Team ewelinaksiez at SemEval-2026 Task 11: Reducing Content Bias in Syllogistic Reasoning via Semantic Abstraction

Ewelina Księżniak
Poznań University of Business and Economics
Poznań, Poland
ewelina.ksiezniak@ue.poznan.pl

Abstract

This paper presents our system for SemEval-2026 Task 11 Subtask 1 on content-independent syllogistic reasoning. The task evaluates whether language models can determine the formal validity of logical arguments independently of their semantic plausibility. To reduce content-driven biases, we propose a data augmentation strategy that progressively abstracts lexical semantics by replacing content words with symbolic placeholders and pseudo-words while preserving logical structure. Experiments based on fine-tuning *microsoft/deberta-large-mnli* show that abstraction-based augmentation reduces Content Effect and improves accuracy, leading to competitive performance on the official leaderboard. However, we observe substantial sensitivity to random initialization, suggesting that evaluation outcomes are partly influenced by stochastic factors. To better understand these effects, we conduct a layer-wise probing analysis using a Minimum Description Length framework, showing that the proposed approach decreases the accessibility of plausibility information in later transformer layers, indicating a shift toward more structure-oriented reasoning.

1 Introduction

This paper presents our system for Subtask 1 of SemEval-2026 Task 11, which investigates whether language models can perform logical reasoning independently of semantic content and real-world plausibility. The shared task focuses on syllogistic reasoning, where models must determine the formal validity of logical arguments while ignoring whether their conclusions are consistent with world knowledge. Our main strategy focused on reducing content-driven biases by systematically abstracting away lexical semantics during training. Building on the pretrained microsoft/deberta-large-mnli model (He et al., 2021), we iteratively expanded the training set with augmented examples in which the original content terms were replaced either by abstract symbols (e.g., T1, T2) or by randomly generated pseudo-words.

Our best submission achieved 12th place on the official leaderboard, although the top 11 systems obtained tied scores. Despite this competitive ranking, we observed substantial variability in performance across different random seeds during development. This instability suggested that the final leaderboard outcome might be influenced not only by the proposed augmentation strategy but also by stochastic training factors, potentially amplified by the sensitivity of the Total Score metric used for leaderboard ranking on a relatively small evaluation set.

These observations motivated an additional probing analysis aimed at understanding how plausibility-related information is represented inside the model. Specifically, we analyzed layer-wise representations using a Minimum Description Length (MDL) probing framework with a prequential coding scheme to measure how easily plausibility information could be recovered from different transformer layers. The analysis revealed that abstraction-based augmentation generally reduces the accessibility of plausibility cues in later layers.

2 Background

The task introduces a dataset of syllogisms constructed to disentangle logical structure from semantic plausibility. Each argument may be either *plausible* (aligned with world knowledge) or *implausible* (misaligned with real-world expectations), while independently being logically valid or invalid. The task in Subtask 1 is to predict the validity label regardless of its plausibility. System performance is evaluated using both accuracy and content bias-sensitive metrics designed to quantify reliance on semantic content (*Total Content Effect* - lower values indicate reduced dependence

Proceedings of the The 20th International Workshop on Semantic Evaluation (2026), pages 2149–2154

on semantic plausibility). The final ranking metric, referred to as the *Total Score*, is computed as the ratio between accuracy and Total Content Effect (Valentino et al., 2026). The organizers provided 960 training examples in English and a blind test set of 192 instances without labels available to participants.

3 System Overview

Our approach aims to reduce content-driven biases in syllogistic reasoning by progressively abstracting away lexical semantics during training. Starting from the pretrained `microsoft/deberta-large-mnli` (He et al., 2021) model, we iteratively expanded the training data with structurally equivalent but semantically altered variants of the original syllogisms. The central idea was to preserve logical form while weakening real-world semantic cues, thereby encouraging the model to rely on formal reasoning rather than plausibility or background knowledge.

