Nemo DPO (FP:FN 62:38, β=0.3)	0.8162

Table 8: Post-submission results on the English test set (n=1,452). None submitted to CodaBench. mF1 = Macro-F1.

6.3 Post-submission Results

All experiments here were conducted after the official submission deadline and were not submitted to CodaBench. A full beta sweep and additional analysis are in Appendix A.1.

Post-submission experiments replaced Qwen2.5-7B-Instruct with Mistral-Nemo-Instruct-2407 (Mistral AI, 2024), a 12B model with strong multilingual instruction-following.

Table 8 shows that Mistral-Nemo SFT with Rejudged Sonnet data (0.8097) surpasses both the organiser baseline (0.7802) and the label-only variant (0.8019), confirming that structured reasoning data provides a meaningful training signal for the larger model. Applying quality-filtered DPO at β=0.3 further improves to 0.8162 (Polar-SemEval, 2026), up from 0.7664 (rank 52) for the submitted system.

Results of the beta sweep over nine β values (0.1–0.5) are reported in Appendix Table 9.

7 Conclusion

We presented a two-stage system for polarization detection combining structured SFT with DPO refinement. The submitted system, based on Qwen2.5-7B-Instruct, achieved 0.7664 Macro-F1 on the English test set.

Post-submission improvements (Section 6.3) raise performance to 0.8162 Macro-F1: surpassing the organiser baseline (0.7802) on the unofficial English leaderboard (Polar-SemEval, 2026).

Limitations

We observed a subset of ambiguous examples in the training dataset where cues were mixed or context-

dependent, and annotator intent was not always clear from the text alone. These cases were especially challenging for both SFT and DPO, leading to inconsistent predictions. The structured rationale does not fully resolve this ambiguity.

A limitation of our submitted system is that, for Qwen2.5-7B-Instruct, structured rationales do not outperform a label-only baseline. Label-only SFT achieves 0.7811 Macro-F1 versus 0.771 for reasoning SFT. At this scale, rationales mainly recover performance lost during reasoning-based fine-tuning via preference optimization, rather than improving it. Post-submission results indicate this is a scale effect, with clearer gains from structured reasoning emerging at 12B parameters.

The post-submission experiments use a revalidated training set that differs from the data used for the official submission; improvements over the submitted system therefore reflect both the stronger base model and the change in training labels.

Precision remains relatively low across all configurations using rationale-based SFT (best: 0.76), reflecting the subtle boundary between emphatic-but-neutral language and genuinely polarizing content, as well as the model's tendency toward high recall.

Future Work

Post-submission results confirm that preference pair quality is the primary DPO bottleneck. Future work should investigate loss masking (Shi et al., 2025) to recover the recall cost of reasoning SFT, and explore SimPO (Meng et al., 2024) or KTO (Ethayarajh et al., 2024) as more stable preference optimisation alternatives. The precision-recall trade-off and the causes of DPO instability in smaller models warrant further study — in particular, whether reference-free objectives can stabilise DPO for 7B-scale models. Extending the pipeline to non-English languages in the POLAR benchmark (Naseem et al., 2026b) is a direct next step, as is evaluating whether the Rejudged Sonnet quality improvement transfers to other base models. Generating reasoning chains using a stronger judge model (e.g. GPT-4o or Claude 3.7 Sonnet) could yield higher-quality training data and further improve both SFT and DPO performance. Sourcing additional training data from related polarization and hate-speech datasets (Basile et al., 2019) would also increase coverage of underrepresented manifestation types such as subtle Invalidation.

References

Valerio Basile, Cristina Bosco, Elisabetta Fersini, Debora Nozza, Viviana Patti, Francisco Manuel Rangel Pardo, Paolo Rosso, and Manuela Sanguinetti. 2019. SemEval-2019 task 5: Multilingual detection of hate speech against immigrants and women in Twitter. In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 54–63, Minneapolis, Minnesota, USA. Association for Computational Linguistics.

