HARDER IS BETTER: BOOSTING MATHEMATICAL REASONING VIA DIFFICULTY-AWARE GRPO AND MULTI-ASPECT QUESTION REFORMULATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Reinforcement Learning with Verifiable Rewards (RLVR) offers a robust mechanism for enhancing the mathematical reasoning capabilities of large models. However, we identify that harder questions lack sufficient attention in existing methods from both algorithmic and data perspectives. Algorithmically, widely used Group Relative Policy Optimization (GRPO) and its variants exhibit a critical limitation: their advantage estimation introduces an implicit imbalance where the magnitude of policy updates is lower for harder questions. From a data-centric viewpoint, existing augmentation approaches primarily rephrase questions to enhance diversity, without systematically increasing their intrinsic difficulty. To address these issues, we propose a two-dual MathForge framework to improve mathematical reasoning by targeting harder questions from both perspectives, which comprises a Difficulty-Aware Group Policy Optimization (DGPO) algorithm and a Multi-Aspect Question Reformulation (MQR) strategy. Specifically, DGPO first rectifies the implicit imbalance in GRPO via difficulty-balanced group advantage estimation and further prioritizes more challenging questions by difficulty-aware question-level weighting. Meanwhile, MQR reformulates questions across multiple aspects to increase their difficulty while maintaining the original gold answer. Overall, MathForge creates a synergistic loop: MQR expands the data frontier, and DGPO efficiently masters the augmented data. Extensive experiments demonstrate that MathForge markedly outperforms existing methods on various mathematical reasoning tasks. The code and augmented data will all be available.

1 INTRODUCTION

Recently, large language models (LLMs) have demonstrated remarkable reasoning capabilities, fundamentally altering the landscape of artificial intelligence (Jaech et al., 2024; Comanici et al., 2025; Guo et al., 2025). In this context, reinforcement learning with verifiable rewards (RLVR) has been proven as a promising training paradigm (Guo et al., 2025; Wen et al., 2025), especially for enhancing mathematical reasoning. It adopts rule-based rewards instead of neural reward models, thereby significantly reducing computational overhead and mitigating the risk of reward hacking.

From an algorithmic perspective, the most representative approach to support RLVR is Group Relative Policy Optimization (GRPO) (Shao et al., 2024), which estimates relative advantages of a group of responses to the same question. However, we reveal and mathematically prove a critical limitation in GRPO and its variants: their advantage estimation function introduces an implicit imbalance where the update magnitudes are suppressed for both easier and harder questions and peak for those of moderate difficulty. The neglect of more challenging yet solvable questions is detrimental to RL training. Such questions are ideal training material, as they expose the model's incomplete mastery while also offering at least one correct response for targeted improvement. Therefore, harder questions should be emphasized to focus the model on overcoming its solvable weaknesses, while easier ones necessitate only minimal yet sufficient weighting to prevent forgetting. Zhang & Zuo (2025) also recognize the importance of question difficulty in GRPO, but their method proposes a complex difficulty-aware advantage reweighting without rectifying the underlying imbalance.

Meanwhile, from a data perspective, traditional augmentation methods for reasoning often generate entirely new question-answer pairs (Luo et al., 2023; Li et al., 2023; 2024a), but the quality of the answers is difficult to guarantee, especially for competition-level problems. As for those tailored for RLVR, only Liang et al. (2025) explore rephrasing questions while sustaining the original answer to enhance data diversity. However, the question difficulty dimension still lacks attention. Recognizing that solving mathematical reasoning problems requires varying skills, we contend that systematically increasing question difficulty by reformulating them to target and challenge these skills is a crucial approach for pushing the model's performance boundaries.

To address these issues, we introduce a comprehensive framework termed MathForge to enhance mathematical reasoning by focusing on more challenging questions from both algorithmic and data perspectives. Specifically, MathForge comprises two key components: a Difficulty-Aware Group Policy Optimization (DGPO) algorithm and a Multi-Aspect Question Reformulation (MQR) strategy. From the algorithmic perspective, DGPO first rectifies the implicit imbalance of the update magnitudes in GRPO via difficulty-balanced group advantage estimation, which normalizes group relative advantages by the mean absolute deviation of rewards rather than the standard deviation employed in GRPO. Furthermore, DGPO prioritizes more challenging questions using difficulty-aware question-level weighting, where the difficulty of a single question is quantified as the negative mean accuracy calculated across all its corresponding responses. From the data perspective, MQR reformulates the original questions across multiple aspects to increase their difficulty and diversity, including adding story background, introducing abstract terminology, and nesting sub-problems. A critical constraint is that all reformulations must preserve the original gold answer, so that MQR can maintain the essential mathematical logic of the question and obviate the need for solution regeneration. Overall, our MathForge creates a powerful synergistic loop, where MQR expands the data frontier and DGPO efficiently learns from these augmented data.

The main contributions of this paper can be summarized as follows:

1. We introduce a Difficulty-Aware Group Policy Optimization (DGPO) algorithm, which rectifies the implicit imbalance of GRPO and further upweights more challenging questions.
2. We propose Multi-Aspect Question Reformulation (MQR), a data augmentation strategy tailored for RLVR, which reformulates questions across multiple aspects to increase their difficulty while preserving the original gold answer.
3. Experiments show that our MathForge markedly outperforms existing methods on various models and mathematical reasoning benchmarks, validating its effectiveness and generalizability.