The proposed training pipeline iteratively expanded the training data with progressively abstracted variants of each syllogism. Starting from the original examples, additional versions were introduced that preserved the underlying logical structure while systematically replacing content terms with **symbolic placeholders** or **nonsensical tokens (pseudo-words)**. For example, a syllogism such as *„All cars are a type of vehicle. No animal is a car. Therefore, no animal can be a vehicle."* was transformed into versions: *„All T1 are a type of T2. No T3 is a T1. Therefore, no T3 can be a T2."* (symbolic placeholders) or *„All wugdax are a type of zorptav. No mipzor is a wugdax. Therefore, no mipzor can be a zorptav."* (nonsensical tokens (pseud-words)). In addition to lexical abstraction, we augmented the dataset with **swapped** variants obtained by permuting the order of premises while keeping the conclusion unchanged (for example: *„No animal is a car. All cars are a type of vehicle. Therefore, no animal can be a vehicle."*). Swapped versions were generated for all data representations, including the original natural syllogisms, symbolically abstracted variants, and pseudo-word versions. To identify premises automatically, we excluded sentences beginning with explicit conclusion markers (e.g., *Conclusion*, *Therefore*, *Thus*, *Hence*, or *So*), detected using regular-expression matching. If no such marker was present, all sentences except the final one were treated as premises, while the last sentence was assumed to express the conclusion.

Symbolic placeholders generation To generate symbolic placeholders, we used the GPT API with the GPT-5.2 model (Singh et al., 2025). The transformation relied on a fixed prompt instructing the model to replace all content-bearing terms with abstract symbols while preserving the logical structure of the syllogism. Generation was performed with temperature set to 0. The prompt used in all experiments was as follows:

```
You are a "syllogism translator".
Your job is to remove all
content/plausibility by replacing
each distinct term-class (nouns like
"cars", "animals", "vehicles", etc.)
with neutral symbols T1, T2, T3, ...

Keep quantifiers and the logical
structure unchanged.

Rules: Use the same symbol for the same
term throughout the syllogism. Different
terms must get different symbols. Do not
use world knowledge. Keep sentence order
unchanged. Preserve conclusion markers
("Therefore", "So", etc.). Do not add
new content.

Output ONLY valid JSON with keys:
"symbolized" and "mapping".
```

Nonsensical tokens generation (pseudo-words) To generate versions with nonsensical tokens (pseudo-words), we replaced the symbolic terms (`T1`, `T2`, `T3`, `...`) with automatically generated pseudo-words. The procedure operated on the symbolized representations and consisted of the following steps:

- extract all abstract tokens (`T1`, `T2`, ...),
- generate a set of unique nonsense words using randomly sampled combinations of predefined syllables (`wug`, `dax`, `blick`, `zorp`, `tav`, `mip`, `nork`, `plin`, `rud`, `glarp`, `fep`, `zor`),
- apply a deterministic mapping replacing each symbolic token with a pseudo-word.

Data splitting. The dataset provided by the shared task organizers was divided using a stratified train–validation–test split with proportions $0.7/0.1/0.2$. All augmentations were applied exclusively to the training portion to avoid information leakage, while validation and test sets remained in their natural form.

Training configurations. Based on the augmentation pipeline, we constructed several progressively expanded training sets:

1. **Natural**: original syllogisms from the training data (691 training instances).

2. **Natural + Symbolic**: original examples augmented with symbolically abstracted variants (1382 training instances).

3. **Natural + Symbolic + Nonsensical tokens(pseudo-words)**: additional pseudo-word variants expanded across four abstraction levels (`pseudo`, `pseudo_1` –`pseudo_3`), resulting in 2,073, 2,764, 3,455, and 4,146 training instances, respectively.

4. **Swapped counterparts**: each of the above datasets further augmented with premise-swapped versions resulting in 2764, 4146, 5528, 6910, 8292 training instances, respectively.

All models were fine-tuned from the `microsoft/deberta-large-mnli` (He et al., 2021) checkpoint.