Rishi Bommasani, Drew A. Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S. Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, Erik Brynjolfsson, Shyamal Buch, Dallas Card, Rodrigo Castellon, Niladri Chatterji, Annie Chen, Kathleen Creel, Jared Quincy Davis, Dora Demszky, Chris Donahue, Moussa Doumbouya, Esin Durmus, Stefano Ermon, John Etchemendy, Kawin Ethayarajh, Li Fei-Fei, Chelsea Finn, Trevor Gale, Lauren Gillespie, Karan Goel, Noah Goodman, Shelby Grossman, Neel Guha, Tatsunori Hashimoto, Peter Henderson, John Hewitt, Daniel E. Ho, Jenny Hong, Kyle Hsu, Jing Huang, Thomas Icard, Saahil Jain, Dan Jurafsky, Pratyusha Kalluri, Siddharth Karamcheti, Geoff Keeling, Fereshte Khani, Omar Khattab, Pang Wei Koh, Mark Krass, Ranjay Krishna, Rohith Kuditipudi, Ananya Kumar, Faisal Ladhak, Mina Lee, Tony Lee, Jure Leskovec, Isabelle Levent, Xiang Lisa Li, Xuechen Li, Tengyu Ma, Ali Malik, Christopher D. Manning, Suvir Mirchandani, Eric Mitchell, Zanele Munyikwa, Suraj Nair, Avanika Narayan, Deepak Narayanan, Ben Newman, Allen Nie, Juan Carlos Niebles, Hamed Nilforoshan, Julian Nyarko, Giray Ogut, Laurel Orr, Isabel Papadimitriou, Joon Sung Park, Chris Piech, Eva Portelance, Christopher Potts, Aditi Raghunathan, Rob Reich, Hongyu Ren, Frieda Rong, Yusuf Roohani, Camilo Ruiz, Jack Ryan, Christopher Ré, Dorsa Sadigh, Shiori Sagawa, Keshav Santhanam, Andy Shih, Krishnan Srinivasan, Alex Tamkin, Rohan Taori, Armin W. Thomas, Florian Tramèr, Rose E. Wang, William Wang, Bohan Wu, Jiajun Wu, Yuhuai Wu, Sang Michael Xie, Michihiro Yasunaga, Jiaxuan You, Matei Zaharia, Michael Zhang, Tianyi Zhang, Xikun Zhang, Yuhui Zhang, Lucia Zheng, Kaitlyn Zhou, and Percy Liang. 2022. On the opportunities and risks of foundation models.

DeepSeek-AI. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding.

Kawin Ethayarajh, Winnie Xu, Niklas Muennighoff, Dan Jurafsky, and Douwe Kiela. 2024. KTO: Model alignment as prospect theoretic optimization. In *Proceedings of the 41st International Conference on Machine Learning (ICML)*.

Giorgio Franceschelli and Mirco Musolesi. 2025. On

the creativity of large language models. *AI & society*, 40(5):3785–3795.