2 Preliminaries

Notation. In this paper, an autoregressive language model, parameterized by θ, is treated as a policy model, where π_θ and $\pi_{\theta_{\text{old}}}$ represent the current and old policies, respectively. For a given query q sampled from a question dataset $\mathcal{D}$, multiple responses $\{o_i\}$ are generated using the old policy $\pi_{\theta_{\text{old}}}$. A scalar reward r_i for each query-response pair (q, o_i) is then assigned by a rule-based verifier. By default, we only use the accuracy reward, 1 if the response is correct and 0 otherwise. In the context of batch processing, $\{q_s\}$ signifies a batch of queries sampled from the question dataset $\mathcal{D}$, and the corresponding responses and rewards are denoted by $\{o_{si}\}$ and $\{r_{si}\}$, respectively.

Group Relative Policy Optimization (GRPO). GRPO (Shao et al., 2024) is a variant of Proximal Policy Optimization (PPO) (Schulman et al., 2017), which eliminates the critic model, and estimates relative advantages of responses within a group of responses to the same query. Moreover, Chu et al. (2025) and Yu et al. (2025) remove the KL divergence and employ a token-level policy gradient loss to enhance the performance of GRPO. These modifications have been experimentally validated and are more commonly used in practice, becoming the default settings in TRL (von Werra et al., 2020).

Specifically, GRPO optimizes the policy model π_θ by maximizing the following objective:

$$\mathcal{J}_{\text{GRPO}}(\theta) = \mathbb{E}\left[q \sim \mathcal{D}, \{o_i\}_{i=1}^{G} \sim \pi_{\theta_{\text{old}}}(\cdot \mid q)\right]$$
$$\frac{1}{\sum_{i=1}^{G}|o_i|} \sum_{i=1}^{G} \sum_{t=1}^{|o_i|} \left\{ \min\left[I_{it}(\theta)\hat{A}_{\text{GR},i}, \text{clip}\left(I_{it}(\theta), 1-\varepsilon, 1+\varepsilon\right)\hat{A}_{\text{GR},i}\right]\right\}, \quad (1)$$

$$\text{where } I_{it}(\theta) = \frac{\pi_\theta\left(o_{i,t} \mid q, o_{i,<t}\right)}{\pi_{\theta_{\text{old}}}\left(o_{i,t} \mid q, o_{i,<t}\right)}, \ \hat{A}_{\text{GR},i} = \frac{r_i - \text{mean}\left(\{r_i\}_{i=1}^{G}\right)}{\text{std}\left(\{r_i\}_{i=1}^{G}\right)}. \quad (2)$$

Here, $I_{it}(\theta)$ denotes the importance sampling ratio of the token $o_{i,t}$, and $\hat{A}_{\text{GR},i}$ signifies the advantage of the response o_i obtained by group relative advantage estimation (GRAE). G is the number of generated responses to each query q (i.e., the group size), and ε is the clipping range of $I_{it}(\theta)$.

3 Methodology

In this section, we introduce the MathForge framework to enhance mathematical reasoning by concentrating on more challenging questions from both algorithmic and data perspectives. Specifically, it consists of two core components: the Difficulty-Aware Group Policy Optimization (DGPO) algorithm and the Multi-Aspect Question Reformulation (MQR) strategy.

3.1 Difficulty-Aware Group Policy Optimization

Although GRPO achieves strong reasoning performance, we mathematically prove that its optimization objective is unbalanced with respect to the update magnitudes for questions with varying difficulties, which primarily stems from its group relative advantage estimation (i.e., $\hat{A}_{\text{GR},i}$ in Equation 2). This imbalance potentially reduces the extent to which the policy updates for more challenging yet solvable questions. However, such questions are ideal training material that expose the model's incomplete mastery while also offering at least one correct response for targeted improvement. Moreover, harder questions may be more complex compositions or reformulations of easier ones, thus mastering harder ones can potentially enhance the model's performance on easier ones.

To resolve this issue, our Difficulty-Aware Group Policy Optimization (DGPO) algorithm first proposes difficulty-balanced group advantage estimation (DGAE) to normalize the update magnitudes across questions. Secondly, it employs difficulty-aware question-level weighting (DQW) to prioritize more challenging questions further.

Specifically, the optimization objective of DGPO is defined as follows:

$$\mathcal{J}_{\text{DGPO}}(\theta) = \mathbb{E}\left[\{q_s\}_{s=1}^{B} \sim \mathcal{D}, \{o_{si}\}_{i=1}^{G} \sim \pi_{\theta_{\text{old}}}(\cdot \mid q_s)\right]$$
$$\frac{1}{\sum_{s=1}^{B_{\text{v}}}\sum_{i=1}^{G}|o_{si}|}\sum_{s=1}^{B_{\text{v}}}\lambda_s\sum_{i=1}^{G}\sum_{t=1}^{|o_{si}|}\left\{\min\left[I_{sit}(\theta)\hat{A}_{\text{DG},si}, \text{clip}\left(I_{sit}(\theta), 1-\varepsilon, 1+\varepsilon\right)\hat{A}_{\text{DG},si}\right]\right\}, \quad (3)$$

where $I_{sit}(\theta)$ is the importance sampling ratio of the token $o_{si,t}$, and $\hat{A}_{\text{DG},si}$ is the advantage of the response o_i obtained by DGAE, respectively given by:

