Cost-Effective Improvement of Thai Legal QA using GRPO and Semantic Reward Proxies

Anonymous ACL submission

Abstract

Citation-sensitive legal question answering in low-resource settings, such as Thai law, poses unique challenges for large language models (LLMs). We investigate how to align large language models for citation-sensitive legal question answering in Thai using Group-Relative Policy Optimization (GRPO). Focusing on affordable alignment, we compare semantic similarity–based reward proxies against large LLM judge models. Experiments on the NitiBench benchmark show that semantic reward achieves competitive performance in in-domain settings, with up to 90% Citation F1 improvement over instruction tuning and 2.5× reduced compute cost compared to judge-based supervision. Ablation studies further reveal the importance of answer-level reward components, while correlation analysis supports the partial validity of semantic signals as reward proxies. These results offer actionable insights into affordable and robust alignment for legal LLMs.

1 Introduction

Recent advances in large language models (LLMs) have enabled new possibilities for legal question answering (QA) (Colombo et al., 2024; Lab, 2024; Corporation, 2025). However, delivering accurate and grounded responses remains challenging, especially in domains like Thai law, where legal complexity and limited training data lead to frequent hallucinations and citation errors (Akarajaradwong et al., 2025).

Retrieval-augmented generation (RAG) (Lewis et al., 2021) has been proposed to improve factuality, but existing systems often fail to cite relevant laws even when provided with the right context. While instruction tuning improves general fluency, it offers limited control over citation behavior. This motivates the need for more targeted alignment techniques that not only improve factuality but also enforce verifiable citation standards.

To address these challenges, we start with an observation in which there exists a gap between retrieval performance and LLM citation performance in RAG (Akarajaradwong et al., 2025). This discrepancy highlights the limited LLM performance in citing the correct documents necessary to answer the question. Additionally, in many legal applications, the ability to correctly ground a response based on relevant law documents is critical, highlighting the need to improve LLM to improve its citation correctness. With proper document citation, we suspect that this could potentially lead to better QA capability.

This work explores how reinforcement learning (RL) can be recontextualized to meet the practical demands of legal QA, where citation accuracy is critical. We align LLMs toward citation-sensitive outputs using Group-Relative Policy Optimization (GRPO) (Shao et al., 2024), with reward shaping tailored to legal citation structure and response quality (Yasui et al., 2019).

Our central question is: *How can we align large language models for domain-specific question answering, such as Thai legal QA, in a way that balances alignment quality, cost, and real-world applicability?* To answer this, we conduct two studies comparing alignment strategies with different reward designs under practical training constraints.

Study 1: Cost-effective alignment via semantic reward: We examine whether semantic similarity can serve as an efficient substitute for large judge models, particularly in in-domain settings. We compare Coverage, Consistency, and Citation F1 under both reward conditions.

Study 2: Reward composition ablation: We assess how different components of the reward signal, citation-only vs. full answer reward, affect model performance under judge-based supervision. This isolates the contribution of factual correctness signals in reward shaping.

2 Related Work

Enhancing LLM legal citation performance. A growing body of work seeks to improve citation verifiability in legal QA. CitaLaw (Zhang et al., 2025) adapts the ALCE benchmark (Gao et al., 2023) to the legal domain, introducing a syllogism-based citation metric and supporting both statutes and case law. ALCE evaluates grounding via an NLI verifier, requiring every claim to be backed by retrieved evidence. Shareghi et al. (2024) compare citation accuracy across three retrieval setups, retriever-only, LLM query-rewrite, and hybrid, and find that task-specific instruction tuning boosts citation accuracy, particularly in Australian case law. LegalBench-RAG (Pipitone and Alami, 2024) isolates retriever contributions by testing expert-annotated snippets under varying chunking and top-k settings, revealing a retrieval-imposed ceiling on citation F1.

Usage of embedding-based reward models. Yasui et al. (2019) finetune BERT (Devlin et al., 2019) on Semantic Textual Similarity (STS) and employ the tuned model as a REINFORCE reward for machine translation. Kumar and Subramaniam (2019) optimize a summarizer using BERTScore (Zhang et al., 2020), achieving higher fluency and lower redundancy than ROUGE-reward baselines. More recently, Sun et al. (2025) distil preference scores from the "gold" reward model of Dong et al. (2023, 2024) into lightweight proxies, an MLP and a LightGBM, that take paired Gemma-2B embeddings as input, achieving judge-level quality. These studies show that inexpensive embedding-based rewards can rival LLM judges in generation tasks, though their integration into modern preference optimization frameworks remains under-explored.

3 Our Approach

We frame Thai legal question answering (QA) as a citation-sensitive generative task. The model must produce correct free-form responses and cite the relevant legal statutes using official Thai citation formats. To align model outputs with these two requirements, we design two modular reward functions, citation accuracy and response quality, which are jointly optimized using Group-Relative Policy Optimization (GRPO).

3.1 Citation Accuracy Reward Functions

We design a multi-component verifiable reward function that ensures correct legal citation. In particular, our reward formulation decomposes citation quality into three measurable dimensions:

- **Format Reward** $f_1(x)$ = 1 if the output x follows the correct XML format. $f_1(x) = 0$ otherwise.
- **Non-Hallucination Reward** $f_2(x) = 0.5$ if $f_1(x) = 1$ and x cites one of the law provisions in the retrieval results. $f_2(x) = 0$ otherwise.
- **Citation F1 Reward** $f_3(x) = F_1$ score of the citation in x.

3.2 Response Quality Reward Functions

In addition to the citation accuracy reward, we also design a reward to ensure that the quality of the response is acceptable given the reference answer from the ground truth. While strong judges like preference models or advanced reasoning LLMs (e.g., OpenAI o1 (OpenAI et al., 2024), Deepseek R1 (DeepSeek-AI et al., 2025)) are too slow or costly for online training, we explore more computationally efficient proxies. We propose to use semantic similarity between generated and ground-truth responses as a reward instead. Additionally, we use coverage and contradiction metrics used in Akarajaradwong et al. (2025) directly as reward functions.

