d'Olle Grieze at SemEval-2026 Task 11: Comparing the Impact of Supervised Fine-Tuning and Activation Steering on Mitigating Content Effect Bias in Syllogistic Reasoning

Twan Huiskens
t.n.huiskens@student.rug.nl

Tian Niezing
t.c.niezing@student.rug.nl

Koen Snelten
k.d.snelten@student.rug.nl

Abstract

We investigate the content effect bias in Large Language Models (LLMs) as part of SemEval 2026 Task 11. We compare the impact of supervised fine-tuning using low-rank adaptation against activation steering across several model families, including LLaMA, Gemma and Qwen. Our results show that SFT improves accuracy, with LLaMa 8B reaching 98.75% accuracy. Activation steering offers limited effectiveness in mitigating the content effect bias. A logit lens analysis further reveals that fine-tuning successfully shifts the model's focus toward logical structure, specifically within the later layers.

1 Introduction

The development of Large Language Models (LLMs) has led to significant progress in natural language understanding, yet their ability to perform pure logical reasoning remains a subject of debate. A primary challenge in this domain is the "content effect" bias, where models struggle to separate the formal validity of an argument from the real-world plausibility of its conclusion. For example, the syllogism *"No human is a tree; all trees are women; therefore, some women are not human"* is logically valid despite being factually implausible.

The SemEval 2026 Task 11 (Valentino et al., 2026) on separating content and formal reasoning in LLMs, addresses this fundamental issue by requiring systems to assess the logical validity of syllogisms independently of their factual truth. This task is critical for evaluating whether LLMs can function as reliable reasoners in domains where world knowledge might conflict with logical structure. This paper focuses on Subtask 1, which provides a binary classification challenge to determine the formal validity of syllogisms in English.

Our approach combines Supervised Fine-Tuning (SFT) and Activation Steering. To adapt pretrained instruction-tuned models like LLaMA (Touvron et al., 2023), Gemma (Team et al., 2025), and Qwen (Yang et al., 2025), we use Low-Rank Adaptation (LoRA) (Hu et al., 2021).

Beyond traditional fine-tuning, we explore activation steering to intervene in the model's internal processing at inference time. Our goal is to determine if these targeted internal adjustments are more effective at mitigating reasoning errors than standard fine-tuning alone.

Participating in this shared task revealed that while fine-tuning models show promising performance, they remain highly sensitive to content-effect bias. In our baseline experiments, Gemma 3 27B achieved the highest accuracy of 86.67% using a small system prompt. However, official evaluation results showed a significant performance gap when tested on the full test set. Our final submission (participant name "koensnelten") with a fine-tuned model of LLaMa 3 8B and a small system prompt, achieved a combined score of 31.28 with an accuracy of 93.72 % and a content effect bias of 6.36, ranking 34th overall (out of the 45 submissions). Compared to the other submissions, we have a relatively high content effect bias which suggests that our model struggle a lot ignoring highly plausible but logically invalid conclusions. This shows the difficulty of separating a model's world knowledge from formal logic.

2 Background

2.1 Data Description

The primary dataset for this study is provided by the SemEval 2026 Task 11 organizers (Valentino et al., 2026), specifically focusing on Subtask 1. This dataset consists of 960 syllogistic arguments, each labeled for both logical validity and content plausibility. The distribution between valid and invalid arguments, as well as plausible and implausible conclusions, is balanced to ensure that a model cannot rely on label frequency to achieve high per-

Proceedings of the The 20th International Workshop on Semantic Evaluation (2026), pages 1268–1281

formance.

In addition to the official task data, we used an additional dataset from Bertolazzi et al. (2024) containing 1,280 syllogisms. This dataset is also divided into plausible and implausible instances.

Because we want to create a dataset with equal valid and invalid syllogisms, we have preprocessed this dataset. Hereby we randomly chose 320 believable and 320 unbelievable syllogisms and picked one of the options that makes the syllogism invalid.

By combining these two datasets we got 2240 syllogisms. The test set of 240 syllogisms ($\approx 11\%$) is from the original SemEval-Task dataset and is shared between all students, so the results can be compared. The rest of the 2000 syllogisms are split into 1600 syllogisms ($\approx 71\%$) for the training set and 400 syllogisms ($\approx 18\%$) for the development set.

Each data instance includes a unique identifier, a natural language syllogism, the plausibility, and the target validity label. An example from the training set is provided below:


```
{
  "id":
    "50146f21-d265-4e3a-8d93-8165cdbe89a3",
  "syllogism":
    "All cars are a type of vehicle.
    No animal is a car. Therefore,
    no animal can be a vehicle.",
  "validity": false,
  "plausibility": true
}
```


2.2 Related Work

Recent research indicates that while LLMs show promise in reasoning tasks, they are highly sensitive to "content effect" or belief bias (Bertolazzi et al., 2024).

Comparative studies between humans and LLMs suggest that models often mirror human cognitive biases, struggling significantly when a logically valid argument leads to an unbelievable conclusion (Eisape et al., 2023). To mitigate these biases, recent advancements have moved beyond simple prompting. For instance, the k-CAST (kNN-Based Conditional Activation Steering) method demonstrates that intervening in a model's internal activations during inference time can improve formal reasoning accuracy by up to 15% little to no negative impact on the model's multilingual abilities (Valentino et al., 2025).