$$I_{sit}(\theta) = \frac{\pi_\theta\left(o_{si,t} \mid q_s, o_{si,<t}\right)}{\pi_{\theta_{\text{old}}}\left(o_{si,t} \mid q_s, o_{si,<t}\right)}, \quad \hat{A}_{\text{DG},si} = \frac{r_{si} - \text{mean}\left(\{r_{si}\}_{i=1}^{G}\right)}{\text{MAD}\left(\{r_{si}\}_{i=1}^{G}\right)}, \quad (4)$$

$$\text{where } \text{MAD}\left(\{r_{si}\}_{i=1}^{G}\right) = \frac{1}{G}\sum_{i=1}^{G}\left|r_{si} - \text{mean}\left(\{r_{si}\}_{i=1}^{G}\right)\right|. \quad (5)$$

Here, $\text{MAD}(\cdot)$ denotes the mean absolute deviation function. Furthermore, λ_s is the difficulty-aware weight for the query q_s computed by DQW as follows:

$$\lambda_s = B_{\text{v}} \cdot \frac{\exp\left(D_s/T\right)}{\sum_{s=1}^{B_{\text{v}}}\exp\left(D_s/T\right)}, \quad \text{where } D_s = -\,\text{mean}\left(\{r_{si}\}_{i=1}^{G}\right). \quad (6)$$

Here, B represents the global batch size, and B_{v} signifies the number of valid queries in the batch. A query is considered valid if its G corresponding responses are not uniformly correct or incorrect. Without loss of generality, we assume that the first B_{v} queries in the batch are valid. The token-level average loss is calculated exclusively on valid queries, a procedure we refer to as valid token-level loss averaging. This design is inspired by GPG (Chu et al., 2025) and DAPO (Yu et al., 2025) and is not a key contribution of DGPO. It aims to prevent sharp gradient fluctuations caused by inconsistent valid token ratios across batches, thereby ensuring training stability, and also serves as the basis for valid query reweighting in the following DQW.

In the following subsections, we will describe the two key techniques of DGPO: difficulty-balanced group advantage estimation and difficulty-aware question-level weighting.

3.1.1 Difficulty-Balanced Group Advantage Estimation

Consider a single question q and its corresponding responses $\{o_i\}_{i=1}^G$, the unclipped policy gradient calculated in GRPO is as follows:

$$\begin{aligned} g_{\text{GRPO}} &= \frac{1}{\sum_{i=1}^G |o_i|} \sum_{i=1}^G \sum_{t=1}^{|o_i|} \hat{A}_{\text{GR},i} \nabla_\theta I_{it}(\theta) \\ &= \frac{1}{\sum_{i=1}^G |o_i|} \sum_{i=1}^G \sum_{t=1}^{|o_i|} \operatorname{sgn}\left(\hat{A}_{\text{GR},i}\right) \left|\hat{A}_{\text{GR},i}\right| \operatorname{detach}\left(I_{it}(\theta)\right) \nabla_\theta \log\left(\pi_\theta\left(o_{i,t} \mid q, o_{i,<t}\right)\right), \end{aligned} \quad (7)$$

where $\operatorname{sgn}(\cdot)$ is the sign function and $\operatorname{detach}(\cdot)$ is the stop-gradient operator. The full derivation is provided in Appendix B.1. In this equation, $\operatorname{detach}\left(I_{it}(\theta)\right)$ and $\nabla_\theta \log\left(\pi_\theta\left(o_{i,t} \mid q, o_{i,<t}\right)\right)$ respectively represent the importance sampling ratio and likelihood gradient for each token $o_{i,t}$. Crucially, $\operatorname{sgn}(\hat{A}_{\text{GR},i})$ indicates whether the policy π_θ should be updated to increase or decrease the probability of generating the response o_i, while $|\hat{A}_{\text{GR},i}|$ determines the corresponding update magnitude. Therefore, the total update magnitude for a single question q can be upper-bounded and well-approximated by the sum of these individual magnitudes across all G responses, i.e., $\sum_{i=1}^G |\hat{A}_{\text{GR},i}|$. The complete derivation is provided in Appendix B.2. This magnitude has a closed-form expression, as formalized in the following theorem. The complete proof is provided in Appendix B.3.

Theorem 1 (Update Magnitude for a Single Question using GRAE)**.** *Given a single question q and its corresponding responses $\{o_i\}_{i=1}^G$, each query-response pair receives a binary accuracy reward $r_i \in \{0, 1\}$, and p represents the accuracy rate, i.e., the proportion for a reward of 1. Then, the total update magnitude without clipping for the single question q when using GRAE satisfies:*

$$\sum_{i=1}^G \left|\hat{A}_{\text{GR},i}\right| = \sum_{i=1}^G \left| \frac{r_i - \operatorname{mean}\left(\{r_i\}_{i=1}^G\right)}{\operatorname{std}\left(\{r_i\}_{i=1}^G\right)} \right| = 2G\sqrt{p(1-p)}, \text{ where } p = \frac{1}{G}\sum_{i=1}^G r_i. \quad (8)$$

This total update magnitude varies with respect to the accuracy rate p, reaching its maximum when $p = 0.5$ and gradually decreasing as p approaches either 0 or 1.