- **Semantic Similarity Reward** $0 < g_1(x) < 1$ computes the similarity score between the generated answer text and the ground-truth answer using an embedding model.
- **Coverage Reward** g_2 measures semantic coverage between generated response x and ground-truth responses $\hat{x}$ whether x is *no coverage* ($g_2(x, \hat{x}) = 0$), *partial coverage* ($g_2(x, \hat{x}) = 0.5$), or *full coverage* ($g_2(x, \hat{x}) = 1$) following Laban et al. (2024); Akarajaradwong et al. (2025).
- **Contradiction Reward** $g_3(x, \hat{x}) = 1$ if x does not contradict $\hat{x}$. $g_3(x, \hat{x}) = 0$ otherwise.

4 Experimental Setup

Training Data and Benchmark: We use WangchanX-Legal-ThaiCCL-RAG (Akarajaradwong et al., 2025) as a training set. One instance of the data contains a question, a ground-truth relevant legal sections, a reference answer. When preparing the training set, we construct prompts using the question and top retrieved sections instead of the ground-truth law sections. We use BGE-M3 (Chen et al., 2024) with a multi-head

strategy (dense/sparse/ColBERT weights set to 0.4, 0.2, 0.4) to retrieve top 10 relevant law sections. The ground-truth sections are used for citation evaluation/reward, and the reference answer is used for answer evaluation/reward. The query construction process is detailed in Appendix C. Qwen2.5-72B-instruct was used an LLM judge for coverage and contradiction reward.

For the benchmark, we utilize **NitiBench**[1] dataset (Akarajaradwong et al., 2025), specifically designed for Thai Legal QA. The benchmark contains two splits:

- **NitiBench-CCL:** Focuses on general Thai corporate/commercial law.
- **NitiBench-Tax:** Comprises complex, multi-positive Thai tax rulings. Used exclusively as a test set to evaluate generalization to very complex legal reasoning tasks.

Evaluation Metrics: We adopt the End-to-End (E2E) metrics from NitiBench (Akarajaradwong et al., 2025). However, instead of using a contradiction score, we use *Consistency Score*, the inverse of contradiction. This is averaged with Citation F1 and Coverage Score to calculate the *Joint Score*. Each metric is described as follows.

- **Citation F1:** F1-score of cited legal sections compared to the ground truth.
- **Coverage:** Reference answer overlap between generated and ground-truth answers based on a 0/50/100 scale. the value was then normalized to range from 0 to 1.
- **Consistency:** Factual consistency of the generated answer with the ground truth. Calculated as `1 - Contradiction`, leveraging the Contradiction score from NitiBench where 0: No-Contradiction, 1: Contradiction.
- **Joint Score:** An average of the metrics above.

Each model configuration was run 3 times on NitiBench-CCL and NitiBench-Tax using vLLM (Kwon et al., 2023) with different random seeds (Appendix B.3 for details). We also used GPT-4o (`gpt-4o-2024-08-06`) as the judge, with NitiBench prompts, ensuring consistency and comparability with the original benchmark.

Training Objectives The LLMs in our experiments are qwen2.5-7b-instruct, typhoon2-qwen2.5-7b-instruct, OpenThaiGPT1.5-7B. Post-training is done via Low-Rank Adaptation (LoRA) (Hu et al., 2021) (see Appendix B.1 for LoRA configuration). All GRPO setups was trained using Unsloth (Daniel Han and team, 2023) on a single NVIDIA A100 80gb GPU. All training hyperparameters can be found in Appendix B.2. As a strong baseline, we instruction-finetuned the LLMs on the same training dataset with LoRA for three epochs.

[1] https://huggingface.co/datasets/VISAI-AI/nitibench

5 Results

This section presents a comparison between the baseline performance and our proposed method. Detailed results are provided in Table 1; Table 5 includes relative gains and standard deviations.

Note that the Citation F1 metric is inherently limited by the performance of the upstream BGE-M3 retriever, which achieves an F1 score of 0.922 on NitiBench-CCL and 0.481 on NitiBench-Tax. This represents the theoretical upper bound for Citation F1 that the LLM could achieve, as it cannot cite documents not provided by the retriever.

5.1 Cost-effective alignment via semantic reward (Study 1)

Semantic similarity performs competitively in-domain. On the NitiBench-CCL test set, GRPO models trained with semantic reward perform on par with or better than those using judge-based rewards across all three base models. For example, the *+LoRA GRPO (semantic reward)* variant of Typhoon2 yields the highest Coverage (0.774) and a strong Joint Score (0.777), outperforming its judge-based counterpart. For OpenThaiGPT1.5, the semantic reward variant achieves a higher Joint Score (0.760 vs. 0.753) and comparable Citation F1. Even for the language-generic Qwen2.5, semantic reward provides competitive results across all metrics. In addition, we observe that GRPO, particularly with cov/con reward, substantially narrows the performance gap between 7B-scale models and proprietary systems (GPT-4o, Gemini 1.5 Pro, and Claude 3.5), indicating its potential as a cost-effective alternative for legal QA. These findings suggest that when reference answers are semantically well-aligned with the context, lightweight semantic rewards offer effective, low-cost supervision for in-domain legal QA.

Cov/Con reward improves generalization. On the out-of-distribution NitiBench-Tax set, GRPO models using cov/con reward consistently outperform their semantic reward counterparts. Although semantic reward can lead on specific metrics, cov-/con reward yields a stronger Joint Score. For