Our contribution builds upon these findings by specifically applying activation steering to the problem of content effect bias and compare its results with Supervised Fine-Tuning. While previous work has explored general reasoning improvements, our approach is novel in its focus on separating the internal representations of "validity" during the reasoning process, using targeted steering to force the model into a more abstract and structural processing.

3 System Overview

In this section, we describe our experimental methodology. We first give a general overview of the models and baselines used, then explain our fine-tuning and activation steering setups, and conclude with the logit lens approach, which is used to better understand content effect bias.

3.1 General Overview

Baseline To establish a comprehensive baseline, we evaluated various model architectures and sizes, alongside different system prompts and preprocessing techniques. Temperature was consistently set to 0 across all experiments in order to maximize instruction adherence. The most effective configurations for each model and size variant are summarized in Table A1. We used the baselines to find and select the prompts, which later were also used for the other SFT and steering setups (for details on the prompting strategy see: Appendix A).

Models In this research we use multiple instruction-tuned language models with different parameter sizes, from smaller models with 1-8B parameters to models with a size of 27-30B parameters. We opted to not choose the largest sizes of these models because research by Eisape et al. (2023) shows that the larger models do not always outperform the smaller models. Additionally, we selected the 1-8B models for this research due to their substantially lower computational cost, making them more feasible within our available resources. We focus on the models LLaMA, Gemma and Qwen, because these models are open-source, and that helps with the reproducibility of our research.

In addition to open-source architectures, we evaluated Gemini 3 Flash and Gemini 3 Pro only for the baselines. To ensure a direct comparison with our Supervised Fine-Tuning results, these models were tested using a zero-shot approach with a small system prompt and split-premise preprocessing.

3.2 Supervised Fine-tuning

To adapt the pretrained LLMs to syllogistic reasoning, we use a supervised fine-tuning strategy using the English training data. Since fully fine-tuning large models is computationally very expensive, our approach uses LoRa (Hu et al., 2021). We use a single prompt configuration consisting of a short instruction, premises presented on separate lines, and a requirement that the model output be formatted as JSON.

In our first experiment, we performed SFT with LoRA on all small models ranging from 1B to 8B parameters, without any hyperparameter optimization. The hyperparameters are shown in the appendix D in table A5

For our second experiment, we control both LoRA-specific and general training hyperparameters using a grid search. The LoRA parameters include the rank (r), which determines the dimensionality of the update matrices, and the scaling factor alpha (α), which regulates the contribution of the LoRA modules to the model output. In addition, we tune standard training parameters such as the learning rate, batch size, and number of training epochs. The full set of values explored during the grid search is summarized in A3.

For our last experiment, we focused on the plausible and implausible labels. Specifically, we trained separate versions of each LLM using only plausible syllogisms and only implausible syllogisms. Using this setup, we also investigate content bias, as training on only one plausibility type may cause the model to rely more on the believability of the content rather than on logical validity. This allows us to compare these specialized models with those trained on the full dataset and assess how plausibility influences reasoning behavior and performance.

3.3 Activation Steering

In addition to SFT, we will use activation steering to control model behavior at inference time without altering the weights. This is achieved by adding a precomputed steering vector, $\vec{v}$, to the model's activations. The vector is calculated by subtracting the mean activations of the negative prompt set from those of the positive set ($\vec{v} = \bar{A}_{\text{pos}} - \bar{A}_{\text{neg}}$). This vector, scaled by a hyperparameter α, is then used to steer the model's forward pass toward or away from the target behavior (Turner et al., 2024). We will apply these steering techniques to all of the models we also test SFT on to create a fair comparison between the two methods. We will use several steering approaches with different steering vectors to test different aspects of the syllogistic reasoning process in LLMs. We will evaluate the approaches on the same metrics used for SFT.

3.3.1 Approach 1: Isolating Logical Validity

Our first approach creates a steering vector targeting logical validity while controlling for plausibility. The mean positive activations ($\bar{A}_{\text{pos}}$) are calculated from valid syllogisms with balanced plausibility (50% plausible, 50% implausible). The mean negative activations ($\bar{A}_{\text{neg}}$) are calculated from invalid syllogisms with the same balanced distribution. The resulting steering vector $\vec{v} = \bar{A}_{\text{pos}} - \bar{A}_{\text{neg}}$ is hypothesized to represent validity in the model's activation space, with plausibility effects canceling out.

3.3.2 Approach 2: Eliminating Content Effect Bias

Our second approach directly targets content effect bias by contrasting the two conflicting conditions. The mean positive activations ($\bar{A}_{\text{pos}}$) come from implausible but valid syllogisms, where the model must rely purely on logical structure. The mean negative activations ($\bar{A}_{\text{neg}}$) come from plausible but invalid syllogisms, where the model is most susceptible to content bias. The resulting vector $\vec{v} = \bar{A}_{\text{pos}} - \bar{A}_{\text{neg}}$ is hypothesized to represent the distinction between logic-based and bias-driven reasoning. This approach assumes baseline susceptibility to content effect bias; if the model already performs correctly on these cases, the steering vector will be ineffective.