Theorem 1 implies that within a training batch, questions with moderate accuracy rates have a greater influence on the policy update, while easier or harder questions have a smaller impact. However, we argue that challenging questions, yet have non-zero accuracy rates, warrant greater attention. These questions are ideal for training because they identify areas of the policy model's incomplete mastery while providing at least one correct response for targeted learning. Consequently, to counteract the inherent imbalance of GRAE, we develop a novel difficulty-balanced group advantage estimation (DGAE) strategy. Specifically, the advantage function of DGAE is defined as follows:

$$\hat{A}_{\text{DG},i} = \frac{r_i - \operatorname{mean}\left(\{r_i\}_{i=1}^G\right)}{\operatorname{MAD}\left(\{r_i\}_{i=1}^G\right)}, \text{ where } \operatorname{MAD}\left(\{r_i\}_{i=1}^G\right) = \frac{1}{G}\sum_{i=1}^G \left| r_i - \operatorname{mean}\left(\{r_i\}_{i=1}^G\right)\right|. \quad (9)$$

Here, the denominator $\operatorname{MAD}(\cdot)$ is the mean absolute deviation of rewards across all G responses. This normalization ensures that the total update magnitude for a single question is a constant value, as formalized in the following theorem. The complete proof is provided in Appendix B.4.

Theorem 2 (Update Magnitude for a Single Question using DGAE)**.** *Given a single question q and its corresponding responses $\{o_i\}_{i=1}^G$, each query-response pair receives a reward r_i. Then, the total update magnitude without clipping for the single question q when using DGAE satisfies:*

$$\sum_{i=1}^G \left|\hat{A}_{\text{DG},i}\right| = \sum_{i=1}^G \left| \frac{r_i - \operatorname{mean}\left(\{r_i\}_{i=1}^G\right)}{\frac{1}{G}\sum_{i=1}^G \left| r_i - \operatorname{mean}\left(\{r_i\}_{i=1}^G\right)\right|} \right| = G. \quad (10)$$

Crucially, Theorem 2 removes the binary reward constraint ($r_i \in \{0, 1\}$) in Theorem 1, rendering it suitable for a wide array of policy optimization scenarios.

3.1.2 Difficulty-Aware Question-Level Weighting

Building upon the DGAE strategy, we further introduce a difficulty-aware question-level weighting (DQW) scheme, which explicitly prioritizes learning from more challenging questions within each training batch. Specifically, DQW assigns a weight λ_s to each question q_s as follows:

$$\lambda_s = B_{\mathrm{v}} \cdot \frac{\exp\left(D_s/T\right)}{\sum_{s=1}^{B_{\mathrm{v}}} \exp\left(D_s/T\right)}, \quad \text{where } D_s = -\operatorname{mean}\left(\{r_{si}\}_{i=1}^{G}\right). \tag{11}$$

Here, D_s is the negative mean reward across all responses of the question q_s, serving as a measure of its relative difficulty at the current training stage, and T denotes the temperature hyperparameter that controls the distribution sharpness. Compared to advantage reweighting of Zhang & Zuo (2025), DQW is simpler and has fewer hyperparameters. Moreover, it is derived based on the analysis of the implicit update magnitude imbalance in GRPO and the balanced advantages of DGAE. This two-step "balance-then-reweight" procedure offers improved interpretability and controllability.

3.2 Multi-Aspect Question Reformulation

DGPO enhances mathematical reasoning from an algorithmic perspective by optimizing the learning process on existing data. To complement this, we propose the Multi-Aspect Question Reformulation (MQR) approach as a data-centric solution, which automatically reformulates training questions by a large reasoning model to generate variants that cover more complex and comprehensive aspects. A critical constraint is that *all reformulations must preserve the original gold answer*. In this manner, MQR can maintain the essential mathematical logic of the question and obviate the need for solution regeneration, thereby placing minimal demands on the reformulator model.

Specifically, MQR adds story background, introduces abstract terminology, and nests sub-problems into the original question. The default reformulator model is OpenAI o3, while some smaller open-source models can also competently handle this task. The prompts are provided in Appendix C, and the core instructions for these strategies are as follows:

Core Instructions for Multi-Aspect Question Reformulation

1. **Background:** Add a story background that is not related to the core mathematical content of the given question, but seems to be related to the question. If the given question already has such a background, change it to a new, complexer background.
2. **Term:** Invent a new, abstract mathematical term to define a concept that is central to the given question, and restate the entire question using this term.
3. **Sub-Problem:** Convert a key numerical condition of the given question which have a definite value into an independent sub-problem. The sub-problem may belong to any branch of mathematics (e.g., algebra, geometry, number theory, combinatorics).

The newly generated questions respectively challenge the policy model's ability to: 1) identify critical mathematical information amidst noise; 2) grasp abstract mathematical concepts; and 3) perform reasoning that requires multiple steps and cross-domain knowledge. Successfully solving these more difficult questions provides a strong reinforcement signal, compelling the policy model to develop these crucial reasoning skills. Examples of each reformulation aspect are provided in Appendix D.

Overall, the MQR-augmented data, which combines the original and reformulated questions, serves as ideal training material for DGPO, rendering MathForge a synergistic loop where the data extends the model's performance boundaries, and the algorithm efficiently learns from these challenges.