Gemma Team, Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej, Sarah Perrin, Tatiana Matejovicova, Alexandre Ramé, Morgane Rivière, Louis Rouillard, Thomas Mesnard, Geoffrey Cideron, Jean bastien Grill, Sabela Ramos, Edouard Yvinec, Michelle Casbon, Etienne Pot, Ivo Penchev, Gaël Liu, Francesco Visin, Kathleen Kenealy, Lucas Beyer, Xiaohai Zhai, Anton Tsitsulin, Robert Busa-Fekete, Alex Feng, Noveen Sachdeva, Benjamin Coleman, Yi Gao, Basil Mustafa, Iain Barr, Emilio Parisotto, David Tian, Matan Eyal, Colin Cherry, Jan-Thorsten Peter, Danila Sinopalnikov, Surya Bhupatiraju, Rishabh Agarwal, Mehran Kazemi, Dan Malkin, Ravin Kumar, David Vilar, Idan Brusilovsky, Jiaming Luo, Andreas Steiner, Abe Friesen, Abhanshu Sharma, Abheesht Sharma, Adi Mayrav Gilady, Adrian Goedeckemeyer, Alaa Saade, Alex Feng, Alexander Kolesnikov, Alexei Bendebury, Alvin Abdagic, Amit Vadi, András György, André Susano Pinto, Anil Das, Ankur Bapna, Antoine Miech, Antoine Yang, Antonia Paterson, Ashish Shenoy, Ayan Chakrabarti, Bilal Piot, Bo Wu, Bobak Shahriari, Bryce Petrini, Charlie Chen, Charline Le Lan, Christopher A. Choquette-Choo, CJ Carey, Cormac Brick, Daniel Deutsch, Danielle Eisenbud, Dee Cattle, Derek Cheng, Dimitris Paparas, Divyashree Shivakumar Sreepathihalli, Doug Reid, Dustin Tran, Dustin Zelle, Eric Noland, Erwin Huizenga, Eugene Kharitonov, Frederick Liu, Gagik Amirkhanyan, Glenn Cameron, Hadi Hashemi, Hanna Klimczak-Plucińska, Harman Singh, Harsh Mehta, Harshal Tushar Lehri, Hussein Hazimeh, Ian Ballantyne, Idan Szpektor, Ivan Nardini, Jean Pouget-Abadie, Jetha Chan, Joe Stanton, John Wieting, Jonathan Lai, Jordi Orbay, Joseph Fernandez, Josh Newlan, Ju yeong Ji, Jyotinder Singh, Kat Black, Kathy Yu, Kevin Hui, Kiran Vodrahalli, Klaus Greff, Linhai Qiu, Marcella Valentine, Marina Coelho, Marvin Ritter, Matt Hoffman, Matthew Watson, Mayank Chaturvedi, Michael Moynihan, Min Ma, Nabila Babar, Natasha Noy, Nathan Byrd, Nick Roy, Nikola Momchev, Nilay Chauhan, Noveen Sachdeva, Oskar Bunyan, Pankil Botarda, Paul Caron, Paul Kishan Rubenstein, Phil Culliton, Philipp Schmid, Pier Giuseppe Sessa, Pingmei Xu, Piotr Stanczyk, Pouya Tafti, Rakesh Shivanna, Renjie Wu, Renke Pan, Reza Rokni, Rob Willoughby, Rohith Vallu, Ryan Mullins, Sammy Jerome, Sara Smoot, Sertan Girgin, Shariq Iqbal, Shashir Reddy, Shruti Sheth, Siim Põder, Sijal Bhatnagar, Sindhu Raghuram Panyam, Sivan Eiger, Susan Zhang, Tianqi Liu, Trevor Yacovone, Tyler Liechty, Uday Kalra, Utku Evci, Vedant Misra, Vincent Roseberry, Vlad Feinberg, Vlad Kolesnikov, Woohyun Han, Woosuk Kwon, Xi Chen, Yinlam Chow, Yuvein Zhu, Zichuan Wei, Zoltan Egyed, Victor Cotruta, Minh Giang, Phoebe Kirk, Anand Rao, Kat Black, Nabila Babar, Jessica Lo, Erica Moreira, Luiz Gustavo Martins, Omar Sanseviero, Lucas Gonzalez, Zach Gleicher, Tris Warkentin, Vahab Mirrokni, Evan Senter, Eli Collins, Joelle Barral, Zoubin Ghahramani, Raia Hadsell, Yossi Matias, D. Sculley, Slav Petrov, Noah Fiedel, Noam Shazeer, Oriol Vinyals, Jeff Dean, Demis Hassabis, Koray Kavukcuoglu, Clement Farabet, Elena Buchatskaya, Jean-Baptiste Alayrac, Rohan Anil, Dmitry, Lepikhin, Sebastian Borgeaud, Olivier Bachem, Armand Joulin, Alek Andreev, Cassidy Hardin, Robert Dadashi, and Léonard Hussenot. 2025. Gemma 3 technical report.

Aaron Grattafiori et al. 2024. The llama 3 herd of models.

Beliz Gunel, Jingfei Du, Alexis Conneau, and Veselin Stoyanov. 2021. Supervised contrastive learning for pre-trained language model fine-tuning. In *International Conference on Learning Representations (ICLR)*.

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

Junyi Li, Tianyi Tang, Wayne Xin Zhao, Jian-Yun Nie, and Ji-Rong Wen. 2024. Pre-trained language models for text generation: A survey. *ACM Computing Surveys*, 56(9):1–39.

Edoardo Loru, Alessandro Galeazzi, Anita Bonetti, Emanuele Sangiorgio, Niccolò Di Marco, Matteo Cinelli, Max Falkenberg, Andrea Baronchelli, and Walter Quattrociocchi. 2025. Ideology and polarization set the agenda on social media. *Scientific Reports*, 15(1).

Michele Joshua Maggini, Dhia Merzougui, Rabiraj Bandyopadhyay, Gaël Dias, Fabrice Maurel, and Pablo Gamallo. 2025. Are llms enough for hyperpartisan, fake, polarized and harmful content detection? evaluating in-context learning vs. fine-tuning.