3.3.3 Hyperparameter Selection

Hyperparameter selection follows a two-stage process using the development set. First, we conduct a grid search over $n_{\text{icl}} \in \{2, 4, 8\}$ and $n_{\text{prompts}} \in \{16, 32, 64\}$ while holding the steering coefficient fixed at $\alpha = 2.0$. In the second stage, we take the best-performing configuration and sweep α from 1.0 to 10.0 in 0.5 increments to optimize steering strength. The final optimal parameters are then evaluated on the test set. This two-stage search procedure will allow us to separately optimize the vector quality (via n_{icl} and n_{prompts}) and the application strength (via α). For more details refer to appendix C.

4 Logit Lens Analysis

The logit lens technique (Nostalgebraist, 2020) provides insight into a model's intermediate computations by projecting hidden states at each layer. For a given layer, we compute a probability distribution over the vocabulary by applying the final layer normalization and the language model head. We then apply softmax to determine the probabilities assigned to the tokens "valid" and "invalid." By tracking their probability trajectories, we can observe how the model's prediction evolves during the forward pass.

This technique is particularly suited for studying content effect bias. In aligned cases (VP, II) logical structure and plausibility point toward the same answer, while in conflicting cases (VI, IP) we hypothesize an preference for plausibility before logical reasoning emerges in later layers. We apply logit lens analysis to the baseline models and our best fine-tuned model to reveal whether fine-tuning alters internal reasoning dynamics, and use the results to identify intervention layers for activation steering.

5 Results

Table 1 presents a comparison of all intervention methods across model families.

Model	Baseline	SFT	AS-1	AS-2
Gemma-3 1B	57.92	**76.25**	56.67	55.00
Gemma-3 4B	75.42	**92.08**	75.83	74.58
Gemma-3 27B	84.17	–	–	–
LLaMA-3.2 1B	52.50	**66.67**	52.92	47.50
LLaMA-3.2 3B	68.75	**86.25**	63.33	61.25
LLaMA-3.1 8B	65.00	**93.75**	65.00	65.17
Qwen3-4B	66.25	**94.58**	66.25	67.92
Qwen3-30B	71.67	–	–	–

Table 1: Test accuracy (%) across all intervention methods. Baseline: zero-shot performance; SFT: supervised fine-tuning; AS-1: activation steering approach 1; AS-2: activation steering approach 2.

5.1 SFT Result

Table 2 presents the evaluation results of the different models on the test set provided by the organizers. Among the models, Qwen3-4B achieved the highest accuracy at 95.28%, while LLaMA-3.1 8B reached 93.71% accuracy with the lowest content effect of 6.3608. We uploaded the LLaMA-3.1 8B model to the final CodaBench evaluation, which secured 34th place in Subtask 1.

Model	Acc ↑	CE ↓	Comb ↑
Gemma-3 4B	91.62	13.8298	24.7855
LLaMA-3.1 8B	93.71	6.3608	31.279
Qwen3-4B	95.28	7.4468	30.4066

Table 2: Model Evaluation Results on the test data given by the organizers. Acc = accuracy (%), CE = content effect bias score, Comb = combined metric.

For the second experiment applied SFT with LoRa to all pretrained models and optimized the hyperparameters through a grid search. This procedure substantially improved performance on the syllogistic reasoning task compared to the baseline models, yielding consistently high accuracy across architectures. The results also reveal that performance is sensitive to the choice of LoRA rank and scaling parameters, confirming that careful tuning is necessary to obtain optimal behavior for reasoning tasks.

In appendix D table A2 shows that the highest accuracy was achieved by a fine-tuned Llama 3.1 8B with 98.75%, indicating that this model can learn the target reasoning patterns almost perfectly under the right hyperparameter settings. The lowest content effect, was achieved by a fine-tuned Qwen 3 4B with a score of 0.9968. When considering the combined score, which balances correctness with bias reduction, the same fine-tuned Qwen 3 4B achieved the best overall combined score of 57.6394. Together, these findings suggest that SFT can enhance different aspects of model behavior, with Llama 3.1 8B excelling in accuracy and Qwen 3 4B showing the strongest overall balance between performance and bias reduction.

Determining the importance of training data In appendix D table A4 shows that the plausibility of the training data strongly influences both model accuracy and content effect. Across all models, training exclusively on implausible syllogisms leads to higher accuracy and smaller content effect than training on plausible ones. Models trained on plausible data show substantially lower accuracy and larger content effects, indicating greater susceptibility to belief bias, whereas training on implausible data reduces this bias. This pattern suggests that plausible examples encourage reliance on real world knowledge, while implausible examples push models to focus more on formal logical validity. Overall, these results indicate that training data plausibility plays a crucial role in shaping

reasoning strategies, with implausible only training producing models that reason more accurately and with less interference from prior world knowledge.

5.2 Activation Steering Results

We evaluated both steering approaches across six models. We had to leave out the larger models (Qwen 30B and Gemma-3 27B) due to computational constraints. The full steering configurations with the optimal hyperparameters can be found in Appendix F (Table A7). Table A6 in Appendix E reports the performance of each model and approach on the test set.