4 Experiments

4.1 Experimental Setup

Models and Datasets. In the main experiments, we train the Qwen2.5-Math-7B model (Yang et al., 2024) on the MATH dataset (Hendrycks et al., 2021). To evaluate the model-agnostic effectiveness of MathForge, we conduct experiments on three other models of varying sizes and types: Qwen2.5-Math-1.5B (Yang et al., 2024), Qwen2.5-3B (Team, 2025), and DeepSeek-Math-7B (Shao et al., 2024). For cold start, DeepSeek-Math-7B is fine-tuned using 80k data sampled from NuminaMath-CoT (Li et al., 2024c). Furthermore, we apply DGPO in the multimodal domain, training Qwen2.5-VL-3B-Instruct (Bai et al., 2025) on the GEOQA-8k dataset (Chen et al., 2025).

Table 1: Comparative results of methods trained on the MATH dataset using Qwen2.5-Math-7B.

Methods	AIME24	AIME25	AMC23	MATH500	Minerva	Olympiad	Avg./Δ_{GRPO}
Base Model	12.19	4.79	35.23	48.60	15.07	16.33	22.04
GRPO	20.94	8.44	58.98	72.20	27.76	37.33	37.61
Dr.GRPO	21.04	8.23	58.59	72.05	28.58	35.89	37.40$_{-0.21}$
GPG	21.98	9.06	59.61	72.05	27.21	37.67	37.93$_{+0.32}$
DAPO	21.25	8.75	58.20	72.70	29.50	37.22	37.94$_{+0.33}$
GSPO	19.38	8.33	<u>60.16</u>	73.00	28.12	37.26	37.71$_{+0.10}$
GRPO-AD	21.56	9.48	59.06	73.25	29.14	37.07	38.26$_{+0.65}$
DGPO	23.85	10.21	**61.02**	74.25	31.07	38.33	39.79$_{+2.18}$
MQR	**25.00**	<u>11.77</u>	59.38	<u>77.85</u>	<u>31.43</u>	<u>40.81</u>	<u>41.04</u>$_{+3.43}$
MathForge	<u>24.58</u>	**12.60**	59.84	**79.95**	**33.36**	**42.67**	**42.17**$_{+4.56}$

Benchmarks. In the text-only experiments, we assess models on six commonly used mathematical reasoning benchmarks: AIME24, AIME25, AMC23, MATH500 (Hendrycks et al., 2021), Minerva (Lewkowycz et al., 2022), and Olympiad (He et al., 2024). To ensure stable results, we perform 32 runs for AIME24, AIME25, and AMC-23, and 4 runs for other benchmarks, reporting the average performance across the respective runs. For the multimodal domain, we evaluate on the GeoQA test set (Chen et al., 2021) using greedy decoding. All evaluations are conducted in a zero-shot setting.

Compared Methods. We compare our MathForge framework against several state-of-the-art methods: GRPO (Shao et al., 2024), Dr.GRPO (Liu et al., 2025a), GPG (Chu et al., 2025), DAPO (Yu et al., 2025), GSPO (Zheng et al., 2025), and GRPO-AD (Zhang & Zuo, 2025). For a fair algorithm-level comparison, we disable the resampling components in GPG and DAPO, and add the Advantage reweighting for Difficulty (AD) technique of Zhang & Zuo (2025) into the GRPO baseline as GRPO-AD. To isolate the contribution of each component in MathForge, we also evaluate DGPO and MQR separately. The MQR setting refers to training on the MQR-augmented data, including the original and MQR-generated data, using GRPO.

Implementation Details. We used 8 NVIDIA H20 GPUs to conduct all experiments. To ensure fair comparison and reproducibility, our implementation is built upon the Open-R1 codebase (Hugging Face, 2025). For the DGPO algorithm, the temperature hyperparameter T in the DQW component is set to 2.0. For the MQR strategy, the data augmentation cost is reported in Appendix E. All other implementation details are provided in Appendix F.

4.2 Main Results

Table 1 presents the comparative results of various methods trained on the MATH dataset using the Qwen2.5-Math-7B model. In the following, we will analyze the effectiveness of DGPO, MQR, and their combination, MathForge, respectively.

Effectiveness of DGPO. Our DGPO algorithm, when applied alone, elevates the average score to 39.79%, a substantial gain of 2.18% over the strong GRPO baseline (37.61%). This result validates our hypothesis that prioritizing more challenging questions through DGAE and DQW effectively enhances the RL training process. By rectifying the update magnitude imbalance of GRPO and explicitly focusing the model on its solvable weakness, DGPO fosters a more efficient and targeted optimization. Additionally, DGPO also surpasses other advanced policy optimization techniques, highlighting the superior design and efficacy of our proposed difficulty-aware mechanisms.

Effectiveness of MQR. The use of MQR in training also yields significant improvements, reaching an average score of 41.04%, which is a 3.43% increase over GRPO. This demonstrates the validity of our three question reformulation strategies. By augmenting the training data with questions that introduce narrative noise (Background), abstract concepts (Term), and nested logic (Sub-Problem), MQR creates a more challenging and diverse learning environment. This substantial performance improvement indicates the effectiveness of compelling the model to develop more robust reasoning skills by tackling these more complex reformulated questions.

Effectiveness of MathForge. The combination of DGPO and MQR in the full MathForge framework achieves the best overall performance, outperforming both individual components and reach-

Table 2: Comparative results of methods trained on the MATH dataset using varying base models.