Model	Citation F1 ↑	Coverage ↑	Consistency ↑	Joint score ↑	Citation F1 ↑	Coverage ↑	Consistency ↑	Joint score ↑
	NitiBench-CCL				**NitiBench-Tax**			
qwen2.5-7b-instruct	0.410	0.591	0.840	0.614	0.211	0.333	0.573	0.373
+LoRA IT	0.569	0.583	0.834	0.662	0.098	0.287	0.507	0.297
+LoRA GRPO (cov/con reward)	0.680	0.632	0.860	0.724	0.168	0.293	0.563	0.342
+LoRA GRPO (semantic reward)	0.715	0.720	0.823	0.753	0.156	0.317	0.567	0.346
typhoon2-qwen2.5-7b-instruct	0.360	0.559	0.855	0.591	0.127	0.333	0.547	0.336
+LoRA IT	0.574	0.621	0.857	0.684	0.107	0.263	0.567	0.312
+LoRA GRPO (cov/con reward)	0.651	0.709	**0.903**	0.755	0.204	**0.380**	0.583	**0.389**
+LoRA GRPO (semantic reward)	0.683	**0.774**	0.876	**0.777**	0.211	0.363	0.493	0.356
openthaigpt1.5-qwen2.5-7b-instruct	0.430	0.556	0.823	0.603	0.185	0.337	0.540	0.354
+LoRA IT	0.561	0.593	0.837	0.664	0.104	0.327	0.580	0.337
+LoRA GRPO (cov/con reward)	**0.720**	0.668	0.871	0.753	0.209	0.367	0.560	0.378
+LoRA GRPO (semantic reward)	0.702	0.721	0.855	0.760	**0.248**	0.250	**0.600**	0.366
+LoRA GRPO (semantic + cov/con rewards)	0.691	0.611	0.853	0.718	0.183	0.307	0.527	0.339
+LoRA GRPO (w/o answer reward)	0.670	0.548	0.804	0.674	0.166	0.313	0.533	0.338
`gpt-4o-2024-08-06`	0.714	0.852	0.945	0.837	0.438	0.500	0.540	0.492
`gemini-1.5-pro-002`	0.651	0.865	0.952	0.823	0.332	0.440	0.520	0.431
`claude-3-5-sonnet-20240620`	0.595	0.897	0.960	0.817	0.457	0.510	0.560	0.509

Table 1: Comparison (average on 3 runs) on Nitibench-CCL and Nitibench-Tax: Baseline vs. IT, GRPO (cov-/con reward), GRPO (semantic reward). Relative performance gains over baseline are indicated. Comparison provided against 3 proprietary LLM results from Akarajaradwong et al. (2025) on the same settings. Also shows OpenThaiGPT1.5-7B-Instruct with combined (semantic + cov/con) vs. LoRA GRPO (w/o answer reward).

instance, Typhoon2's *+LoRA GRPO (cov/con reward)* variant achieves a higher Joint Score (0.389 vs. 0.356) compared to the semantic version. This pattern extends to OpenThaiGPT1.5, where cov-/con reward offers greater robustness to distribution shifts, but not on Qwen2.5. This might be due to the fact that Qwen2.5, lacking Thai-specific pretraining, exhibits weaker priors for legal citation tasks. These results suggest that coverage and consistency supervision helps models generalize better in more structurally diverse legal contexts, though at a higher computational cost. We further investigate how Semantic Similarity correlates with Coverage and Consistency scores in Appendix F.

5.2 Reward composition ablation (Study 2)

To understand reward contributions, we performed ablations on OpenThaiGPT1.5-7B (see Table 1), comparing our main GRPO variants against configurations using: (1) combined semantic and coverage/consistency rewards ('semantic + cov/con rewards'), and (2) only citation-related rewards ('w/o answer reward').

Reward composition impacts alignment effectiveness. Combining semantic and cov/con rewards without reward tuning underperforms both individual configurations, likely due to imbalanced scaling between the two signals. This finding highlights the importance of careful reward calibration when mixing objectives.

Citation-only reward is insufficient. When we remove the answer-level component and retain only the citation reward, we observe a modest gain in Citation F1, but at the cost of significantly lower Coverage and Consistency, resulting in a reduced Joint Score. While in-domain Citation F1 improved over baseline, Coverage and Consistency degraded below baseline levels. This variant also performed the worst among GRPO configurations on CCL citation and failed to generalize on Tax. This strongly indicates that **generation quality aspects are coupled**; optimizing citations alone harms overall quality and generalization, demonstrating the need for answer quality rewards even to maximize citation performance within GRPO.

6 Conclusion

We study how to affordably align large language models (LLMs) for citation-sensitive legal question answering in Thai. Using Group-Relative Policy Optimization (GRPO), we compare two reward strategies: a lightweight semantic similarity proxy (BGE-M3) and a large LLM judge model (Qwen2.5-72B-Instruct) scoring coverage and consistency. Our results show that semantic rewards yield comparable in-domain performance to judge-based supervision, while requiring significantly less compute. In contrast, cov/con rewards offer better generalization on out-of-distribution tasks but at a higher cost. These findings offer practical guidance for aligning LLMs in legal QA, balancing performance and cost under domain-specific constraints, and show that GRPO can meaningfully close the gap between compact open models and proprietary systems.

Limitations

While this study provides valuable insights into applying GRPO for Thai Legal QA, we acknowledge certain limitations primarily stemming from constraints on computational resources and time during the experimental phase.

First, our exploration of combining different reward signals for answer quality. Specifically, the semantic similarity reward and the coverage/consistency rewards from the Qwen2.5-72B-Instruct judge were limited. The ablation study used a naive summation without tuning, which underperformed relative to individual signals. Due to resource constraints, we were unable to explore alternative reward calibration strategies such as weighting, normalization, or learning rate adjustments. A well-tuned combination may offer synergistic benefits, but this remains unexplored.

Second, our experiments focused exclusively on applying GRPO to models that had already undergone instruction tuning. We applied GRPO only to models that had already undergone instruction tuning. We did not evaluate applying GRPO directly to base models (e.g., (DeepSeek-AI et al., 2025)). Investigating its effect from different model initialization states may yield further insights, but was beyond our current scope.

Third, we used the standard GRPO algorithm as described by Shao et al. (2024). While conducting our experiments, an improved variant named "Dr. GRPO" (Done Right GRPO) was proposed (Liu et al., 2025), specifically designed to address optimization biases present in the original GRPO formulation, particularly those related to response length normalization, which can affect token efficiency. Due to the timing of its release relative to our experimental runs and resource limitations, we were unable to incorporate Dr. GRPO into our comparisons. We acknowledge the potential biases in standard GRPO identified by Liu et al. (2025) and recognize that employing Dr. GRPO might yield different results, particularly regarding token efficiency and potentially performance dynamics.