Limited Effectiveness. The activation steering experiments did not yield much improvement over the baseline and generally fall short compared to the SFT approach. As shown in Table A6, Gemma 4B, the best-performing steered model, achieved an accuracy of 75.83% using approach 1. It also got the highest combined score of 18.26 using approach 2. However, this is only a negligible improvement over its baseline accuracy of 75.42%. Similarly, LLaMA 8B achieved 65.00% accuracy with approach 1, which is identical to its baseline. The performance of several other models degraded compared to their baselines. Approach 2 for Qwen 3 4B did improve slightly on its baseline with an accuracy of 67.92% (an increase of 1.67 percentage points) however, approach 1 resulted in exactly the same accuracy as the baseline. This strongly indicates that our activation steering approach is not able to mitigate the content effect bias.

Approach comparison. Both approaches yielded very similar results, which suggests that both methods impact the models reasoning process in comparable ways. While Valentino et al. (2025) demonstrated that activation steering can mitigate content effect bias, our steering approaches may fail to cleanly isolate the validity or content effect bias resulting in these limited results.

Error analysis. The per-quadrant accuracies (Table A8 in Appendix G) show that steering does not improve logical reasoning, but instead we see different types of failures. First, we observe a bias shift where models collapse towards a single class. Gemma 1B achieves near-perfect accuracy on valid quadrants (VP: 100.0% VI: 98.5%) but collapses on invalid quadrants (IP: 9.5%, II: 19.3%), indicating the model defaults to "valid" regardless of logical structure. Llama 8B exhibits a milder version of this pattern. Interestingly, Qwen 4B shows the opposite pattern, with strong invalid quadrant performance (IP: 84.1% II: 94.7%) but poor valid quadrant accuracy (VP: 59.3%, VI: 30.3%). Llama 3B displays a similar shift, with VI dropping to 31.8% under approach 1 while II rises to 82.5%. This suggests the steering vector steered towards a certain class instead of enhancing reasoning. Across all models the content effect bias remains persistent. The performance on congruent quadrants (VP, II) is consistently better than on incongruent quadrants (VI, IP). This confirms that the steering interventions failed to decouple logical processing from semantic plausibility.

5.3 Logit Lens

To gain insight into how fine-tuning affects internal model dynamics, we applied the logit lens technique to all baseline models and the fine-tuned Qwen 4B model. Across baseline models, intermediate predictions group by plausibility rather than logical validity in the later layers, directly reflecting the content effect bias. The one exception was Gemma 3 27B, which was the only baseline model to achieve validity-aligned grouping. Since Qwen 3 30B, a comparably sized model, still exhibits plausibility-driven trajectories, this appears to reflect model-family-specific properties rather than scale alone. After fine-tuning, this pattern reverses: the two valid quadrants track together toward the correct label regardless of plausibility, as do the two invalid quadrants, indicating a shift from content-driven to logic-driven reasoning. Full results and visualizations are provided in Appendix H.

6 Conclusion

Our investigation into mitigating content effect bias in syllogistic reasoning demonstrates that while LLMs are highly susceptible to plausibility, targeted interventions can significantly improve logical reasoning. Our results indicate a clear performance gap between methods: SFT using LoRA proved highly effective, with LLaMA-3.1 8B achieving up to 98.75% accuracy and Qwen 3 4B providing a robust balance between accuracy and bias reduction. In contrast, activation steering offered limited success. Future work could not only focus in improving the accuracy but also aim to learn more about the underlying logical circuits of these LLMs.

References

Leonardo Bertolazzi, Albert Gatt, and Raffaella Bernardi. 2024. A systematic analysis of large language models as soft reasoners: The case of syllogistic inferences. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 13882–13905, Miami, Florida, USA. Association for Computational Linguistics.

Tiwalayo Eisape, Michael Henry Tessler, Ishita Dasgupta, Fei Sha, Sjoerd van Steenkiste, and Tal Linzen. 2023. A systematic comparison of syllogistic reasoning in humans and language models. *arXiv preprint arXiv:2311.00445*.

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. *Preprint*, arXiv:2106.09685.

Nostalgebraist. 2020. interpreting GPT: the logit lens — LessWrong.

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, and Léonard Hussenot. 2025. Gemma 3 technical report.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurélien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. Llama: Open and efficient foundation language models. *CoRR*, abs/2302.13971.

Alexander Matt Turner, Lisa Thiergart, Gavin Leech, David Udell, Juan J. Vazquez, Ulisse Mini, and Monte MacDiarmid. 2024. Steering Language Models With Activation Engineering. *arXiv preprint*. ArXiv:2308.10248 [cs].

Marco Valentino, Geonhee Kim, Dhairya Dalal, Zhixue Zhao, and André Freitas. 2025. Mitigating content effects on reasoning in language models through fine-grained activation steering. *arXiv preprint arXiv:2505.12189*.

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)*.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, and Zihan Qiu. 2025. Qwen3 technical report.