4 Experimental setup

Training configuration. All models were fine-tuned with HuggingFace `Trainer`. We trained for up to 10 epochs using AdamW with weight decay 0.01 and maximum gradient norm clipping of 1.0. We used a linear learning-rate schedule with no warmup. Training was performed in full precision (FP32; `fp16=False`, `bf16=False`) with per-device batch sizes of 8 for training and 16 for evaluation.

We experimented with learning rates $\{1 \cdot 10^{-5}, 2 \cdot 10^{-5}, 3 \cdot 10^{-5}\}$ and multiple random seeds.

Model selection, checkpointing, and stopping criteria. We evaluated, logged, and checkpointed once per epoch. The best checkpoint was selected by validation accuracy. To prevent overfitting, we applied early stopping with patience 2 epochs.

The final submission model was trained with learning rate $2 \cdot 10^{-5}$ and seed 100. A detailed comparison of performance results for specific hyperparameter configurations is reported in Section 5.

5 Results

Tables 1, 2, 3 summarize the impact of specific training configurations. First, Table 1 compares the Total Score obtained with three learning rates (1×10^{-5}, 2×10^{-5}, and 3×10^{-5}) using a fixed random seed (42) on the official test data. Across all learning rates, progressively richer data augmentation generally improves performance compared to the natural-only baseline, confirming the effectiveness of abstraction-based training. Averaged across setups, the learning rate 2×10^{-5} achieves the best overall performance, and was therefore selected for subsequent experiments.

Table 1: Total Score sensitivity to learning rate (seed=42).

System	2e-5	3e-5	1e-5
Natural (N)	28.51	29.54	25.61
N + Symbolic (NS)	32.73	40.24	28.51
NS + Pseudo (NSP)	54.28	39.59	45.07
NS + P_1 (NSPP1)	39.08	31.39	44.62
NS + P_1 + P_2	39.81	40.29	46.23
NS + P_1 + P_2 + P_3	46.05	46.05	40.54
NS (swap)	36.62	40.02	36.06
NSP (swap)	45.56	39.16	36.88
NSPP1 (swap)	39.86	39.59	39.25
NSPP1P2 (swap)	**55.82**	44.13	45.99
NSPP1P2P3 (swap)	33.88	33.69	33.55
Mean	**41.11**	**38.52**	**38.39**

Table 2 presents detailed results for all training variants using the selected learning rate 2×10^{-5} and seed 42, including accuracy and Content Effect (CE). The results demonstrate that applying the proposed data augmentation strategy leads to an overall reduction in Content Effect and an improvement in accuracy. The best trade-off between accuracy and bias reduction is achieved by the NSPP1P2 (swap) configuration, which attains the highest Total Score among systematically explored setups.

Finally, Table 3 analyzes sensitivity to random initialization for the selected configuration (NSPP1P2P3 swapped). Due to time constraints, we were able to evaluate multiple random seeds only for this single configuration rather than for all experimental variants. The results reveal substantial variance across seeds, with seed 100 yielding an exceptionally strong outcome characterized by both high accuracy and near-zero Content Effect. We also note that the Total Score, defined as a ratio between accuracy and Content Effect, may become unstable when CE approaches zero, potentially amplifying small differences in bias into large variations in the final score. Although other

Table 2: Results for seed=42 and learning rate 2×10^{-5} across progressively augmented training setups. Total denotes the official evaluation score (higher is better), while CE (Content Effect) measures plausibility bias (lower is better).

System	Total ↑	Acc ↑	CE ↓
Natural (N)	28.51	92.67	8.49
N + Symbolic (NS)	32.73	93.19	5.34
NS + Pseudo (NSP)	54.28	94.76	1.11
NS + P_1 (NSPP1)	39.08	95.29	3.21
NS + P_1 + P_2	39.81	96.86	3.19
NS + P_1 + P_2 + P_3	46.05	97.91	2.08
NS (swap)	36.62	97.38	4.26
NSP (swap)	45.56	96.86	2.08
NSPP1 (swap)	39.86	96.34	3.12
NSPP1P2 (swap)	**55.82**	96.86	1.09
NSPP1P2P3 (swap)	33.88	96.34	5.32

Table 3: Performance variability across random seeds for the final training configuration.