These limitations reflect the demonstrative nature of this study, which aims to assess the potential of GRPO for citation-sensitive legal QA under domain-specific constraints. Addressing them may deepen our understanding of GRPO's behavior in legal settings and inform strategies for best utilizing it in practice.

References

Pawitsapak Akarajaradwong, Pirat Pothavorn, Chompakorn Chaksangchaichot, Panuthep Tasawong, Thitiwat Nopparatbundit, and Sarana Nutanong. 2025. Nitibench: A comprehensive study of llm framework capabilities for thai legal question answering. *Preprint*, arXiv:2502.10868.

Jianlv Chen, Shitao Xiao, Peitian Zhang, Kun Luo, Defu Lian, and Zheng Liu. 2024. Bge m3-embedding: Multi-lingual, multi-functionality, multi-granularity text embeddings through self-knowledge distillation. *Preprint*, arXiv:2402.03216.

Pierre Colombo, Telmo Pessoa Pires, Malik Boudiaf, Dominic Culver, Rui Melo, Caio Corro, Andre F. T. Martins, Fabrizio Esposito, Vera Lúcia Raposo, Sofia Morgado, and Michael Desa. 2024. Saullm-7b: A pioneering large language model for law. *Preprint*, arXiv:2403.03883.

Counsel AI Corporation. 2025. Harvey ai. Accessed: 2025-04-25.

Michael Han Daniel Han and Unsloth team. 2023. Unsloth.

DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, and 181 others. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *Preprint*, arXiv:2501.12948.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. *Preprint*, arXiv:1810.04805.

Hanze Dong, Wei Xiong, Deepanshu Goyal, Yihan Zhang, Winnie Chow, Rui Pan, Shizhe Diao, Jipeng Zhang, Kashun Shum, and Tong Zhang. 2023. Raft: Reward ranked finetuning for generative foundation model alignment. *Preprint*, arXiv:2304.06767.

Hanze Dong, Wei Xiong, Bo Pang, Haoxiang Wang, Han Zhao, Yingbo Zhou, Nan Jiang, Doyen Sahoo, Caiming Xiong, and Tong Zhang. 2024. Rlhf workflow: From reward modeling to online rlhf. *Preprint*, arXiv:2405.07863.

Tianyu Gao, Howard Yen, Jiatong Yu, and Danqi Chen. 2023. Enabling large language models to generate text with citations. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 6465–6488, Singapore. Association for Computational Linguistics.

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.

Vivek Kumar and Arjun Subramaniam. 2019. Abstractive summarisation with bertscore reward. CS229 Project Report.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*.

Harvard Law School Library Innovation Lab. 2024. Caselaw access project. Accessed: 2024-08-05.

Philippe Laban, Alexander R. Fabbri, Caiming Xiong, and Chien-Sheng Wu. 2024. Summary of a haystack: A challenge to long-context llms and rag systems. *Preprint*, arXiv:2407.01370.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2021. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Preprint*, arXiv:2005.11401.

Zichen Liu, Changyu Chen, Wenjun Li, Penghui Qi, Tianyu Pang, Chao Du, Wee Sun Lee, and Min Lin. 2025. Understanding r1-zero-like training: A critical perspective. *Preprint*, arXiv:2503.20783.

NVIDIA. 2024. Tensorrt-llm.

OpenAI, :, Aaron Jaech, Adam Kalai, Adam Lerer, Adam Richardson, Ahmed El-Kishky, Aiden Low, Alec Helyar, Aleksander Madry, Alex Beutel, Alex Carney, Alex Iftimie, Alex Karpenko, Alex Tachard Passos, Alexander Neitz, Alexander Prokofiev, Alexander Wei, Allison Tam, and 244 others. 2024. Openai o1 system card. *Preprint*, arXiv:2412.16720.

Nicholas Pipitone and Ghita Houir Alami. 2024. Legalbench-rag: A benchmark for retrieval-augmented generation in the legal domain. *Preprint*, arXiv:2408.10343.

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

Ehsan Shareghi, Jiuzhou Han, and Paul Burgess. 2024. Methods for legal citation prediction in the age of llms: An australian law case study. *Preprint*, arXiv:2412.06272.

Hao Sun, Yunyi Shen, Jean-Francois Ton, and Mihaela van der Schaar. 2025. Reusing embeddings: Reproducible reward model research in large language model alignment without gpus. *Preprint*, arXiv:2502.04357.

Go Yasui, Yoshimasa Tsuruoka, and Masaaki Nagata. 2019. Using semantic similarity as reward for reinforcement learning in sentence generation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, pages 400–406, Florence, Italy. Association for Computational Linguistics.

Kepu Zhang, Weijie Yu, Sunhao Dai, and Jun Xu. 2025. Citalaw: Enhancing llm with citations in legal domain. *Preprint*, arXiv:2412.14556.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert. *Preprint*, arXiv:1904.09675.

A Efficiency of Reward Signal Proxies

The practicality of RL hinges on reward computation efficiency. We observed a stark difference between using BGE-M3 semantic similarity versus the large Qwen2.5-72B-Instruct judge for coverage/consistency rewards. As shown in Figure 1, the BGE-M3 approach required significantly fewer resources per GRPO policy training: **104 GPU-hours** (1x A100 80GB GPU), costing approximately **$85**. In contrast, using the Qwen2.5-72B-Instruct judge demanded **264 total GPU-hours** (2x A100 80GB GPUs for 132 hours) - nearly 2.5x the compute time - costing roughly **$216**[2]. This setup was necessary because one GPU was dedicated solely to hosting the 72B judge model as an online reward server with int4_wo precision using TensorRT-LLM (NVIDIA, 2024), while the other GPU handled the training process.

Figure 1: Comparison of total GPU hours and estimated training cost for GRPO variants using different answer reward signals.

This large disparity arises because BGE-M3 embedding calculation is fast, adding minimal latency to the RL loop, while Qwen2.5-72B-Instruct inference for each sample creates a major bottleneck,

[2]Based on A100 80GB PCIE median rental cost of $0.82/hr via `https://vast.ai/pricing/gpu/A100-PCIE` accessed April 2025.

requiring more hardware and time. While a large judge might offer reward signals closer to final evaluation metrics, its computational cost significantly hinders online RL training. BGE-M3 semantic similarity, despite being a proxy, proves vastly more efficient. Its strong performance, especially in-domain, confirms its value as a cost-effective method for injecting an answer quality signal during GRPO training.

