COMPUTE-MATCHED EVALUATION OF TRANSFORM-AUGMENTED GRPO FOR MATHEMATICAL REASONING

FARS
Analemma
fars@analemma.ai

ABSTRACT

Transform-Augmented GRPO (TA-GRPO) improves mathematical reasoning by generating semantic transformations of training prompts and pooling advantages across variants. However, prior comparisons with standard GRPO are confounded by compute differences: TA-GRPO uses $4\times$ more rollouts per original prompt. We present a compute-matched evaluation where both methods consume identical total rollouts ($\sim$725K). Under this fair comparison, TA-GRPO achieves +2.02 percentage points higher Pass@32 than GRPO-Long (49.47% vs 47.45%), demonstrating that semantic transformations provide genuine benefits beyond additional compute. Ablation analysis reveals that 87% of this improvement stems from data augmentation (training on diverse problem reformulations), while only 13% comes from pooled advantage normalization. The advantage grows with inference-time compute (from +1.07pp at $k = 1$ to +2.02pp at $k = 32$), consistent with improved solution diversity.

WARNING: This paper was generated by an automated research system. The code is publicly available.[1]

1 INTRODUCTION

Reinforcement learning has emerged as a powerful paradigm for improving reasoning capabilities in large language models (Zhang et al., 2025). Group Relative Policy Optimization (GRPO) (Shao et al., 2024) has proven particularly effective for mathematical reasoning, achieving strong results by computing advantages relative to group statistics without requiring a separate critic network. Recently, Transform-Augmented GRPO (TA-GRPO) (Le et al., 2026) extended this approach by generating semantically equivalent variants of each training prompt through paraphrasing, variable renaming, and format changes, then pooling advantages across the entire group.

However, TA-GRPO's reported improvements over standard GRPO are confounded by a fundamental difference in compute: by generating N transformations per prompt and sampling rollouts for each variant, TA-GRPO uses $(N + 1)\times$ more rollouts per original prompt. With $N = 3$ transformations, this amounts to $4\times$ more total rollouts than standard GRPO trained for the same number of steps. This raises a critical question: does TA-GRPO's advantage stem from its semantic transformation mechanism, or simply from the additional compute?

In this paper, we design a compute-matched evaluation to isolate the true contribution of semantic transformations. We compare standard GRPO trained for 708 steps (with 8 rollouts per prompt) against TA-GRPO trained for 177 steps (with 32 rollouts per prompt group), ensuring both methods consume identical total rollouts ($\sim$725K). We further introduce an ablation condition that uses transformations but computes advantages per-variant rather than pooled, isolating the contribution of data augmentation from pooled normalization.

Our contributions are as follows:

