dutir_shlee at SemEval-2026 Task 11: Symbolic Augmentation for Content-Bias-Resistant Syllogistic Reasoning

Songhuan Li, Liang Yang*, Shengdi Yin, Qiang Zhang, Hongfei Lin
School of Computer Science and Technology
Dalian University of Technology, China
1341460619@mail.dlut.edu.cn, (liang,zhangq,hflin)@dlut.edu.cn

Abstract

We describe our system for SemEval-2026 Task 11 Subtask 1 (English syllogistic validity). Our approach fine-tunes Qwen2.5-7B-Instruct with LoRA and a symbolic data augmentation (SDA) scheme that replaces real-world entities with abstract placeholders, explicitly decoupling logical form from content. The resulting model achieves 96.34% accuracy and a total content effect (TCE) of 2.15, yielding a primary score of 44.86. We provide detailed ablations and negative results (prompting, self-consistency, contrastive decoding, structured chain-of-thought, and DPO) to characterize why direct LoRA training with SDA is the most robust configuration for this task. Finally, we use a specialist–generalist complementarity setting where a strong API model (ACC 99.48, TCE 1.06, score 57.68) is corrected by the SDA specialist on a single disagreement, producing a merged output with ACC 100 and TCE 0.

1 Introduction

SemEval-2026 Task 11 targets content effects in logical reasoning by requiring models to judge syllogistic validity independently of plausibility. We participate only in Subtask 1, which is English-only binary validity classification. This task is important because belief bias is a persistent failure mode for LLMs in high-stakes reasoning settings, and the evaluation explicitly penalizes content-driven errors. The official task overview is given in (Valentino et al., 2026).

Our main strategy is symbolic data augmentation (SDA) combined with parameter-efficient LoRA fine-tuning. By replacing concrete entities with abstract placeholders, SDA decouples logical form from surface content and forces the model to attend to quantifiers and negation structure. We then fine-tune a Qwen2.5-7B-Instruct base model using LoRA and run deterministic inference with a simple label-only output.

Through participation, we found that the LoRA+SDA system achieves 96.34% accuracy with TCE 2.15 (score 44.86), while chain-of-thought prompting, self-consistency, and DPO did not improve bias robustness. Remaining errors concentrate on quantifier-scope and existential/universal edge cases. We did not track a shared-task leaderboard rank for this report.

2 Background and Task Setup

Each instance contains an English syllogism and a binary validity label. Inputs are short, controlled arguments in syllogistic form; outputs are *Valid* or *Invalid*. For example: "All A are B. No B are C. Therefore, no A are C." → *Valid*. We participate only in Subtask 1 (English binary classification). The official training set contains 960 items and the test split contains 191 items; we report results on the official splits without re-partitioning.

Performance is measured by overall accuracy (ACC) and total content effect (TCE), with the primary metric:

$$\frac{\text{ACC}}{1 + \ln(1 + \text{TCE})}.$$

Lower TCE indicates stronger robustness to content bias. Prior work shows that LLMs systematically exhibit content effects on syllogisms (Dasgupta et al., 2022; Bertolazzi et al., 2024; Valentino et al., 2025; Kim et al., 2025). Related work evaluates deductive competence and syllogistic reasoning across settings and datasets (Seals and Shalin, 2024; Ozeki et al., 2024; Eisape et al., 2024; Wysocka et al., 2025), and explores faithfulness and quasi-symbolic reasoning for chain-of-thought and explanations (Lyu et al., 2023; Xu et al., 2024; Quan et al., 2024; Ranaldi et al., 2025). Our system focuses on eliminating content cues through SDA, encouraging a representation aligned with abstract logical form rather than world knowledge.