A System Prompts

The performance of an LLM is very dependent on the input prompt. To analyze the impact of the prompt on the content effect, we have experimented with different prompts:

- A short prompt (zero shot) (Appendix A.1)
- A long prompt (zero shot)
- A prompt in which we prompted the model to output only the tag "valid" or "invalid", without punctuation and JSON (zero shot)
- A short prompt (fewshot) (Appendix A.2)
- A long prompt (fewshot)
- A prompt in which we prompted the model to output only the tag "valid" or "invalid", without punctuation and JSON (fewshot)

We included few-shot prompts in our baseline experiments to ensure a fair comparison with activation steering, which requires examples to define its target direction. Additionally, we tested all prompt variations (short and long; zero-shot and few-shot) using a specific formatting step: separating premises with line breaks. We introduced this structure to determine if clearly separating the premises helps the model better analyze the argument's logical structure.

To ensure a fair comparison, we formatted all inputs using the native chat template provided for each instruction-tuned model. While the structural tags (e.g., [INST], <|user|>) will differ, the core prompt content remains consistent across all models. With these prompts, we also hope to find if there is a correlation between the length of a prompt and the content effect bias. We argue that a larger prompt could potentially force the model to use more of its world knowledge since more knowledge is required to understand the prompt.

A.1 Small System Prompt

System Prompt (Zero-Shot)

```
You are a formal logic processor. Your task is to determine the logical validity of a given
syllogism. CRITICAL: Ignore factual accuracy. Validity requires the conclusion to be an
inescapable consequence of the premises. Your response MUST be a single, raw JSON object and
nothing else. Do not use markdown or explanations.
The JSON must be: {"validity": <boolean>}
```

A.2 Small System Few-Shot Prompt

System Prompt (Few-Shot)

```
You are a formal logic processor. Your task is to determine the logical validity of a given
syllogism. CRITICAL: Ignore factual accuracy. Validity requires the conclusion to be an
inescapable consequence of the premises. Your response MUST be a single, raw JSON object and
nothing else. Do not use markdown or explanations.
The JSON must be: {"validity": <boolean>}

Here are some examples:
Syllogism: All millionaires are poor. Some millionaires are happy. Therefore, some poor people
are happy.
Answer: {"validity": true}

Syllogism: All mammals are animals. Camels are mammals. Therefore, camels are animals.
Answer: {"validity": true}

Syllogism: No humans are immortal. Some organisms are immortal. Therefore, some organisms are
humans.
Answer: {"validity": false}

Syllogism: Some Yumboes are Tarasques. All Tarasques are Quinotaurs. Some Quinotaurs are
Centaurs. No Griffin is a Centaur. Therefore, no Griffin is a Yumboe.
Answer: {"validity": false}
```

B Baseline scores

Model Name	Prompt	Preprocess	Acc (%)
Qwen 3 30B	Small System	split	71.67
Qwen 3 4B	Small System	split	66.25
LLaMa 3.2 1B	Long System	split	54.58
LLaMa 3.2 3B	Small System	split	68.75
LLaMa 3.1 8B	Small System	split	65.00
Gemma 3 27B	**Small System**	**split**	**84.17**
Gemma 3 4B	Small System	split	75.42
Gemma 3 1B	Small System	none	56.25
Gemini 3 Pro	Small System	split	77.50
Qwen 3 30B	Small System (Few-shot)	split	77.50
Qwen 3 4B	Small System (Few-shot)	split	73.33
LLaMa 3.2 1B	Small System (Few-shot)	split	52.92
LLaMa 3.2 3B	Valid/Invalid Tag (Few-shot)	none	61.67
LLaMa 3.1 8B	Small System (Few-shot)	split	70.00
Gemma 3 27B	**Small System (Few-shot)**	**split**	**86.67**
Gemma 3 4B	Small System (Few-shot)	split	75.42
Gemma 3 1B	Small System (Few-shot)	none	57.92

Table A1: Comparison of model accuracy (%) under different system prompts and preprocessing.

C Steering hyperparameter selection procedure

We perform the hyperparameter selection in two stages to find the optimal configuration for each steering approach. In the first stage, we will perform a grid search over the number of in-context learning examples ($n_{\text{icl}} \in \{2, 4, 8\}$) and the number of contrastive prompt pairs ($n_{\text{prompts}} \in \{16, 32, 64\}$). The n_{icl} parameter will determine how many example syllogisms precede each target syllogism in the prompt, while n_{prompts} will specify the total number of contrastive pairs used to calculate the steering vector. For each configuration, we will compute steering vectors on the training set and evaluate performance on the development set using a fixed steering coefficient ($\alpha = 2.0$).

In the second stage, we take the best-performing configuration from stage one and conduct an α-search to optimize the steering coefficient. We will evaluate α values ranging from 1.0 to 10.0 in increments of 0.5 on the development set. The final optimal configuration (including n_{icl}, n_{prompts}, and α) will then be evaluated on the test set. This two-stage search procedure will allow us to separately optimize the vector quality (via n_{icl} and n_{prompts}) and the application strength (via α).