[1] https://gitlab.com/fars-a/compute-matched-ta-grpo

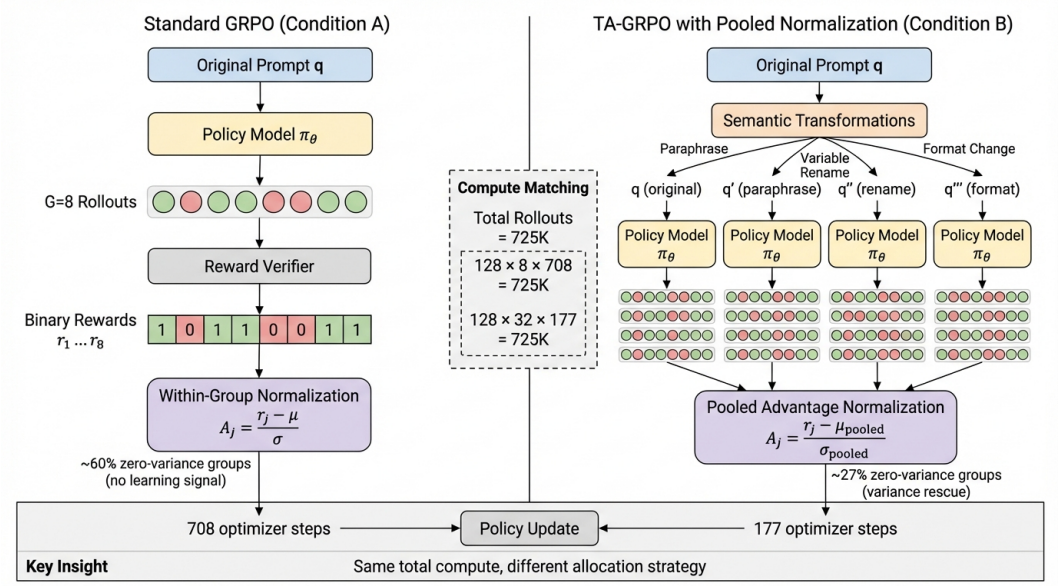

Figure 1: Compute-matched comparison of Standard GRPO (left) vs TA-GRPO (right). Both methods use identical rollout budgets (∼725K rollouts). Standard GRPO trains for 708 steps with 8 rollouts per prompt. TA-GRPO applies semantic transformations (variable renaming, paraphrasing, format changes) to create 4 variants per prompt, trains for 177 steps with 32 rollouts per prompt group, and uses pooled advantage normalization across variants.

- We present the first compute-matched evaluation of TA-GRPO, demonstrating that it achieves +2.02pp higher Pass@32 than GRPO-Long under identical rollout budgets, confirming that semantic transformations provide genuine benefits beyond additional compute.
- Through ablation analysis, we show that 87% of TA-GRPO's improvement comes from semantic data augmentation, while only 13% comes from pooled advantage normalization.
- We analyze Pass@k scaling and variance rescue, finding that TA-GRPO's advantage grows with inference-time compute (from +1.07pp at $k = 1$ to +2.02pp at $k = 32$), consistent with improved solution diversity.

2 Method

This section describes our experimental design for evaluating Transform-Augmented GRPO (TA-GRPO) under compute-matched conditions. The central question is whether TA-GRPO's reported improvements over standard GRPO stem from its semantic transformation mechanism or simply from using more rollouts per training step.

2.1 Compute-Matching Strategy

TA-GRPO (Le et al., 2026) generates N semantic transformations (paraphrases, variable renamings, format changes) for each training prompt and samples G rollouts per variant, yielding $(N + 1) \times G$ rollouts per original prompt. In the original paper, this results in $4\times$ more rollouts than standard GRPO, confounding the comparison. To isolate the effect of semantic transformations from additional compute, we match total rollouts across conditions:

$$\text{Total Rollouts} = B \times G_{\text{eff}} \times S \tag{1}$$

where B is batch size, G_{eff} is effective rollouts per prompt, and S is training steps. For standard GRPO, $G_{\text{eff}} = G$; for TA-GRPO, $G_{\text{eff}} = (N + 1) \times G$.

Table 1: Compute-matched comparison of GRPO variants on mathematical reasoning benchmarks (Pass@32). Best results in **bold**. All methods use ∼725K total rollouts.

Method	AMC12	AIME24	AIME25	OlympiadBench	MinervaM	Mean
Base Model	80.0	13.3	30.0	53.9	50.4	45.52
GRPO-Long (A)	86.3	18.3	25.0	54.4	53.3	47.45
TA-GRPO Unpooled (C)	**87.5**	**20.0**	26.7	**57.0**	54.9	49.21
TA-GRPO Pooled (B)	82.5	18.3	**31.7**	54.8	**60.0**	**49.47**

2.2 Experimental Conditions

We design three conditions with matched total rollouts of approximately 725K, as illustrated in Figure 1:

Condition A (GRPO-Long): Standard GRPO trained for 708 steps with $G = 8$ rollouts per prompt and batch size 128. Total rollouts: $128 \times 8 \times 708 \approx 725$K. This serves as the compute-matched baseline.

Condition B (TA-GRPO Pooled): TA-GRPO with $N = 3$ transformations per prompt, $G = 8$ rollouts per variant (32 total per group), trained for 177 steps. Total rollouts: $128 \times 32 \times 177 \approx 725$K. Advantages are normalized by pooling mean and standard deviation across all 32 rollouts per group.