B Hyperparameters

B.1 LoRA configuration

We applied LoRA to attention layers (`q_proj`, `k_proj`, `v_proj`, `gate`, `up_proj`, `down_proj`), rank $r = 256$ with 16-bit precision.

B.2 Training Hyperparameters

Common parameters related to LoRA configuration, precision, optimizer betas, and data handling were kept consistent where applicable.

Hyperparameter	GRPO Value	IT Value
Learning Rate (lr)	5.00E-06	1.00E-05
LR Scheduler Type	`constant_with_warmup`	`cosine`
Max Gradient Norm	0.2	1.0
Epochs	1	3
Rollout Batch Size	10	N/A
Batch Size	N/A	4
Max Prompt Length	8192	8192
Max Completion Length	2048	2048
LoRA Rank (r)	256	256
Precision	bfloat16	bfloat16
Retrieval Top-k	10	10
Gradient Accumulation Steps	1	1
Weight Decay	0.1	0.1
Warmup Ratio	0.1	0.1
Adam Beta1	0.9	0.9
Adam Beta2	0.99	0.99

Table 2: Comparison of Key Hyperparameters for IT and GRPO Training.

B.3 Inferencing Hyperparameters

These settings were applied consistently across all model configurations (Baseline, IT, GRPO). The following parameters were used for text generation:

Generation Seeds: Inference was repeated three times for each model configuration using the following distinct random seeds: `69420`, `69421`, and `69422`. The final reported metrics are the mean across these 3 runs.

Retrieval Top-k: Set to `10`, same as the `Retrieval Top-k` in the training hyperparameter.

Temperature: Set to `1.0` for standard diversity in the output.

C Query Construction

To manage computational constraints, input queries are capped at 8192 tokens. If the retrieved top 10 sections exceed this limit, we iteratively replace the longest nonground-truth section with the next highest-ranked section from the retriever, ensuring all ground-truth sections are retained while staying within the token limit. The target output format for both IT and GRPO is structured XML-like text including `<reasoning>`, `<answer>`, and `<citation>` tags. Additional details regarding input and output formatting are provided in Appendix D.

D Input and Output Formats

This section provides concrete examples of the input query structure fed to the models and the target output format used during fine-tuning (both IT and GRPO), complementing the description in §4.

D.1 Example Input Query Structure

The following illustrates the format of the input provided to the models. This example assumes the context retrieval resulted in $k = 5$ relevant sections after length management. The `<context>` tags contain the actual text content of the corresponding legal section. The `<law_code>` tags contain unique integer identifiers assigned to each distinct legal section within our corpus; these identifiers are used as keys and do not necessarily correspond to official statutory section numbers.

```
What is the difference between financial
    institution business and financial
    business?

Relevant sections
<law_code>1</law_code><context>...</context>
<law_code>2</law_code><context>...</context>
<law_code>3</law_code><context>...</context>
<law_code>4</law_code><context>...</context>
<law_code>5</law_code><context>...</context>
```

D.2 Example Target Output Structure

The models were trained to generate outputs adhering to the following XML-like structure. This format separates the reasoning process, the final answer, and the cited sources.

```
<reasoning>
The laws related to the method for director
    resignation are ...
</reasoning>
```

```
<answer>
According to Section 1153/1 of the Civil and
    Commercial Code and ...
</answer>
<citation>
<law_code>2</law_code>
<law_code>5</law_code>
</citation>
```

Note: The `<reasoning>` block contains the model's generated explanation or thought process. The `<answer>` block contains the final synthesized answer to the query. The `<citation>` block lists the `<law_code>` identifiers that the model cites as sources for its answer. During IT, this structure represents the target output. During GRPO, adherence to this format and the correctness of the content within the tags (`<answer>` and `<citation>`) are evaluated by the reward functions.

E Evaluation of Qwen-72B as an Automated Judge

To assess the viability of using Qwen2.5-72B-Instruct as an online judge for generating Coverage and Consistency rewards in GRPO (§3), we compared its judgment reliability against `gpt-4o-2024-08-06` on the NitiBench-CCL dataset, as it achieved the highest performance among judges evaluated in the original NitiBench paper (Akarajaradwong et al., 2025). We follow NitiBench's decoding hyperparameters: `temperature = 0.5`, `seed = 69420`, and `max_completion_tokens = 2048`.

As shown in Table 3, Qwen-72B achieved high reliability, closely matching GPT-4o. For **Coverage**, Qwen-72B reached an F1-score of 0.84 (vs. 0.88 for GPT-4o), and for **Consistency**, it scored 0.97 (vs. 0.98 for GPT-4o). These results demonstrate that Qwen2.5-72B-Instruct functions as a reliable automated judge for these metrics on this dataset, validating its use for providing sufficiently accurate reward signals during GRPO training as an alternative to external API calls.

Model	Metric	Precision	Recall	F1-score	Support
NitiBench-CCL					
gpt-4o-2024-08-06	Coverage	.88	.88	.88	200
	Consistency	.98	.97	.98	150
Qwen2.5-72B-Instruct	Coverage	.85	.83	.84	200
	Consistency	.98	.97	.97	150

Table 3: Performance comparison of GPT-4o (`gpt-4o-2024-08-06`) and Qwen2.5-72B-Instruct as automated judges for Coverage and Consistency metrics on the NitiBench-CCL dataset.

F Correlation of Semantic Similarity with Coverage and Consistency

We investigated using BGE-M3 semantic similarity as an efficient proxy reward for answer quality during GRPO, avoiding costly LLM-judges for online training. To validate this proxy, we analyzed its correlation with ground-truth Coverage and Consistency scores (determined by offline judge) on both NitiBench test sets in Figures 2, 3 and 4.