Seed	Total Score ↑	Accuracy ↑	CE ↓
42	33.88	96.34	5.32
100	**95.81**	97.91	**0.02**
1111	36.82	97.91	4.26
2222	39.42	95.29	3.12

training configurations achieved more stable average performance during development, this run provided a particularly favorable bias–accuracy tradeoff. Consequently, despite limited seed exploration, the model trained with learning rate 2×10^{-5} and seed 100 was selected as our final submission.

6 Probing analysis

Although the experimental results presented in Section 5 revealed a consistent improvement trend associated with progressively augmented training data, we also observed substantial sensitivity of the official evaluation metric to stochastic factors such as random seed selection. In particular, the final submission achieved an exceptionally high leaderboard score despite noticeably lower Total Scores obtained under alternative seeds for the same configuration (Table 3). This variability suggests that performance differences may not be fully explained by the implemented abstract data augmentation technique, but rather by random factors.

To better understand these effects, we conducted an additional probing analysis aimed at investigating how plausibility information is represented across transformer layers for different training configurations.

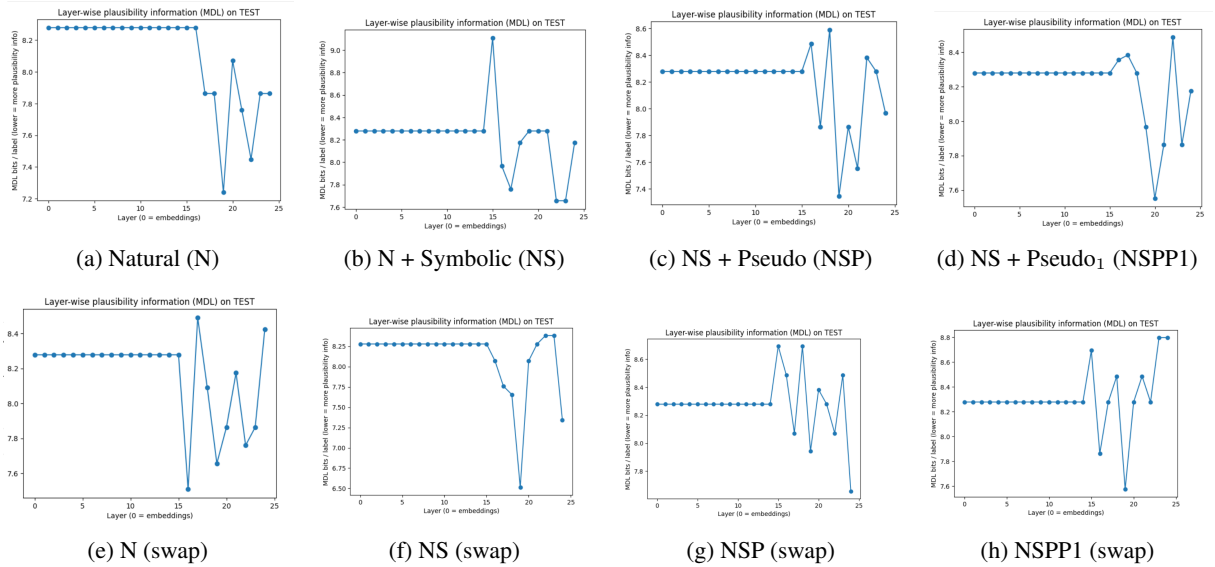
Probing setup. For selected trained models, we extracted hidden representations from all transformer layers. From each layer representation, we trained a lightweight probe to predict the plausibility label using a Minimum Description Length (MDL) framework based on a prequential coding scheme. In this scheme, labels are encoded sequentially while the probe is incrementally updated, and the cumulative coding cost reflects how easily the target information can be recovered from the representations (Voita and Titov, 2020). The resulting value (bits per label) reflects the amount of information about plausibility accessible from that layer; lower values indicate that plausibility can be predicted more easily and is therefore more strongly encoded. These probing experiments were conducted on an internal test set obtained from the train–test split, containing only syllogisms in their original natural form.