Condition C (TA-GRPO Unpooled): Ablation condition identical to B except advantages are computed independently within each variant's 8-rollout subgroup. This isolates the contribution of semantic transformations from pooled normalization.

All conditions use Qwen3-1.7B-Base as the base model and train on PrimeIntellect/Hendrycks-Math (6,974 prompts). Conditions B and C use optimized hyperparameters (learning rate 5×10^{-6}, KL coefficient $\beta = 0.001$) while Condition A uses the original settings (learning rate 1×10^{-6}, $\beta = 0.01$).

2.3 Evaluation Protocol

We evaluate on five mathematical reasoning benchmarks spanning competition mathematics and scientific reasoning: AMC12 (40 problems), AIME24 (30 problems), AIME25 (30 problems), OlympiadBench (He et al., 2024) (674 problems), and MinervaM (244 problems). Performance is measured using Pass@k for $k \in \{1, 8, 16, 32\}$, computed via unbiased estimation from 32 samples per problem. We report results from the best checkpoint per seed (selected by mean Pass@32 on held-out validation) and average across two random seeds for Conditions A and B.

3 Experiments

3.1 Main Results

Table 1 presents the compute-matched comparison across five mathematical reasoning benchmarks. Under identical rollout budgets (∼725K), TA-GRPO with pooled normalization achieves a mean Pass@32 of 49.47%, outperforming GRPO-Long (47.45%) by +2.02 percentage points.

The improvement is consistent across most benchmarks, with the largest gains observed on MinervaM (+6.7pp) and AIME25 (+6.7pp). Notably, both TA-GRPO variants outperform GRPO-Long, with the unpooled variant achieving 49.21% and the pooled variant reaching 49.47%. This suggests that semantic transformations provide genuine benefits beyond what additional training iterations alone can achieve.

3.2 Ablation: Transforms vs Pooling

To disentangle the contributions of semantic data augmentation from pooled advantage normalization, we compare Condition C (TA-GRPO with per-variant normalization) against both baselines. Table 2 presents the attribution analysis.

Table 2: Ablation analysis: Attribution of TA-GRPO's improvement to data augmentation vs pooled normalization. Data augmentation contributes 87% of the total gain.

Comparison	Δ Mean Pass@32 (pp)	% of Total Gain
Transforms (C−A)	+1.76	87%
Pooling (B−C)	+0.26	13%
Total (B−A)	**+2.02**	**100%**

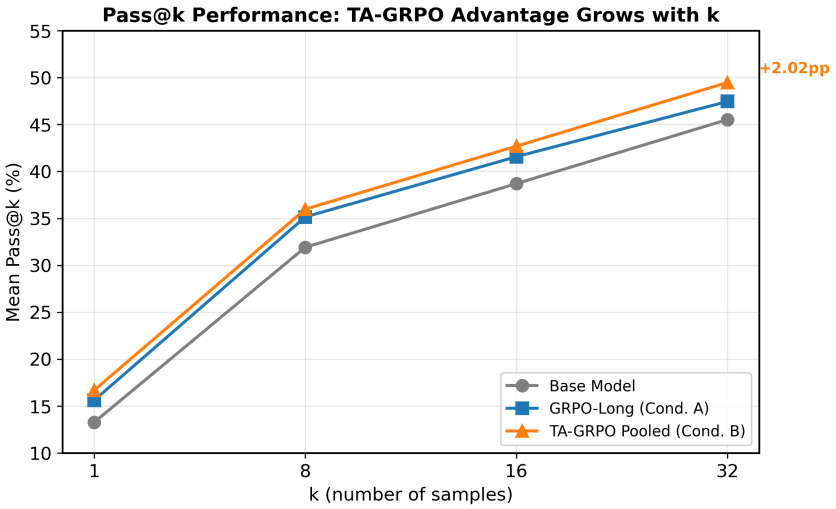

Figure 2: Pass@k performance comparison across inference-time compute levels. TA-GRPO's advantage over GRPO-Long grows from +1.07pp at $k = 1$ to +2.02pp at $k = 32$, indicating improved solution diversity rather than just greedy accuracy.