Yu Meng, Mengzhou Xia, and Danqi Chen. 2024. SimPO: Simple preference optimization with a reference-free reward. In *Advances in Neural Information Processing Systems (NeurIPS)*.

Mistral AI. 2024. Mistral nemo. `https://mistral.ai/news/mistral-nemo/`. Accessed: April 2026.

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, Dheeraj Kodati, Sahar Moradizeyveh, Firoj Alam, Ye Kyaw Thu, Shantipriya Parida, Ihsan Ayyub Qazi, Nelson Odhiambo Onyango, Clemencia Siro, Ibrahim Said Ahmad, Lilian Wanzare, Adem Chanie Ali, Martin Semmann, Chris Biemann, Shamsuddeen Hassan Muhammad, and Seid Muhie Yimam. 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 Acartürk, Aisha Jabr, Saba Anwar, Abinew Ali Ayele, Simona Frenda, Alessandra Teresa Cignarella, Elena Tutubalina, Oleg Rogov, Aung Kyaw Htet, Xintong Wang, Surendrabikram Thapa, Kritesh Rauniyar, Tanmoy Chakraborty, Arfeen Zeeshan, Dheeraj Kodati, Satya Keerthi, Sahar Moradizeyveh, Firoj Alam, Arid Hasan, Syed Ishtiaque Ahmed, Ye Kyaw Thu, Shantipriya Parida, Ihsan Ayyub Qazi, Lilian Wanzare, Nelson Odhiambo Onyango, Clemencia Siro, Jane Wanjiru Kimani, Ibrahim Said Ahmad, Adem Chanie Ali, Martin Semmann, Chris Biemann, Shamsuddeen Hassan Muhammad, and Seid Muhie Yimam. 2026b. Polar: A benchmark for multilingual, multicultural, and multi-event online polarization.

Lata Pangtey, Anukriti Bhatnagar, Shubhi Bansal, Shahid Shafi Dar, and Nagendra Kumar. 2025. Large language models meet stance detection: A survey of tasks, methods, applications, challenges and future directions.

Polar-SemEval. 2026. POLAR @ SemEval-2026 Leaderboards. https://github.com/Polar-SemEval/Leaderboards. Accessed: April 2026.

POLAR Task Organizers. 2026. Polar @ semeval-2026 task 9: Detecting multilingual, multicultural and multievent online polarization. Accessed: 2026-02-27.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, and Chelsea Finn. 2024. Direct preference optimization: Your language model is secretly a reward model. *arXiv preprint arXiv:2305.18290*.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxian Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y.K. Li, Y. Wu, et al. 2024. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*.

Xiaofeng Shi, Qian Kou, Yuduo Li, and Hua Zhou. 2025. Rethinking supervised fine-tuning: Emphasizing key answer tokens for improved llm accuracy.

Arman Engin Sucu, Yixiang Zhou, Mario A. Nascimento, and Tony Mullen. 2025. Exploiting contextual information to improve stance detection in informal political discourse with llms. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 4: Student Research Workshop)*, page 1097–1110. Association for Computational Linguistics.

Qwen Team. 2024. Qwen2.5: A party of foundation models.

Tianduo Wang, Shichen Li, and Wei Lu. 2024. Self-training with direct preference optimization improves chain-of-thought reasoning. *arXiv preprint arXiv:2407.18248*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, 35:24824–24837.

Jiachen Yu, Shaoning Sun, Xiaohui Hu, Jiaxu Yan, Kaidong Yu, and Xuelong Li. 2025. Improve llm-as-a-judge ability as a general ability.

Wenxuan Zhang, Yue Deng, Bing Liu, Sinno Pan, and Lidong Bing. 2024. Sentiment analysis in the era of large language models: A reality check. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 3881–3906, Mexico City, Mexico. Association for Computational Linguistics.

A Appendix

A.1 Post-submission Experimental Detail

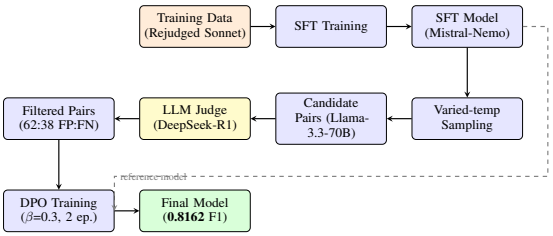

Figure 2: Post-submission pipeline. Rejudged Sonnet training data produces an SFT checkpoint, which then generates completions at varied temperatures to form candidate preference pairs. DeepSeek-R1 filters these into a balanced set (62:38 FP:FN ratio). DPO then refines the same SFT checkpoint (dashed arrow) using the filtered pairs, producing the best post-submission result of 0.8162 Macro-F1. The Rejudged Sonnet training data was produced by generating reasoning with GPT OSS 120B Nitro and filtering labels with Claude 3.5 Sonnet as a judge (see Appendix A.7).