Interpreting the probing curves. Figure 1 presents layer-wise MDL values for training configuration lr: 2e-5, seed 100.

- Across all configurations, plausibility information remains at a relatively similar level up to approximately layer 15 (around 8.3 bits). This may indicate that earlier transformer layers have not yet specialized in representing semantic plausibility and instead capture more general linguistic features.

- For most premise-swapped variants, plausibility information becomes less accessible in later layers (i.e., higher MDL values requiring more bits for encoding). Notably, a clear shift is already visible in the swapped baseline configuration, suggesting that structural perturbations influence how plausibility cues are processed in higher layers.

- The highest number of bits required to encode plausibility information - meaning plausibility is hardest to recover - is observed for the *N + Symbolic (NS)* configuration (exceeding 9 bits around layer 15) and for the *NSPP1 (swap)* variant (approximately 8.8 bits in the final layers). At the same time, all configurations exhibit noticeable fluctuations across the later layers.



(a) Natural (N) (b) N + Symbolic (NS) (c) NS + Pseudo (NSP) (d) NS + Pseudo$_1$ (NSPP1)

(e) N (swap) (f) NS (swap) (g) NSP (swap) (h) NSPP1 (swap)

Figure 1: Layer-wise MDL probing on the test set measuring plausibility information contained in layer representations (lower MDL indicates more accessible plausibility information).

- Except for the baseline and the *N + Symbolic (NS)* setup, most augmented variants require more bits in the final layers compared to the earlier layers, indicating reduced accessibility of plausibility information at later processing stages. In contrast, a decrease in MDL values in the late layers is primarily observed in the baseline model, which may suggest a stronger reliance on semantic plausibility rather than logical structure. This observation provides evidence that the proposed augmentation method encourages the model to rely more on logical reasoning rather than semantic cues.

7 Conclusion

We presented a system for SemEval-2026 Task 11 Subtask 1 that aims to improve content-independent logical reasoning by progressively abstracting lexical semantics during training. Our results show that abstraction-based data augmentation reduces content effects while maintaining or improving accuracy, leading to competitive leaderboard performance. However, we also observed substantial sensitivity to random initialization, suggesting that evaluation outcomes are influenced not only by modeling choices but also by stochastic training dynamics and metric sensitivity. Additional MDL-based probing analysis provided evidence that the proposed approach reduces the accessibility of plausibility information in later transformer layers, indicating a shift toward more structure-based reasoning.

Limitations

This work has a few limitations. First, experiments were conducted only on a relatively small test set, which may amplify variability of the Total Score and limit the stability of conclusions. Second, due to computational constraints, extensive seed exploration was performed only for the selected configuration, preventing a systematic analysis across all variants.

References

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2021. Deberta: Decoding-enhanced bert with disentangled attention. In *International Conference on Learning Representations*.

Aaditya Singh, Adam Fry, Adam Perelman, Adam Tart, Adi Ganesh, Ahmed El-Kishky, Aidan McLaughlin, Aiden Low, AJ Ostrow, Akhila Ananthram, and 1 others. 2025. Openai gpt-5 system card. *arXiv preprint arXiv:2601.03267*.

Marco Valentino, Leonardo Ranaldi, Giulia Pucci, Federico Ranaldi, and André Freitas. 2026. Semeval-2026 task 11: Disentangling content and formal reasoning in large language models. In *Proceedings of*

the 20th International Workshop on Semantic Evaluation (SemEval-2026).

Elena Voita and Ivan Titov. 2020. Information-theoretic probing with minimum description length. *arXiv preprint arXiv:2003.12298*.