D Supervised fine tuning

Model	Acc. (%)	CE	Comb	α	r	LR	Epochs	Batch
LLama 3.1 8B	**98.75**	**1.5873**	**50.6250**	32	16	2.00×10^{-4}	5	1
LLama 3.1 8B	97.92	2.3810	44.1432	32	8	2.00×10^{-4}	5	1
LLama 3.1 8B	97.08	1.6234	49.4198	16	8	2.00×10^{-4}	3	1
Qwen 3 4B	**97.92**	1.7544	48.6375	32	32	2.00×10^{-4}	5	2
Qwen 3 4B	97.53	2.9828	40.9321	32	8	2.00×10^{-4}	5	1
Qwen 3 4B	97.53	**0.9968**	**57.6394**	32	16	2.00×10^{-4}	5	2
Gemma 3 4B	**97.08**	3.9683	37.2957	32	8	2.00×10^{-4}	3	1
Gemma 3 4B	96.67	**3.1746**	**39.7966**	32	16	2.00×10^{-4}	3	1
Gemma 3 4B	96.67	15.5556	33.5612	32	16	2.00×10^{-4}	5	1

Table A2: Model performance across various hyperparameter configurations our test set.

Parameter	Values
LoRA Alpha (α)	8, 16, 32
LoRA Rank (r)	8, 16, 32
Learning Rate	1×10^{-4}, 2×10^{-4}
Epochs	3, 5
Batch Size	1, 2

Table A3: Grid search hyperparameters used for supervised fine-tuning with LoRA.

Model	Plausibility	Acc (%)	CE	Comb
LLama 3.1 8B	Plausible	73.33	37.09	15.81
	Implausible	88.33	10.16	25.88
Qwen 3 4B	Plausible	84.58	18.36	21.34
	Implausible	92.08	7.97	28.83
Gemma 3 4B	Plausible	82.08	21.70	19.91
	Implausible	87.50	13.49	23.82

Table A4: Model performance metrics across Plausible and Implausible conditions, including accuracy, content effect, and combined scores.

Hyperparameter	Value
Max Sequence Length	512
LoRA Alpha (α)	16
LoRA Rank (r)	16
LoRA Dropout	0.05
Learning Rate	2.00×10^{-4}
Training Epochs	3
Batch Size	2
Temperature	0

Table A5: LoRA Fine-tuning hyperarameters

E Activation Steering Results per Model and Approach

	Approach 1			Approach 2		
Model	Acc ↑	CE ↓	Comb ↑	Acc ↑	CE ↓	Comb ↑
Gemma 1B	56.67	45.24	11.72	55.00	44.44	11.42
Gemma 4B	74.58	22.09	18.02	74.17	20.37	**18.26**
LlaMa 1B	52.92	8.20	16.44	47.50	16.40	12.32
LlaMa 3B	63.33	29.15	14.37	61.25	36.17	13.27
LlaMa 8B	65.00	35.56	14.13	64.17	33.73	14.11
Qwen 4B	66.25	32.22	14.71	67.92	32.22	15.08

Table A6: Activation steering test set results. Acc = accuracy (%), CE = content effect bias score, Comb = combined metric.

F Activation Steering Grid Search Results

Model	**App.**	n_{icl}	$n_{prompts}$	α	**Layer**	**Dev Acc**	**Test Acc**
Gemma-3 1B	1	2	16	7.0	22	48.50%	56.67%
Gemma-31B	2	4	16	7.0	22	49.50%	55.00%
Gemma-3 4B	1	4	16	6.0	22	68.75%	74.58%
Gemma-3 4B	2	2	16	6.0	22	69.25%	74.17%
LLaMA-3.2-1B	1	8	64	3.0	21	49.00%	52.92%
LLaMA-3.2-1B	2	8	32	3.0	21	52.25%	47.50%
LLaMA-3.2-3B	1	4	64	2.0	21	61.75%	63.33%
LLaMA-3.2-3B	2	8	16	4.0	21	63.00%	61.25%
LLaMA-3.1-8B	1	2	16	2.0	21	60.75%	65.00%
LLaMA-3.1-8B	2	2	16	2.0	21	60.50%	64.17%
Qwen3-4B	1	2	32	6.0	15	67.50%	66.25%
Qwen3-4B	2	4	16	1.0	15	68.00%	67.92%

Table A7: Activation steering grid search results. Hyperparameters were selected via grid search on the development set, then optimal α values were determined through separate search. n_{icl} = number of in-context learning examples. $n_{prompts}$ =number of prompts for steering vector computation. α = steering coefficient. Layer = starting layer for steering intervention.

G Per-Quadrant Accuracy for Steering Approaches

	Approach 1				Approach 2			
Model	VP	VI	IP	II	VP	VI	IP	II
Gemma 1B-IT	100.0	98.5	9.5	19.3	100.0	93.9	11.1	15.8
Gemma 4B-IT	98.1	77.3	54.0	71.9	96.3	75.8	55.6	71.9
LlaMa 1B	59.3	57.6	42.9	52.6	63.0	60.6	30.2	36.8
LlaMa 3B	75.9	31.8	68.3	82.5	70.4	25.8	63.5	91.2
LlaMa 8B	96.3	60.6	36.5	71.9	96.3	57.6	39.7	68.4
Qwen 4B	59.3	30.3	84.1	94.7	63.0	30.3	87.3	94.7