Methods	AIME24	AIME25	AMC23	MATH500	Minerva	Olympiad	Avg./Δ_{GRPO}
Qwen2.5-Math-1.5B	6.87	3.65	30.94	34.95	8.55	21.93	17.82
+ GRPO	11.35	3.96	46.48	64.85	20.13	29.59	29.39
+ DGPO	11.25	5.73	49.84	65.45	21.14	30.85	30.71$_{+1.32}$
+ MQR	11.98	5.42	50.08	69.65	23.81	33.67	32.44$_{+3.05}$
+ MathForge	**13.23**	**7.71**	**52.34**	**70.10**	**25.74**	**33.89**	**33.84**$_{+4.45}$
Qwen2.5-3B	2.81	0.73	22.66	48.65	13.69	19.37	17.99
+ GRPO	5.31	1.56	33.28	63.35	22.89	26.41	25.47
+ DGPO	**6.98**	1.56	36.56	**65.80**	25.28	26.96	27.19$_{+1.72}$
+ MQR	5.10	1.56	39.53	65.20	25.74	29.19	27.72$_{+2.25}$
+ MathForge	5.73	**1.77**	**40.70**	65.40	**28.86**	**31.59**	**29.01**$_{+3.54}$
DeepSeek-Math-7B	0.42	0.10	13.28	31.05	9.56	9.00	10.57
+ GRPO	0.63	0.10	19.14	41.45	14.71	13.44	14.91
+ DGPO	1.98	0.42	21.02	41.85	18.93	15.00	16.53$_{+1.62}$
+ MQR	1.98	**0.83**	20.86	**44.25**	17.00	15.74	16.78$_{+1.87}$
+ MathForge	**3.12**	0.73	**21.72**	43.60	**20.68**	**16.74**	**17.77**$_{+2.86}$

Table 3: Ablation Results of DGPO trained on the MATH dataset using Qwen2.5-Math-7B.

Methods	AIME24	AIME25	AMC23	MATH500	Minerva	Olympiad	Avg./Δ_{GRPO}
GRPO	20.94	8.44	58.98	72.20	27.76	37.33	37.61
DGPO (w/o DGAE & DQW)	20.21	9.06	59.45	72.40	28.58	36.56	37.71$_{+0.10}$
DGPO (w/o DQW)	21.77	9.69	60.00	73.45	29.04	37.93	38.65$_{+1.04}$
DGPO (full)	**23.85**	**10.21**	**61.02**	**74.25**	**31.07**	**38.33**	**39.79**$_{+2.18}$
DGPO ($T = 1.0$)	23.12	9.06	59.45	74.15	30.61	37.78	39.03$_{+1.42}$
DGPO ($T = 2.0$)	**23.85**	10.21	61.02	74.25	**31.07**	**38.33**	**39.79**$_{+2.18}$
DGPO ($T = 5.0$)	22.81	**11.35**	60.55	73.80	30.42	38.26	39.53$_{+1.92}$
DGPO ($T = 10.0$)	21.35	9.79	**62.27**	**74.55**	29.96	37.67	39.27$_{+1.66}$

ing an average of 42.17%. This result highlights a powerful synergy between the data-centric and algorithmic components of our framework. MQR provides the ideal training material, diverse and challenging questions that expose the model's limitations, while DGPO capitalizes on this data by ensuring the model focuses its updates on mastering these challenges. Additionally, the performance gaps between DGPO and GRPO, as well as between MathForge and MQR, further demonstrate the robustness of DGPO under different query difficulty, as MQR makes questions harder.

Model-Agnosticism of MathForge. To substantiate the model-agnosticism of MathForge, we further compare methods across different model sizes and types, as presented in Table 2. MathForge consistently delivers the best performance on all models, and the individual components, DGPO and MQR, also robustly outperform GRPO. This highlights that the principles of MathForge are fundamental and not contingent on a specific model, underscoring its broad generalizability.

4.3 Analysis of DGPO

Ablation Studies. As shown in Table 3, we conduct ablation experiments to isolate the contribution of each component in DGPO. Specifically, the valid token-level loss averaging, DGAE, and DQW components contribute average performance improvements of 0.10%, 0.94%, and 1.14%, respectively. This highlights that DGAE effectively corrects the update magnitude imbalance of GRPO, and DQW provides a significant and complementary benefit by explicitly prioritizing more challenging questions. Additionally, we investigate the sensitivity of the temperature hyperparameter T in DQW. The results indicate that $T = 2.0$ yields the best overall performance. A lower temperature may potentially lead to an overly sharp distribution that focuses too narrowly on the hardest question in a batch, while a higher temperature flattens the weighting distribution, diminishing the prioritization effect of DQW. This confirms that $T = 2.0$ strikes an optimal balance, effectively emphasizing

Table 4: Synergistic results of DGPO with other policy optimization methods trained on the MATH dataset using Qwen2.5-Math-7B.