Proceedings of the The 20th International Workshop on Semantic Evaluation (2026), pages 509–513

Algorithm 1 LoRA + SDA Training

Require: Training set $\mathcal{D}$, base model θ, symbol map ϕ
Ensure: LoRA adapter Δ
1: $\mathcal{D}_{\text{sda}} \leftarrow \{(\phi(x), y) \mid (x, y) \in \mathcal{D}\}$
2: $\mathcal{D}' \leftarrow \mathcal{D} \cup \mathcal{D}_{\text{sda}}$
3: Initialize LoRA adapters Δ on attention layers
4: Optimize $\theta + \Delta$ on $\mathcal{D}'$ with cross-entropy
5: Save Δ for inference

The dataset is intentionally constructed to disentangle validity from plausibility, containing both believable yet invalid arguments and implausible yet valid ones; the metric explicitly penalizes bias through TCE.

3 System Overview

Our best system is a LoRA fine-tuned Qwen2.5-7B-Instruct model trained on a mix of original and symbolic training data. We target a low-parameter adaptation strategy to maximize reproducibility and efficiency, while explicitly reshaping the input distribution toward structural reasoning. The main challenges are (i) belief bias that favors plausibility over validity and (ii) limited training data (960 items). We address these by injecting symbolically augmented samples to decouple content from logic, and by using parameter-efficient adaptation to avoid overfitting.

3.1 Key Design Decisions and System Pipeline

We adopt QLoRA with attention-only adapters and greedy decoding to reduce memory while preserving instruction-following behavior. Our pipeline has three stages: (1) generate augmented training data with SDA; (2) fine-tune a LoRA adapter on the mixed corpus; and (3) run deterministic inference with the adapter and evaluate with the official script. This keeps training and inference aligned and enables clean ablations.

3.2 Algorithmic Specification

Let x be a syllogism and $y \in \{\text{Valid}, \text{Invalid}\}$ its label. SDA defines a mapping $\phi(\cdot)$ that replaces content words with symbols while preserving quantifiers and negation. The training set becomes $\mathcal{D}' = \mathcal{D} \cup \phi(\mathcal{D})$. We then optimize a LoRA-adapted model $\theta + \Delta$ by minimizing cross-entropy on $\mathcal{D}'$.

3.3 Symbolic Data Augmentation (SDA)

We replace concrete entities with randomized symbolic placeholders (e.g., *Wug*, *Zarp*, *A*, *B*) while preserving logical form. This removes plausibility cues and prevents lexical overlap from being grounded in world knowledge. For example:

> Original: "All dogs are mammals. No mammals are fish. Therefore, no dogs are fish."
> Augmented: "All Wugs are Zips. No Zips are Mors. Therefore, no Wugs are Mors."

SDA is implemented with template-driven replacement of subject, predicate, and middle terms while preserving quantifiers and negation, and the augmented samples are mixed with the original training data.

3.4 Concrete Example and Prompting Format

Given "All dogs are mammals. No mammals are fish. Therefore, no dogs are fish." (label *Valid*), SDA produces "All Wugs are Zips. No Zips are Mors. Therefore, no Wugs are Mors." Both forms are included in training; inference uses the original text and greedy decoding. We use a concise, instruction-style prompt that presents the syllogism and asks for a binary validity judgment. The output space is restricted to the two labels *Valid* and *Invalid*. We avoid chain-of-thought or intermediate rationales in the primary system because the LoRA adapter was trained to map directly from the full syllogism to the final label. This alignment between training and inference minimizes exposure bias and reduces variance in TCE.

3.5 Why SDA Helps

SDA reduces lexical priors and semantic anchoring by removing recognizable entities, forcing the model to rely on quantifiers and negation structure. In practice, we observed consistent TCE reductions when symbolic samples are included in training.

3.6 Model and Training

We fine-tune Qwen2.5-7B-Instruct using LoRA (QLoRA, 4-bit NF4). We adapt attention layers with a moderate rank (e.g., 16) and train on a mixture of original and symbolic samples. We use standard instruction-format prompts and train the model to output a single token (*Valid*/*Invalid*) without explicit reasoning. This "direct intuition" setup consistently yields the best balance between accuracy and bias. We observed significant sensitivity

to random seed: seed 42 produced the most favorable accuracy–bias trade-off, while other seeds increased TCE.