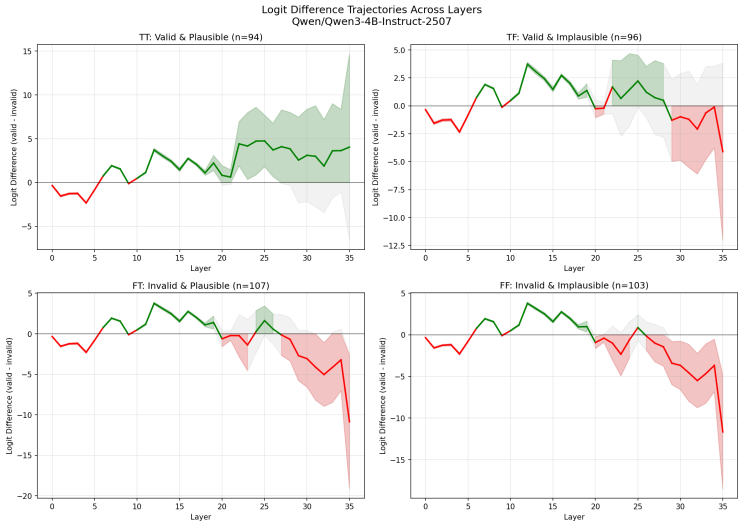
Table A8: Per-quadrant accuracy (%) for activation steering on the test set. VP = valid-plausible, VI = valid-implausible, IP = invalid-plausible, II = invalid-implausible.

H Logit Lens Analysis

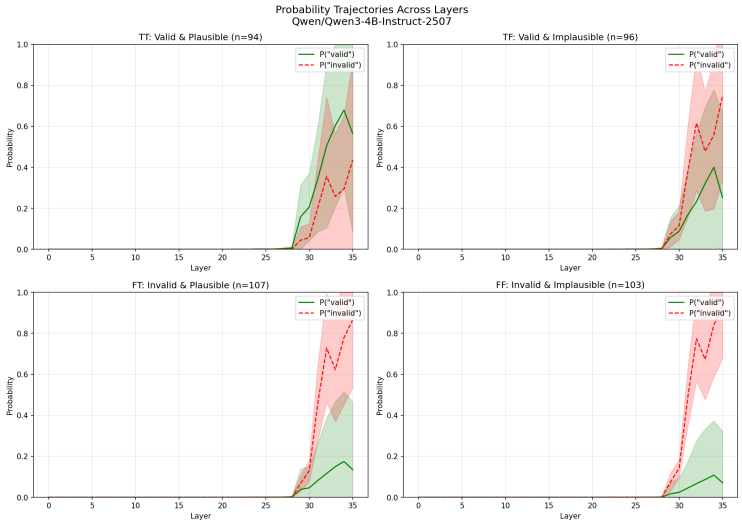
We applied the logit lens technique to all baseline models across the three architecture families. The smallest models (Gemma 3 1B, LlaMa 3.2 1B) showed no meaningful differentiation between quadrants. Among the remaining models, a consistent pattern was visible, in the later layers, trajectories grouped by plausibility rather than logical validity. The one exception was Gemma 3 27B, which was the only baseline model to achieve validity aligned grouping.

To illustrate how fine-tuning changes these internal dynamics, we compare the baseline Qwen 4B model and the fine-tuned Qwen 4B model (see Figure A1 and Figure A2). In the baseline model, the quadrant comparison (Figure A1 c) shows that the model's intermediate predictions are shaped by plausibility. Valid plausible syllogisms are handled correctly, but for VI syllogisms the logit difference trajectories follow almost the same pattern as for the invalid cases, resulting in wrong answers. This effect is primarily caused by the content effect bias.

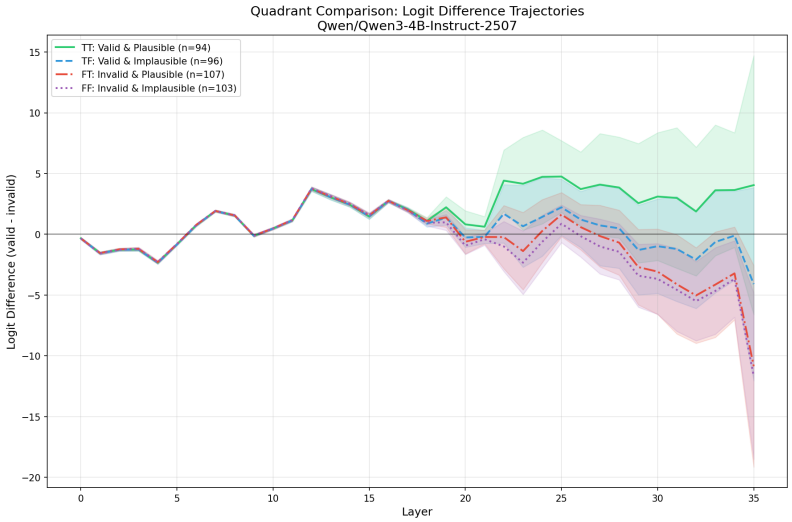
After fine-tuning, this pattern reverses (Figure A2 c). The two valid quadrants now track together towards the correct label regardless of plausibility, as do the two invalid quadrants. This indicates that the model has learned to base its prediction on logical structure rather than content. Notably, this change is confined to the later layers. Early layer processing remains largely unchanged between the two models, suggesting that the fine-tuning does not alter how the model encodes the input or understands the language (which are usually parts of the early layers), but rather how it uses that information to arrive at a logical conclusion. This also means that logical reasoning is more dominant in later layers.