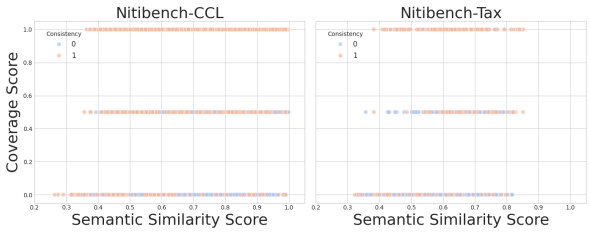

Figure 2: Semantic Similarity vs. Coverage scores, colored by Consistency, on (a) NitiBench-CCL and (b) NitiBench-Tax. A positive trend between similarity and coverage is more evident on CCL than on Tax.

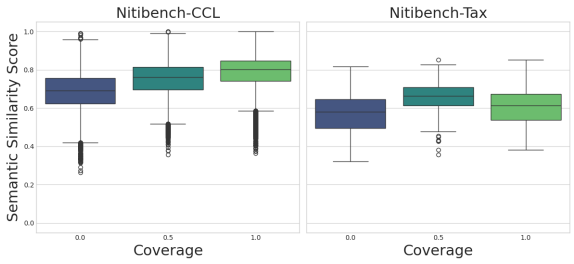

Figure 3: Semantic Similarity distributions by Coverage score level on (a) NitiBench-CCL and (b) NitiBench-Tax. Median similarity tends to increase with coverage on CCL, a trend not observed on Tax.

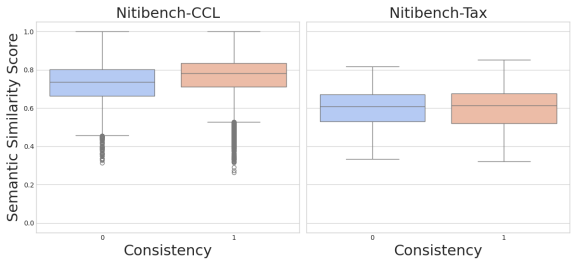

Figure 4: Semantic Similarity distributions by Consistency score on (a) NitiBench-CCL and (b) NitiBench-Tax. Consistent answers on CCL show higher similarity; this distinction is less clear on Tax.

For **NitiBench-CCL**, we observed a noticeable positive correlation. Higher semantic similarity generally aligns with higher Coverage and Consistency scores, as seen in both scatter and box plots.

This suggests semantic similarity provides a meaningful, though imperfect, signal for answer quality on this simpler, in-domain dataset, supporting its use as a proxy reward here.

Conversely, for the more complex **NitiBench-Tax**, semantic similarity showed negligible correlation with Coverage or Consistency. The scatter plot lacked clear trends, and box plots revealed largely overlapping distributions for semantic similarity across different quality levels.

This contrast demonstrates that the utility of semantic similarity as a reward proxy is highly context-dependent. While adequate for simpler tasks (NitiBench-CCL), it fails to capture crucial aspects of correctness and factual consistency on complex reasoning tasks requiring synthesis (NitiBench-Tax), where semantic overlap alone is insufficient. The limitations of this efficient proxy become apparent on harder generalization problems. Appendix G provides a detailed comparison highlighting the increased complexity of NitiBench-Tax relative to NitiBench-CCL.

G Complexity of NitiBench-Tax over NitiBench-CCL

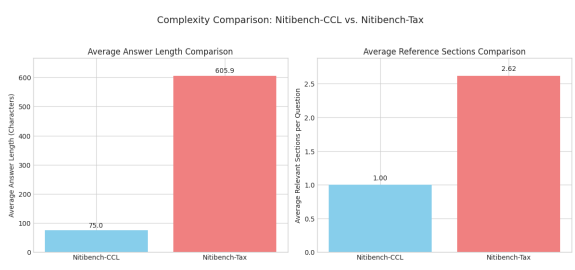

Figure 5: Complexity Comparison of NitiBench-CCL vs. NitiBench-Tax.

While both NitiBench-CCL and NitiBench-Tax evaluate Thai Legal QA, the NitiBench-Tax dataset presents a significantly more complex challenge, designed specifically to test model generalization and deeper reasoning capabilities (see Figure 5 for answer length and section per answer comparison). This difference stems from several key aspects of their origin and structure:

1. **Dataset Origin and Curation:**
    - **NitiBench-CCL:** This dataset was curated manually by legal experts who crafted question-answer pairs primarily based on single, specific legal sections from a defined corpus of 35 financial laws. The process involved a two-tiered expert review to ensure quality. While its corresponding training data (from WangchanX-Legal-ThaiCCL-RAG[3]) could be multi-label due to semi-automated generation, the test set used for evaluation predominantly consists of single-label instances.
    - **NitiBench-Tax:** This dataset originates from real-world tax rulings scraped directly from the Thai Revenue Department's official website[4] (cases from 2021 onwards). These represent authentic inquiries and official responses, reflecting the complexity of actual tax law application. The curation involved extracting relevant cited sections and condensing the official responses using an LLM, after filtering out non-interpretive cases.

    The use of real, official rulings in NitiBench-Tax inherently introduces more complex scenarios and language compared to the expert-crafted, typically single-provision-focused questions in the NitiBench-CCL test set.

2. **Answer Length and Complexity:** The complexity difference is reflected in the average length of the ground-truth answers (after condensation). The average answer length in **NitiBench-CCL is approximately 75 characters**, whereas in **NitiBench-Tax, it is roughly 606 characters** - over eight times longer on average. This suggests that Tax answers inherently require significantly more detail and potentially cover more sub-points derived from the underlying complex rulings.

3. **Multi-Label Nature (Sections per Answer):** This is a critical quantitative differentiator. The NitiBench-CCL test set is explicitly single-label, with an average of **1 ground-truth relevant legal section** per question. In contrast, NitiBench-Tax is inherently multi-label, with an average of **2.62 relevant sections** per case. This requires models not just to identify relevant sections but to synthesize information and reason across multiple legal provisions simultaneously, significantly increasing the reasoning complexity compared to the single-label focus of CCL.