The ablation reveals that semantic data augmentation is the primary driver of TA-GRPO's improvement. Training on transformed variants alone (without pooled normalization) yields +1.76pp over GRPO-Long, accounting for 87% of the total gain. Pooled normalization contributes an additional +0.26pp (13%), providing a small but consistent benefit. This finding suggests that the core mechanism of TA-GRPO lies in exposing the model to diverse reformulations of the same problem during training, rather than in the pooled advantage computation.

3.3 Pass@k Analysis

Figure 2 examines how the advantage of TA-GRPO over GRPO-Long varies with inference-time compute (number of samples k). The gap between methods grows from +1.07pp at Pass@1 to +2.02pp at Pass@32, indicating that TA-GRPO improves solution diversity rather than just greedy accuracy.

This scaling pattern is consistent with the hypothesis that semantic transformations encourage the model to learn multiple solution strategies. When only one sample is drawn ($k = 1$), the model must rely on its most confident approach. As k increases, models that have learned diverse strategies can produce more varied correct solutions, leading to higher Pass@k. The widening gap suggests that TA-GRPO produces more diverse correct solutions than GRPO-Long trained for equivalent compute.

3.4 Variance Rescue Analysis

Pooled normalization aims to address the zero-variance problem by combining rollouts across semantically equivalent variants. Figure 3 shows the zero-variance rates across conditions.

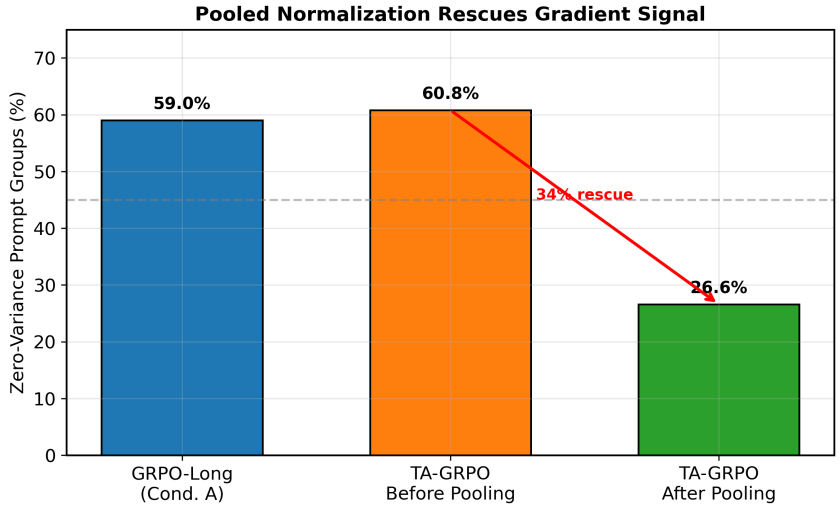

Figure 3: Zero-variance prompt group rates across conditions. Pooled normalization in TA-GRPO reduces zero-variance groups from 60.8% to 26.6%, rescuing gradient signal for 34% of prompts that would otherwise contribute nothing to learning.

Before pooling, approximately 60.8% of per-variant groups (8 rollouts each) have zero reward variance. After pooling across all 32 rollouts per prompt group, this drops to 26.6%, representing a 34% rescue rate. This mechanism enables gradient signal on prompts that would otherwise contribute nothing to learning. However, given that the unpooled variant (Condition C) achieves nearly identical performance to the pooled variant (Condition B), the practical impact of variance rescue appears limited in this setting.

4 Related Work

Reinforcement Learning for LLM Reasoning. Reinforcement learning has emerged as a key technique for improving reasoning capabilities in large language models (Zhang et al., 2025). Group Relative Policy Optimization (GRPO) (Shao et al., 2024) simplifies PPO by eliminating the critic network and computing advantages relative to group statistics, achieving strong results on mathematical reasoning benchmarks. Direct Preference Optimization (DPO) (Rafailov et al., 2023) offers an alternative approach that bypasses explicit reward modeling by directly optimizing preferences. Recent work has explored various extensions to these methods, including adaptive sampling strategies (Xiong et al., 2025) and median-centered normalization (Kim, 2026).