We also explored alternative adaptation targets (e.g., all-linear layers) and higher ranks, but these increased capacity without improving bias robustness. In practice, attention-only LoRA with a moderate rank provided the most stable performance across random seeds and avoided overfitting to surface patterns in the training set.

3.7 System Variants and Inference

We evaluated: (i) **Baseline LoRA** (no SDA), (ii) **LoRA + SDA (primary)**, (iii) prompt-based variants (few-shot, structured CoT), and (iv) post-hoc variants (contrastive decoding, confidence filtering, DPO). The primary submission is (ii); others are ablations/negative results. Inference is performed with greedy decoding to avoid sampling noise. We found that increasing temperature or sampling multiple reasoning paths (self-consistency) consistently degraded the content-bias metric. The final system is therefore a single deterministic pass, which improves stability and reduces variance across runs.

4 Experimental Setup

We use the official English training set (960 items) and the official test set (191 items). The organizers do not provide a separate dev set; we therefore evaluate only on the official test split and do not re-partition the data. All ablations use the same test set. The augmented dataset is generated with `data_augmentation.py` and combined with the original training data.

4.1 Preprocessing

We apply only the symbolic replacement pipeline described in Section 3 and keep tokenization unchanged. Quantifiers and negation markers are preserved verbatim. During training, we interleave original and augmented examples within each batch to avoid distribution shift between epochs.

4.2 Hyperparameters and Tuning

We use QLoRA with 4-bit NF4 quantization, attention-only LoRA adapters, and rank 16. We tune the symbolic–original mixing ratio to minimize TCE while retaining accuracy, fix random seed 42, and use a small learning rate with early stopping. Detailed hyperparameters and command lines are provided in the project archive.

For reproducibility, our training uses Qwen2.5-7B-Instruct with 4-bit NF4 quantization (double quantization enabled, BF16 compute) via `bitsandbytes`. We tokenize with the model's tokenizer and set `pad_token = eos_token`. We train for 3 epochs with learning rate 2×10^{-4}, per-device batch size 8, gradient accumulation 2 (effective batch size 16), max sequence length 512, and no packing. We use TRL `SFTTrainer` with BF16 and save one checkpoint per epoch. Our environment uses Python 3.12 on Ubuntu 22.04, PyTorch 2.8.0, and CUDA 12.8.

LoRA is applied to `q_proj`, `k_proj`, `v_proj`, `o_proj`, `gate_proj`, `up_proj`, and `down_proj` with rank 16, `lora_alpha` 32, and `lora_dropout` 0.05. Each example is formatted as a three-turn chat: a system prompt that emphasizes logical validity over factual truth, a user prompt of the form "Argument: ¡syllogism¿ Answer:", and an assistant label of *VALID* or *INVALID*.

Training was performed on a single NVIDIA GTX 4090 GPU.

4.3 External Tools, Libraries, and Evaluation Measures

Training and inference use `transformers`[1], `peft`[2], `trl`[3], and `torch`[4], with `bitsandbytes`[5] for quantization. Versions follow the accompanying `requirements.txt`. We report overall accuracy (ACC) and total content effect (TCE). The official ranking metric is $\mathrm{ACC}/(1 + \ln(1 + \mathrm{TCE}))$, which rewards high accuracy while penalizing bias. We compute these using the official evaluation script.