(a) Logit difference ($logit_{\text{valid}} - logit_{\text{invalid}}$) across layers for each quadrant.

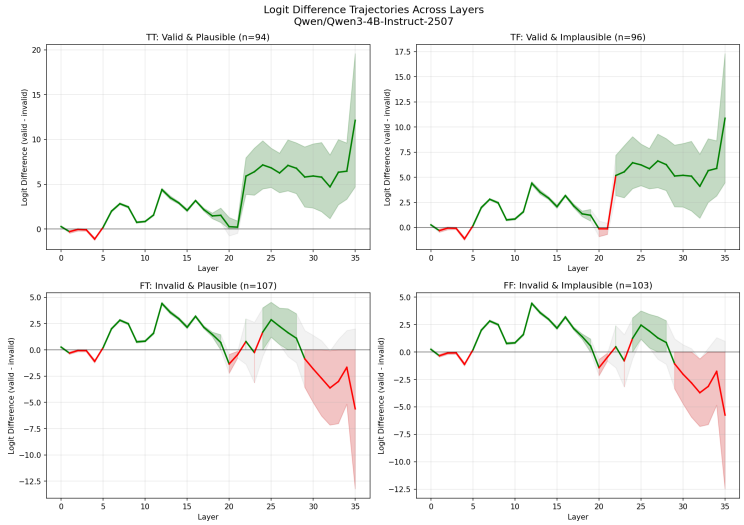


(b) Per-quadrant probability trajectories for P("Valid") and P("Invalid") across layers.

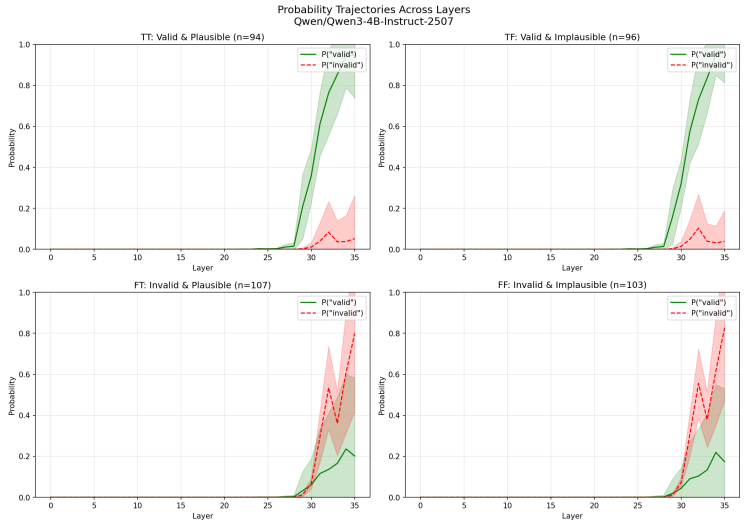


(c) Overlay of all four quadrants, showing trajectories grouped by plausibility rather than validity in later layers, consistent with content effect bias.

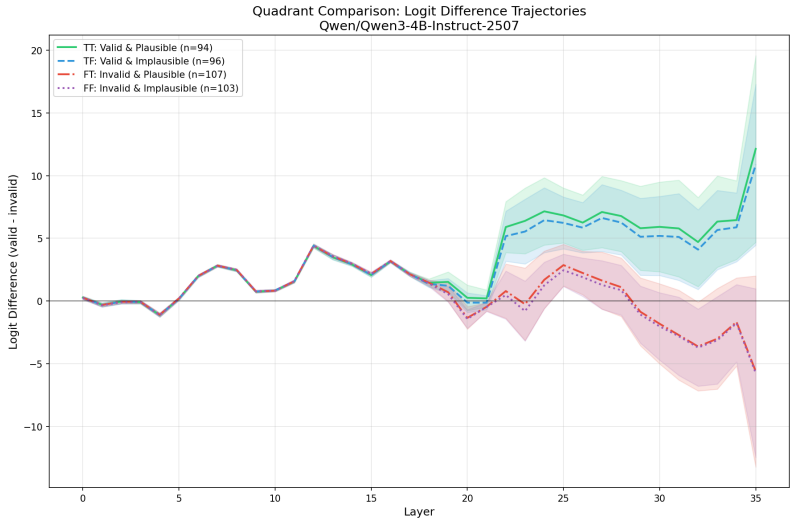
Figure A1: Logit lens analysis for Qwen/Qwen3-4B-Instruct (baseline).



(a) Logit difference ($logit_{\text{valid}} - logit_{\text{invalid}}$) across layers for each quadrant.



(b) Per-quadrant probability trajectories for P("Valid") and P("Invalid") across layers.



(c) Overlay of all four quadrants, showing trajectories grouped by plausibility rather than validity in later layers, consistent with content effect bias.

Figure A2: Logit lens analysis for the SFT model (Qwen/Qwen3-4B-Instruct).