The post-submission setup is described in Section 6.3. We provide stability and sensitivity details below.

Across 20 DPO runs on Mistral-Nemo, using stale preference pairs, unfiltered pairs, or excessive training epochs consistently degraded performance. Epoch count is especially critical: 10 epochs on 260 pairs collapsed to Macro-F1 0.6508 vs. 0.8012 at 2 epochs.

A subsequent beta sweep over 9 values (0.1–0.5) on the same pair set confirms robustness: all configurations beat the SFT baseline (0.8097), with Macro-F1 ranging from 0.8065 to 0.8142 and no single beta dominating clearly (Table 9). This flatness suggests pair quality and composition (FP:FN ratio) are the binding constraints, not β.

β	mF1	Acc.	P(1)	R(1)
0.10	0.8127	0.8230	0.739	0.801
0.15	0.8125	0.8230	0.740	0.799
0.20	0.8136	0.8237	0.738	0.805
0.25	0.8108	0.8209	0.734	0.803
0.30	**0.8162**	0.8230	0.736	0.809
0.35	0.8079	0.8182	0.731	0.799
0.40	0.8071	0.8189	0.727	0.811
0.45	0.8142	0.8258	0.737	0.816
0.50	0.8065	0.8223	0.733	0.811
SFT baseline	0.8097	0.8182	0.719	0.827

Table 9: Beta sweep on 299 DeepSeek-R1-filtered DPO pairs (62:38 FP:FN ratio), Mistral-Nemo-Instruct-2407, 2 epochs. mF1 = Macro-F1; P(1)/R(1) = precision/recall for the polarized class.

A.2 Qualitative Example

Table 10 shows a representative case from the English test set where SFT produced a false negative that DPO subsequently corrected. The SFT model marked all manifestation categories as absent and assigned label 0. After DPO refinement, the model identified *Stereotype*, *Vilification*, *Lack of Empathy*, and *Invalidation* as present, driving the correct label-1 prediction. This suggests that the structured manifestation checklist is doing real work in the DPO model: the label change is anchored to specific checklist fields rather than arising from an unconstrained shift in output distribution.

A.3 Hyperparameters

Post-submission experiments use Mistral-Nemo-Instruct-2407 as the base model. We attribute its stronger response to reasoning-based SFT — relative to Qwen2.5-7B-Instruct — to its larger parameter count (12B vs. 7B), explicit multilingual instruction-following training, and greater capacity to jointly learn a structured output schema and the classification objective. Larger models are known to benefit more from chain-of-thought style supervision (Wei et al., 2022).

Table 11 reports the SFT hyperparameters.

Table 12 provides the complete DPO hyperparameter configuration.

Input text (`eng_03a6df98`)
"Ending 2024 with an illadvised text conversation with my brother that ended with 'Really little bro? I can deal with the atheism, but socialism?"'
Gold label: 1 (Lack of Empathy: present; Invalidation: present)
SFT output (label: 0 — false negative)
Target: specific individual Claim type: evaluative opinion
Stereotype: absent Vilification: absent
Dehumanization: absent Extreme Language: absent
Lack of Empathy: **absent** Invalidation: **absent**
Decision basis: The text is a personal, non-political anecdote about a family disagreement that does not contain any of the defined polarization manifestations.
Final Answer: **0**
SFT + DPO output (label: 1 — corrected)
Target: **specific group** Claim type: evaluative opinion
Stereotype: **present** Vilification: **present**
Dehumanization: absent Extreme Language: absent
Lack of Empathy: **present** Invalidation: **present**
Decision basis: The text vilifies socialism by framing it as an invalid or illegitimate belief and invalidates the opposing viewpoint by dismissing it as "illadvised"; the conditional acceptance structure ("I can deal with X, but not Y") constitutes Lack of Empathy.
Final Answer: **1**

Table 10: Qualitative comparison of SFT (Mistral-Nemo baseline) vs. best DPO model (299 DeepSeek-R1-filtered pairs, β=0.3) on a false-negative example from the test set. Bold fields differ between the two outputs. SFT misreads the implicit ideological framing as a personal anecdote and predicts 0. DPO recovers the correct label (1) by identifying Lack of Empathy and Invalidation, though it also flags Stereotype and Vilification as present, which are absent in the gold annotation.