5 Results

5.1 Main Results

Our best standalone model (LoRA+SDA) achieves 96.34% accuracy and a TCE of 2.15, yielding a primary score of 44.86 under the official metric. These results represent the strongest trade-off between correctness and bias among methods that do not rely on external APIs. We did not record a shared-task leaderboard rank for this report. Compared with the unadapted Qwen2.5-7B-Instruct baseline (ACC 67.02, TCE 32.69, score 14.84), LoRA+SDA yields

[1] https://github.com/huggingface/transformers
[2] https://github.com/huggingface/peft
[3] https://github.com/huggingface/trl
[4] https://pytorch.org
[5] https://github.com/TimDettmers/bitsandbytes

Method	ACC	TCE	Score
Qwen2.5-7B-Instruct (base)	67.02	32.69	14.84
Baseline LoRA (no SDA)	95.81	3.12	39.64
LoRA + SDA (ours)	96.34	2.15	44.86
Gemini-3-Pro (API only)	99.48	1.06	57.68
Merged API + SDA (analysis)	100.00	0.00	100.00
Few-shot prompting	96.86	3.12	42.02
Self-consistency (temp > 0)	95.3	3.5	39.1
Contrastive decoding (alpha=1.0)	89.0	16.6	23.0
Structured CoT	60.0	4.62	22.0
DPO on same data	95.81	2.15	44.62

Table 1: Key ablations and results on the official test split (191 items). API and merged scores are reported for analysis and are not the primary submission. Numbers summarize representative runs from our logs.

a large gain in accuracy while sharply reducing content effects, indicating that the improvement is not merely a calibration shift but a substantive reduction in bias-driven errors.

We also evaluated a Gemini-3-Pro API baseline on the test set, which achieved 99.48% accuracy, TCE 1.06, and score 57.68. This provides a strong generalist reference point but still makes one error on the 191-item test set. The comparison highlights a complementary pattern: the API model is generally stronger but still fails on a small number of cases where the SDA specialist is correct.

5.2 Ablation Analysis

Effect of symbolic augmentation. Removing SDA (Baseline LoRA) reduces the score to 39.64 with higher TCE (3.12). This confirms that symbolic augmentation is the primary driver of bias reduction. The gains cannot be attributed to LoRA alone, since LoRA without SDA improves accuracy but leaves content effects relatively high.

Implicit vs. explicit reasoning. We find that explicit chains of thought (Mapping→Structure→Validity) degrade performance. On 7B models, long structured outputs add errors that overwhelm the final judgment. This points to a capacity mismatch between reasoning complexity and model size for this task, and to a training–inference mismatch when the model is trained on direct labels.

Self-consistency and sampling. Majority voting across sampled paths (temperature > 0) reduced accuracy and increased TCE. The model is already confident; sampling introduces spurious "Invalid" paths, harming both correctness and bias metrics. This suggests that stochastic decoding amplifies the base model's plausibility priors rather than revealing hidden correct paths.

Contrastive decoding. Subtracting base logits was intended to remove common-sense bias, but it also removed genuine logical competence present in the base model. This consistently reduced accuracy and worsened the score, indicating that the base model contributes useful reasoning signals alongside its biases.

DPO on same data. Training a DPO adapter on the same SFT data produced no gain: decision boundaries were already saturated. The resulting model retained TCE but lost small amounts of accuracy, consistent with mild overfitting and limited new supervision signal.

Specialist–generalist complementarity. The SDA specialist and the API model disagree on a small subset of cases (8/191). In those conflicts, the SDA model corrects the single API error, and a deterministic merge yields 100% accuracy and TCE 0. We report this as diagnostic analysis, illustrating how a targeted specialist can complement a strong generalist.

Overall, these results suggest that direct LoRA training with SDA is the most robust and efficient strategy for this dataset, and that methods introducing longer intermediate reasoning or post-hoc logit manipulation are counterproductive at this model scale.

5.3 Error Analysis

We manually inspected errors from the best LoRA+SDA model. Most failures fall into two categories: (i) quantifier-scope confusion in multi-negation cases (e.g., *"Not all A are B"* combined with *"No B are C"*), and (ii) mismatches between existential and universal statements where validity depends on subtle logical form. These cases are precisely where belief bias and shallow heuristics are most likely to interfere. Because the error count is

small, we do not report a confusion matrix; instead, we summarize error subtypes qualitatively.