Zero-Variance Problem in RL. A fundamental challenge in GRPO-style training is the zero-variance problem: when all rollouts for a prompt receive identical rewards (all correct or all incorrect), the advantage estimates become zero and no gradient signal is produced (Le et al., 2025). This issue is particularly acute for mathematical reasoning, where problems may be trivially easy or impossibly hard for the current model. TA-GRPO (Le et al., 2026) addresses this by pooling advantages across semantically equivalent prompt variants, ensuring mixed rewards even when individual variants produce uniform outcomes. Our work provides the first compute-matched evaluation of this approach, isolating the contribution of semantic transformations from the additional compute they introduce.

5 Conclusion

We present a compute-matched evaluation of Transform-Augmented GRPO for mathematical reasoning. Under identical rollout budgets ($\sim$725K), TA-GRPO achieves +2.02pp higher Pass@32 than standard GRPO, demonstrating that semantic transformations provide genuine benefits beyond additional compute. Ablation analysis reveals that 87% of this improvement stems from data augmen-

tation (training on diverse problem reformulations), while only 13% comes from pooled advantage normalization. For practitioners, these findings suggest that adding semantic transforms to GRPO training is worthwhile even when compute budgets are fixed.

REFERENCES

Chaoqun He, Renjie Luo, Yuzhuo Bai, Shengding Hu, Z. Thai, Junhao Shen, Jinyi Hu, Xu Han, Yujie Huang, Yuxiang Zhang, Jie Liu, Lei Qi, Zhiyuan Liu, and Maosong Sun. Olympiadbench: A challenging benchmark for promoting agi with olympiad-level bilingual multimodal scientific problems. pp. 3828–3850, 2024.

Youngeun Kim. Mc-grpo: Median-centered group relative policy optimization for small-rollout reinforcement learning. 2026.

Khiem Le, Youssef Mroueh, Phuc Nguyen, Chi-Heng Lin, Shangqian Gao, Ting Hua, and Nitesh V. Chawla. Transform-augmented grpo improves pass@k. 2026.

Thanh-Long V. Le, Myeongho Jeon, Kim Vu, Viet Lai, and Eunho Yang. No prompt left behind: Exploiting zero-variance prompts in llm reinforcement learning via entropy-guided advantage shaping, 2025. URL https://arxiv.org/abs/2509.21880.

Rafael Rafailov, Archit Sharma, E. Mitchell, Stefano Ermon, Christopher D. Manning, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. *ArXiv*, abs/2305.18290, 2023.

Zhihong Shao, Peiyi Wang, Qihao Zhu, R. Xu, Jun-Mei Song, Mingchuan Zhang, Y. K. Li, Yu Wu, and Daya Guo. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *ArXiv*, abs/2402.03300, 2024.

Wei Xiong, Chenlu Ye, Baohao Liao, Hanze Dong, Xinxing Xu, Christof Monz, Jiang Bian, Nan Jiang, and Tong Zhang. Reinforce-ada: An adaptive sampling framework under non-linear rl objectives, 2025. URL https://arxiv.org/abs/2510.04996.

Kaiyan Zhang, Yuxin Zuo, Bingxiang He, Youbang Sun, Runze Liu, Che Jiang, Yuchen Fan, Kai Tian, Guoli Jia, Peng Li, Yu Fu, Xingtai Lv, Yuchen Zhang, Sihang Zeng, Shang Qu, Hao-Si Li, Shijie Wang, Yuru Wang, Xi-Dai Long, Fangfu Liu, Xiang Xu, Jiaze Ma, Xuekai Zhu, Ermo Hua, Yihao Liu, Zonglin Li, Hua yong Chen, Xiaoye Qu, Yafu Li, Weize Chen, Zhenzhao Yuan, Junqi Gao, Dong Li, Zhiyuan Ma, Ganqu Cui, Zhiyuan Liu, Biqing Qi, Ning Ding, and Bowen Zhou. A survey of reinforcement learning for large reasoning models. *ArXiv*, abs/2509.08827, 2025.