[3] https://huggingface.co/datasets/airesearch/WangchanX-Legal-ThaiCCL-RAG

[4] https://www.rd.go.th

model	Citation F1 ↑	SD	gains (%)	Coverage ↑	SD	gains (%)	Consistency ↑	SD	gains (%)	Joint score	gains (%)
Nitibench-CCL											
openthaigpt1.5-qwen2.5-7b-instruct	0.4299	0.0048		0.5556	0.0010		0.8234	0.0048		0.6030	
+LoRA GRPO (semantic reward)	**0.7017**	0.0016	63.23	**0.7214**	0.0041	29.84	**0.8554**	0.0021	3.89	**0.7595**	25.96
+LoRA GRPO (semantic reward, citation first)	0.6545	0.0044	52.25	0.7065	0.0053	27.16	0.8528	0.0028	3.57	0.7379	22.39
Nitibench-Tax											
openthaigpt1.5-qwen2.5-7b-instruct	0.1850	0.0247		**0.3367**	0.0519		0.5400	0.0849		0.3539	
+LoRA GRPO (semantic reward)	**0.2482**	0.0054	34.16	0.2500	0.0424	-25.74	**0.6000**	0.0490	11.11	0.3661	3.44
+LoRA GRPO (semantic reward, citation first)	0.2172	0.0146	17.43	0.2768	0.0026	-17.79	0.5333	0.0411	-1.24	0.3424	-3.24

Table 4: Comparison of GRPO (semantic reward) performance on OpenThaiGPT1.5-7B using the default output format (reasoning->answer->citation) versus a modified format placing citations before the answer (reasoning->citation->answer).

In summary, the combination of using real-world, complex tax rulings as source material and its inherent multi-label requirement (demanding reasoning across multiple sections) makes NitiBench-Tax a substantially harder benchmark than NitiBench-CCL for evaluating advanced legal reasoning and generalization abilities.

H Impact of Citation and Answer Position in Output Format

The standard output format used in our experiments follows the structure: reasoning -> answer -> citation (as in Appendix D.2), where the model first provides its reasoning, then the synthesized answer, and finally the supporting citations. To investigate whether the position of the citation block relative to the answer block influences performance, we conducted an additional experiment.

We modified the target output structure to: reasoning -> citation -> answer, placing the citation block immediately after the reasoning and before the final answer. We then retrained the OpenThaiGPT1.5-7B-Instruct model using the GRPO (semantic reward) configuration with this modified "citation-first" target format. All other training parameters remained identical to the corresponding main experiment run.

The results of this comparison are presented in Table 4. The data clearly indicates that altering the standard format to place citations before the answer consistently resulted in **lower performance across nearly all metrics** on both the NitiBench-CCL and NitiBench-Tax datasets compared to the default format where citations appear last. Notably, Citation F1, Coverage, and the overall Joint Score decreased in the "citation-first" configuration. On the challenging NitiBench-Tax set, this format led to performance even worse than the baseline in terms of Joint Score (-3.24% gain).

While the exact reasons require deeper analysis, this finding suggests that the default structure (reasoning -> answer -> citation) may provide a more natural or effective flow for the model during generation and training. It's possible that generating the answer text first helps the model consolidate the information needed before explicitly listing the supporting citations. Regardless, based on these results, maintaining the structure with the citation block at the end appears preferable for achieving optimal performance with our GRPO approach.

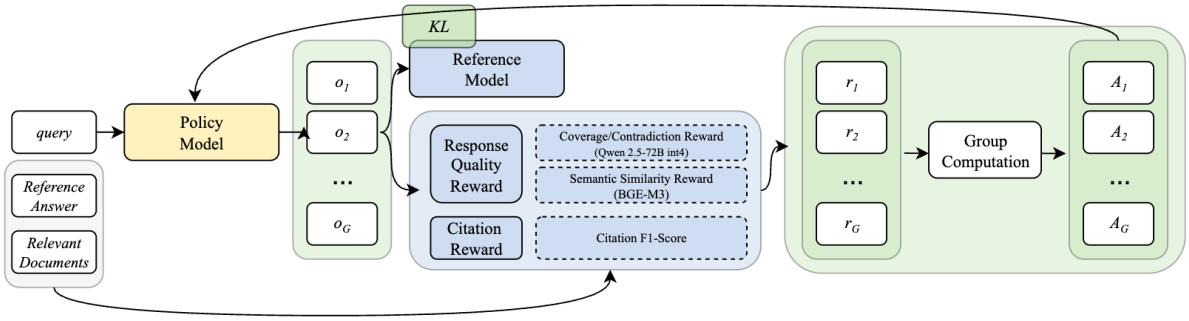

Figure 6: Demonstration of our proposed method. We use GRPO objectives with specialized reward to align LLM towards better citation and response using **Response Quality Reward** (§3.2) and **Citation Reward** (§3.1).