6 Conclusion

We present a simple, robust system for Subtask 1: LoRA fine-tuning with symbolic data augmentation. The approach achieves high accuracy with low content bias, and ablations show that explicit reasoning, self-consistency, contrastive decoding, and DPO do not help at this model scale. SDA provides a practical path to decoupling content from validity; limitations include English-only coverage and potential lexical bias from augmentation. Our final submission uses the merged API + SDA strategy and attains a score of 100; among 45 teams, we are ranked 10th, with 11 teams tied at the top (all scoring 100). Looking ahead, we plan to extend SDA to multilingual settings and to probe failure cases involving quantifier scope and negation more systematically, with the goal of improving robustness without sacrificing efficiency. Looking ahead, we plan to extend SDA to multilingual settings and to probe failure cases involving quantifier scope and negation more systematically, with the goal of improving robustness without sacrificing efficiency.

Acknowledgments

We thank the SemEval-2026 Task 11 organizers for the task design and evaluation resources.

References

Luca Bertolazzi, Albert Gatt, and Raffaella Bernardi. 2024. A systematic analysis of large language models as soft reasoners: The case of syllogistic inferences. In *EMNLP 2024*.

Ishita Dasgupta, Ari K. Lampinen, Stephanie C. Y. Chan, and 1 others. 2022. Language models show human-like content effects on reasoning tasks. *arXiv*.

T. Eisape, Michael Tessler, Ishita Dasgupta, Fei Sha, S. Steenkiste, and Tal Linzen. 2024. A systematic comparison of syllogistic reasoning in humans and language models. In *NAACL 2024*.

Geonhee Kim, Marco Valentino, and Andre Freitas. 2025. Reasoning circuits in language models: A mechanistic interpretation of syllogistic inference. In *Findings of ACL 2025*.

Qing Lyu, Shreya Havaldar, Adam Stein, Li Zhang, Dheeraj Rao, Evangeline Wong, Marianna Apidianaki, and Chris Callison-Burch. 2023. Faithful chain-of-thought reasoning. In *AACL 2023*.

Kenta Ozeki, Ryo Ando, Takumi Morishita, Hiroshi Abe, Koji Mineshima, and Mitsuhiro Okada. 2024. Exploring reasoning biases in large language models through syllogism: Insights from the neubaroco dataset. In *Findings of ACL 2024*.

X. Quan, Marco Valentino, Louise Dennis, and Andre Freitas. 2024. Verification and refinement of natural language explanations through llm-symbolic theorem proving. In *EMNLP 2024*.

Leonardo Ranaldi, Marco Valentino, and Andre Freitas. 2025. Improving chain-of-thought reasoning via quasi-symbolic abstractions. In *ACL 2025*.

T. Seals and V. Shalin. 2024. Evaluating the deductive competence of large language models. In *NAACL 2024*.

Marco Valentino, Geonhee Kim, Dhaniya Dalal, Zhixue Zhao, and Andre Freitas. 2025. Mitigating content effects on reasoning in language models through fine-grained activation steering. *arXiv*.

Marco Valentino, Leonardo Ranaldi, Giulia Pucci, Federico Ranaldi, and Andre Freitas. 2026. Semeval-2026 task 11: Disentangling content and formal reasoning in language models. In *Proceedings of the 20th International Workshop on Semantic Evaluation (SemEval-2026)*.

Magdalena Wysocka, Daniel Carvalho, Oskar Wysocki, Marco Valentino, and Andre Freitas. 2025. Syllobionli: Evaluating large language models on biomedical syllogistic reasoning. In *NAACL 2025*.

Jie Xu, Hao Fei, Li Pan, Qian Liu, Min Lee, and William Hsu. 2024. Faithful logical reasoning via symbolic chain-of-thought. *arXiv*.