Parameter	Value
Max Length	1024
Train Batch Size	1
Eval Batch Size	4
Gradient Accumulation Steps	8
Learning Rate	5e-5
Num Epochs	3
Warmup Ratio	0.0
Precision	BF16
Optimizer	adamw_torch
Class Weights	[1.0, 1.5]
LoRA Rank (r)	8
LoRA Alpha	16
LoRA Dropout	0.05
LoRA Target Modules	q_proj, k_proj, v_proj, o_proj

Table 11: Pre-submission SFT hyperparameters (Qwen2.5-7B-Instruct). Class weights [1.0, 1.5] were used in both pre- and post-submission runs but did not massively improve performance over unweighted training (see Section 5).

Parameter	Value
Max Length	1024
Max Prompt Length	512
Train Batch Size	1
Gradient Accumulation Steps	8
Learning Rate	5e-6
Num Epochs	2
β **(submitted)**	0.1
β **(post-submission)**	0.3
Warmup Ratio	0.0
Precision	BF16

Table 12: DPO hyperparameters. Pre-submission uses Qwen2.5-7B-Instruct with β=0.1; post-submission uses Mistral-Nemo-Instruct-2407 with β=0.3.

A.4 DPO Subset Sweep

To examine whether the FP:FN ratio in the preference pair set affects DPO performance, we construct the subsets using stale preference pairs. We subsampled the class-0 (FP-correcting) pairs while keeping all class-1 (FN-correcting) pairs fixed, yielding three smaller pair sets. All experiments use the same hyperparameters (Mistral-Nemo, β = 0.1, 2 epochs, lr = 5e-6) on the SFT Rejudged Sonnet checkpoint.

Configuration	FP:FN	Total	mF1	Recall
SFT baseline	—	—	0.8097	0.827
Original 190/70	73:27	260	0.8012	0.747
Subset 150/70	68:32	220	0.8024	0.745
Subset 120/70	63:37	190	0.8005	0.779
Subset 100/70	59:41	170	0.7986	0.739

Table 13: DPO subset sweep on the stale unfiltered preference pairs: effect of subsampling class-0 (FP-correcting) pairs while holding class-1 (FN-correcting) pairs fixed at 70. Section 6.3), confirming that pair quality and volume rather than FP:FN ratio alone drive the performance gap. Matching the test-set class distribution (120/70, 63:37) did not improve performance.

The 150/70 split achieved the best DPO result (0.8024), but sits much below the best-performing DPO system (0.8162). This gap persists despite the more balanced pair ratio, reinforcing the conclusion that LLM-judge quality filtering — not pair composition alone — is the key factor separating the best DPO configurations from the rest.

A.5 Label-Only SFT: Training Details

Label-only supervised fine-tuning trains the model to predict the final binary classification label directly, without generating intermediate reasoning steps. This serves as an important baseline for assessing the contribution of structured rationale generation.

Training Data. We use the full official English training split (3,222 examples) with the original annotations, without any re-validation or relabeling.

Input Format. Each training example follows a simplified two-field template:

```
Input:
<text>
Final Answer:
<label>
```

Training Configuration. We fine-tune with LoRA using the same adapter configuration as the reasoning SFT baseline (rank 8, alpha 16, dropout 0.05, attention projections only). Standard causal language modelling loss is applied across all tokens in the sequence; no loss masking is used. Key hyperparameters are listed in Table 14.

Parameter	Value
Base Model	Qwen/Qwen2.5-7B-Instruct
Training Data	3,222 samples (original)
Max Length	1024
Batch Size	8
Gradient Accumulation	1
Learning Rate	5e-5
Num Epochs	10
Precision	BF16
Optimizer	AdamW
LoRA Rank (r)	8
LoRA Alpha	16
LoRA Dropout	0.05
LoRA Target Modules	q_proj, k_proj, v_proj, o_proj
Loss	Standard causal LM

Table 14: Label-only SFT hyperparameters (Qwen2.5-7B-Instruct).