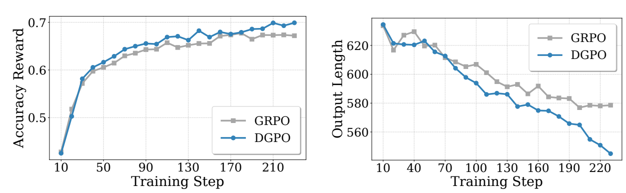
Methods	AIME24	AIME25	AMC23	MATH500	Minerva	Olympiad	Average
GPG	**21.98**	9.06	59.61	72.05	27.21	37.67	37.93
+ DGPO	21.77	**10.00**	**60.00**	**73.45**	**30.06**	**38.26**	**38.92**
DAPO	21.25	8.75	58.20	72.70	29.50	37.22	37.94
+ DGPO	**24.48**	**9.79**	**58.75**	**74.90**	**31.99**	**39.56**	**39.91**
GSPO	19.38	8.33	**60.16**	73.00	28.12	37.26	37.71
+ DGPO	**23.33**	**10.00**	59.14	**74.15**	**30.88**	**38.41**	**39.32**

Table 5: Comparative results of methods trained on the GEOQA-8k dataset using Qwen2.5-VL-3B-Instruct in the multimodal domain.

Methods	Base Model	GRPO	Dr.GRPO	GPG	DAPO	GSPO	GRPO-AD	DGPO
GeoQA/Δ_{GRPO}	39.79	57.43	57.96 +0.53	59.02 +1.59	59.02 +1.59	57.16 −0.27	58.09 +0.66	**59.95** +2.52

difficult questions while maintaining sufficient learning from the entire batch. Because the difficulty score is bounded within $(-1, 0)$, setting $T = 2.0$ ensures that the ratio between the maximum and minimum weights in a batch remains below $e^{0/T}/e^{-1/T} = e^{1/2} \approx 1.65$.

Training Dynamics. Figure 1 shows the training dynamics of DGPO versus GRPO in our main experiments, illustrating the evolution of accuracy rewards and model output lengths on MATH500. As demonstrated in Figure 1(a), DGPO consistently outperforms GRPO after the initial phase, and the performance gap widens as training progresses, underscoring that prioritizing harder questions leads to a more substantial and sustained improvement in accuracy. Meanwhile, Figure 1(b) indicates that DGPO tends to produce more concise responses, highlighting that DGPO not only improves correctness but also encourages the model to find more efficient and direct reasoning paths, trimming unnecessary verbosity and redundant steps.



(a) Accuracy Reward. (b) Output Length.

Figure 1: Training dynamics of DGPO vs. GRPO evaluated on the MATH500 benchmark. Both models are trained on MATH using Qwen2.5-Math-7B.

Compatibility with Other Methods. Our DGPO algorithm primarily introduces an improved advantage estimation and an additional question-level weighting scheme, both of which are compatible with most existing policy optimization methods. To demonstrate this, we integrate DGPO with GPG, DAPO, and GSPO, respectively. The combination forms are detailed in Appendix G. As shown in Table 4, this integration yields consistent performance improvements. Particularly, the combination of DAPO with DGPO results in even higher performance than the standalone DGPO (39.91% vs. 39.79%). This underscores that DGPO addresses a fundamental aspect of the learning process that complements the specific mechanics of other policy optimization methods. In other words, DGPO can function as a general enhancement algorithm rather than a monolithic alternative.

Applicability in the Multimodal Domain. To further verify the domain-agnosticism of DGPO, we apply it to a multimodal mathematical reasoning task. As shown in Table 5, DGPO achieves the best performance of 59.95% again, significantly higher than that of GRPO (57.43%). This demonstrates that the core principle of our DGPO, prioritizing more challenging questions, is not confined to text-only reasoning. It is a robust and generalizable algorithm for enhancing policy learning wherever a quantifiable measure of difficulty (such as accuracy rate) can be established.

4.4 Analysis of MQR

In this subsection, we normalize the total training data volume across all methods for a fair comparison. Since MQR expands the dataset by a factor of four, we achieve this by increasing the training

Table 6: Comparative results of methods trained on the original data vs. the MQR-augmented data using DGPO and varying base models.

Models	Data	AIME24	AIME25	AMC23	MATH500	Minerva	Olympiad	Average
Qwen2.5-Math-7B	Ori.	**26.46**	9.17	58.67	74.65	31.62	38.81	39.90
	MQR	24.58	**12.60**	**59.84**	**79.95**	**33.36**	**42.67**	**42.17**
Qwen2.5-Math-1.5B	Ori.	11.98	5.21	50.62	68.40	24.26	32.59	32.18
	MQR	**13.23**	**7.71**	**52.34**	**70.10**	**25.74**	**33.89**	**33.84**
Qwen2.5-3B	Ori.	**6.04**	1.35	37.66	65.05	25.28	27.93	27.22
	MQR	5.73	**1.77**	**40.70**	**65.40**	**28.86**	**31.59**	**29.01**
DeepSeek-Math-7B	Ori.	2.19	0.21	21.02	**43.60**	18.29	14.52	16.64
	MQR	**3.12**	**0.73**	**21.72**	**43.60**	**20.68**	**16.74**	**17.77**

Table 7: Ablation Results of MQR on the MATH dataset using Qwen2.5-Math-7B.