Model	Citation F1 ↑	SD	gains (%)	Coverage ↑	SD	gains (%)	Consistency ↑	SD	gains (%)	Joint score ↑	gains (%)
Nitibench-CCL (In-Domain)											
qwen2.5-7b-instruct	0.4103	0.0015		0.5908	0.0041		0.8402	0.0030		0.6138	
+LoRA SFT	0.5691	0.0040	38.70	0.5832	0.0075	-1.29	0.8341	0.0024	-0.72	0.6622	7.88
+LoRA GRPO (cov/con reward)	0.6796	0.0020	65.63	0.6322	0.0010	7.00	0.8598	0.0009	2.34	0.7239	17.94
+LoRA GRPO (semantic reward)	0.7146	0.0009	74.14	0.7197	0.0023	21.81	0.8232	0.0024	-2.02	0.7525	22.60
typhoon2-qwen2.5-7b-instruct	0.3597	0.0042		0.5587	0.0061		0.8553	0.0076		0.5912	
+LoRA SFT	0.5744	0.0028	59.71	0.6214	0.0030	11.23	0.8572	0.0030	0.22	0.6843	15.75
+LoRA GRPO (cov/con reward)	0.6514	0.0013	81.10	0.7092	0.0039	26.95	**0.9032**	0.0019	5.60	0.7546	27.63
+LoRA GRPO (semantic reward)	0.6828	0.0028	89.84	**0.7735**	0.0012	38.45	0.8757	0.0028	2.38	**0.7773**	31.48
openthaigpt1.5-qwen2.5-7b-instruct	0.4299	0.0048		0.5556	0.0010		0.8234	0.0048		0.6030	
+LoRA SFT	0.5613	0.0069	30.56	0.5930	0.0024	6.73	0.8371	0.0031	1.66	0.6638	10.08
+LoRA GRPO (cov/con reward)	**0.7197**	0.0020	67.40	0.6680	0.0034	20.23	0.8705	0.0034	5.72	0.7527	24.84
+LoRA GRPO (semantic reward)	0.7017	0.0016	63.23	0.7214	0.0041	29.84	0.8554	0.0021	3.89	0.7595	25.96
gpt-4o-2024-08-06	0.7140			0.8520			0.9450			0.8370	
gemini-1.5-pro-002	0.6510			0.8650			0.9520			0.8227	
claude-3-5-sonnet-20240620	0.5950			0.8970			0.9600			0.8173	
Nitibench-Tax (Out-of-Domain)											
qwen2.5-7b-instruct	0.2110	0.0272		0.3333	0.0082		0.5733	0.0340		0.3726	
+LoRA SFT	0.0975	0.0192	-53.82	0.2867	0.0249	-13.99	0.5067	0.0094	-11.63	0.2969	-20.30
+LoRA GRPO (cov/con reward)	0.1678	0.0196	-20.47	0.2933	0.0047	-12.00	0.5633	0.0094	-1.74	0.3415	-8.34
+LoRA GRPO (semantic reward)	0.1555	0.0135	-26.31	0.3167	0.0249	-4.99	0.5667	0.0249	-1.16	0.3463	-7.05
typhoon2-qwen2.5-7b-instruct	0.1272	0.0150		0.3333	0.0411		0.5467	0.0249		0.3357	
+LoRA SFT	0.1072	0.0315	-15.71	0.2633	0.0205	-21.00	0.5667	0.0189	3.66	0.3124	-6.95
+LoRA GRPO (cov/con reward)	0.2035	0.0197	60.03	**0.3800**	0.0294	14.00	0.5833	0.0189	6.71	**0.3889**	15.85
+LoRA GRPO (semantic reward)	0.2113	0.0134	66.18	0.3633	0.0411	9.00	0.4933	0.0525	-9.76	0.3560	6.04
openthaigpt1.5-qwen2.5-7b-instruct	0.1850	0.0247		0.3367	0.0519		0.5400	0.0849		0.3539	
+LoRA SFT	0.1039	0.0387	-43.84	0.3267	0.0450	-2.97	0.5800	0.0283	7.41	0.3368	-4.81
+LoRA GRPO (cov/con reward)	0.2085	0.0328	12.73	0.3667	0.0205	12.24	0.5600	0.0748	3.70	0.3784	6.93
+LoRA GRPO (semantic reward)	**0.2482**	0.0054	34.16	0.2500	0.0424	-25.74	**0.6000**	0.0490	11.11	0.3661	3.44
gpt-4o-2024-08-06	0.4380			0.5000			0.5400			0.4927	
gemini-1.5-pro-002	0.3320			0.4400			0.5200			0.4307	
claude-3-5-sonnet-20240620	0.4570			0.5100			0.5600			0.5090	

Table 5: Full Performance comparison (avg ± SD, 3 runs) on Nitibench-CCL and Nitibench-Tax, extending Table 1: Baseline vs. SFT, GRPO (cov/con reward), GRPO (semantic reward). Relative performance gains over baseline are indicated.

Model	Citation F1 ↑	SD	gains (%)	Coverage ↑	SD	gains (%)	Consistency ↑	SD	gains (%)	Joint score ↑	gains (%)
Nitibench-CCL (In-Domain)											
openthaigpt1.5-qwen2.5-7b-instruct	0.4299	0.0048		0.5556	0.0010		0.8234	0.0048		0.6030	
+LoRA SFT	0.5613	0.0069	30.56	0.5930	0.0024	6.73	0.8371	0.0031	1.66	0.6638	10.08
+LoRA GRPO (cov/con reward)	**0.7197**	0.0020	67.40	0.6680	0.0034	20.23	**0.8705**	0.0034	5.72	0.7527	24.84
+LoRA GRPO (semantic reward)	0.7017	0.0016	63.23	**0.7214**	0.0041	29.84	0.8554	0.0021	3.89	**0.7595**	25.96
+LoRA GRPO (semantic + cov/con rewards)	0.6912	0.0024	60.77	0.6109	0.0049	9.95	0.8529	0.0032	3.58	0.7183	19.13
+LoRA GRPO (w/o answer reward)	0.6704	0.0022	55.95	0.5484	0.0042	-1.29	0.8037	0.0086	-2.39	0.6742	11.82
Nitibench-Tax (Out-of-Domain)											
openthaigpt1.5-qwen2.5-7b-instruct	0.1850	0.0247		0.3367	0.0519		0.5400	0.0849		0.3539	
+LoRA SFT	0.1039	0.0387	-43.84	0.3267	0.0450	-2.97	0.5800	0.0283	7.41	0.3368	-4.81
+LoRA GRPO (cov/con reward)	0.2085	0.0328	12.73	**0.3667**	0.0205	12.24	0.5600	0.0748	3.70	**0.3784**	6.93
+LoRA GRPO (semantic reward)	**0.2482**	0.0054	34.16	0.2500	0.0424	-25.74	**0.6000**	0.0490	11.11	0.3661	3.44
+LoRA GRPO (semantic + cov/con rewards)	0.1830	0.0048	-1.04	0.3067	0.3682	-8.91	0.5267	0.0499	-2.47	0.3388	-4.26
+LoRA GRPO (w/o answer reward)	0.1662	0.0090	-10.16	0.3133	0.0125	-6.93	0.5333	0.0189	-1.23	0.3376	-4.60

Table 6: Ablation results for OpenThaiGPT1.5-7B-Instruct on Nitibench-CCL and Nitibench-Tax. Compares LoRA GRPO performance using combined semantic and coverage/consistency (semantic + cov/con) rewards vs LoRA GRPO without any answer-specific reward (w/o answer reward).