A.6 SFT training data generation prompt

The supervised fine-tuning dataset was generated using the Gemma 3 27B model served via Ollama's inference framework (Gemma Team et al., 2025). The prompt below is an excerpt from the template employed during data generation. A very similar prompt was used to generate the DPO dataset with varying decoding temperatures; see Section 4.1.2.

```
Input:
<copy the input text verbatim>

Reasoning:
Target referenced: <specific
individual / specific group /
 ↪  none>
```

```
Claim type: <factual description
 ↪  / moral judgment / evaluative
 ↪  opinion / call to action /
 ↪  other>
Manifestations present:
- Stereotype: <present / absent>
- Vilification: <present /
 ↪  absent>
- Dehumanization: <present /
 ↪  absent>
- Extreme Language and
 ↪  Absolutism: <present /
 ↪  absent>
- Lack of Empathy or
 ↪  Understanding: <present /
 ↪  absent>
- Invalidation: <present /
 ↪  absent>
Decision basis:
<one factual sentence explaining
 ↪  how the listed manifestations
 ↪  determine whether the text is
 ↪  polarized>

Final Answer: <output ONLY 0 or
 ↪  1>
```

A.7 Preference Pair Generation: Methods and Prompts

Across all experiments, preference pairs were constructed using the same mixed-outcome extraction procedure: for each training input, multiple completions were sampled at varied decoding temperatures to increase output diversity, classified as CORRECT, FP, or FN relative to the ground truth label, and paired according to the ranking CORRECT $\succ$ FP $\succ$ FN. The inference model used for completion generation differed across experimental stages.

Submitted System. Completions were sampled using Gemma 3 27B (Gemma Team et al., 2025) served via Ollama at varied temperatures. Preference pairs were then extracted from examples with mixed outcomes across completions.

Post-Ranking Qwen Experiments. The completions were sampled using GPT OSS 120B using the OpenRouter API at temperatures 0.6, 0.9, and 1.2, generating six completions per example (two prompt strategies $\times$ three temperatures). Prompt A argued in favor of a polarized label; Prompt B argued against it.

Examples where all completions had the same outcome were excluded.

Post-Submission Mistral-Nemo Experiments. The completions were sampled with Llama-3.3-70B (Grattafiori et al., 2024) from the Mistral-Nemo SFT checkpoint at various temperatures using the same two-prompt strategy. The resulting candidate pairs were then filtered using DeepSeek-R1 (DeepSeek-AI, 2025) as an LLM judge, yielding the final 62:38 FP:FN balanced pair set used for DPO training at β=0.3.

Rejudged Sonnet Judging Prompt. Each training example was evaluated using the following prompt, submitted to Claude 3.5 Sonnet. Reasoning for each example was first generated by GPT OSS 120B Nitro; the judging prompt below was then used to assess whether the generated reasoning supported the original gold label.

```
You are auditing generated
 ↪  rationale data for political
polarization training.
Task: Decide if this sample is
 ↪  acceptable for training data.
Acceptable means:
1. Reasoning is coherent and
 ↪  supports the final label.
2. Final Answer is 0 or 1 and
 ↪  matches GOLD_LABEL.
3. Required structure is present:
 ↪  Input, Reasoning,
   Final Answer.
4. Reasoning includes these slots
 ↪  in substance:
   * Target referenced
   * Claim type
   * Manifestations present (6
    ↪  fields)
   * Decision basis
5. Claim type should be one of:
 ↪  factual description /
   moral judgment / evaluative
    ↪  opinion / call to action /
   other
Important:
* Do NOT fail only for minor
 ↪  formatting issues (e.g., one
  newline instead of two before
   ↪  Final Answer).
* Focus on semantic correctness
 ↪  and training usefulness.
```

```
Return strict JSON only with
 ↪  exactly keys:
\{"valid": bool, "confidence":
 ↪  1-5, "errors":
 ↪  ["error_code"],
"notes": "short"\}
Allowed error codes:
 ↪  missing_section,
 ↪  missing_slot,
invalid_claim_type,
 ↪  invalid_label,
 ↪  label_mismatch,
contradictory_reasoning,
 ↪  weak_decision_basis,
format_violation

INPUT_TEXT: \{input_text\}
GOLD_LABEL: \{gold_label\}

GENERATED_SAMPLE:
 ↪  \{generated_sample\}
```

Examples were retained in the rejudged set only when Claude 3.5 Sonnet returned `"valid": true`.