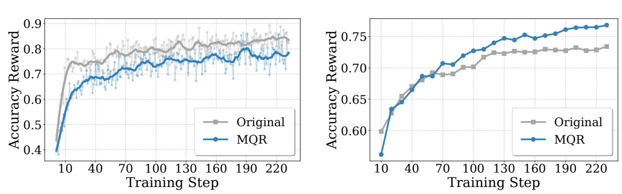
Data	AIME24	AIME25	AMC23	MATH500	Minerva	Olympiad	Avg./$\Delta_{\text{Ori.}}$
Original	26.46	9.17	58.67	74.65	31.62	38.81	39.90
MetaMath-Rephrasing	25.21	11.35	59.45	76.70	31.71	39.93	40.73$_{+0.83}$
Original + Background	25.52	10.73	58.59	77.50	32.90	40.48	40.95$_{+1.05}$
Original + Term	25.52	11.15	58.98	77.75	33.09	40.93	41.24$_{+1.34}$
Original + Sub-Problem	**26.67**	10.94	58.75	77.05	**34.38**	41.36	41.53$_{+1.63}$
MQR	24.58	**12.60**	**59.84**	**79.95**	33.36	**42.67**	**42.17**$_{+2.27}$

epochs for each method accordingly. As shown in Table 6, we compare the performance of methods trained on the original data versus the MQR-augmented data using DGPO and varying base models. MQR consistently yields superior results than the original data across all models, confirming that its effectiveness stems from the qualitative enhancement of the data, not merely an increase in volume. Additionally, we assess the quality of the generated data in Appendix H.

Difficulty Assessment. We first conduct a direct comparison of question difficulty by evaluating the accuracy of Qwen2.5-Math-7B-Instruct on the subsets of MQR-augmented data. The accuracy rates are 79.77% on Original, 77.31% on Background, 76.87% on Term, and 72.04% on Sub-Problem, confirming the increased difficulty of reformulated questions and the effectiveness of MQR.

Ablation Studies. To assess the individual contributions of our three reformulation strategies, we conduct ablation studies where each strategy is utilized separately. MetaMath-Rephrasing (Yu et al., 2024) is also included as a baseline, which uses GPT-3.5-Turbo to simply rephrase questions. We sample 22.5k data from its total 50k rephrased questions, combined with the original data for training. The results, as presented in Table 7, are all trained using DGPO. Each strategy independently improves performance over both the Original and the MetaMath-Rephrasing baselines. Crucially, the MQR approach, which combines all three strategies, achieves the highest average score of 42.17%. This underscores a clear synergy, where these diverse strategies produce a more substantial improvement than any individual component in mathematical reasoning.

Training Dynamics. Figure 2 illustrates the training dynamics of DGPO when trained on the original MATH dataset versus the MQR-augmented dataset. As presented in Figure 2(a), the consistently lower training accuracy on the MQR-augmented data exhibits that the reformulated questions are substantially more challenging. Despite this increased difficulty, the model trained with MQR ultimately achieves superior accuracy on the unseen MATH500 benchmark, as depicted in Figure 2(b). This "train harder, test better" phenomenon



(a) Training. (b) Evaluation.

Figure 2: Training dynamics of Original vs. MQR on training and evaluation data. Both models are trained on MATH and evaluated on MATH500 using Qwen2.5-Math-7B.

Table 8: Comparative results of MQR using varying reformulator models on the MATH dataset.

Reformulators	AIME24	AIME25	AMC23	MATH500	Minerva	Olympiad	Avg./$\Delta_{\text{Ori.}}$
Original	26.46	9.17	58.67	74.65	31.62	38.81	39.90
Qwen2.5-7B-Instruct	<u>25.10</u>	11.98	58.67	76.85	33.00	40.96	41.09$_{+1.19}$
Qwen3-30B-A3B-Thinking	**25.73**	<u>12.29</u>	**59.84**	<u>78.85</u>	<u>33.18</u>	<u>41.22</u>	<u>41.85</u>$_{+1.95}$
OpenAI o3	24.58	**12.60**	**59.84**	**79.95**	**33.36**	**42.67**	**42.17**$_{+2.27}$

suggests that the more challenging questions of MQR result in robust, generalizable reasoning capabilities, enhancing performance while preventing overfitting.

Generality to Less Capable Reformulators. The reformulator model is only required to reformulate questions rather than solve them, thereby imposing lower demands on its reasoning capabilities. To assess the generality of MQR to reformulator models with less capability, we utilize two smaller and open-source models: Qwen2.5-7B-Instruct (Team, 2025) and Qwen3-30B-A3B-Thinking (Yang et al., 2025). As shown in Table 8, while the most capable OpenAI o3 reformulator achieves the best results, the other two models also deliver substantial gains over the original data. This indicates that even moderately capable models can effectively generate challenging question reformulations that enhance mathematical reasoning within the MQR strategy.

5 Related Work

Reinforcement Learning. Policy optimization has become a standard for post-training large language models to enhance their reasoning capabilities (Jaech et al., 2024; Guo et al., 2025; Team et al., 2025). Building upon Proximal Policy Optimization (PPO) (Schulman et al., 2017), Group Relative Policy Optimization (GRPO) (Shao et al., 2024) proposes a highly efficient critic-less paradigm using group relative advantage estimation. This spurred a line of research focused on refining GRPO's stability and performance. For example, Dr.GRPO (Liu et al., 2025a) removes the length bias and PPO-objective bias in GRPO's advantage estimation. GPG (Chu et al., 2025), DAPO (Yu et al., 2025), and GRPO-LEAD (Zhang & Zuo, 2025) address issues in reward design, advantage estimation, and oversampling, while GSPO (Zheng et al., 2025) and GMPO (Zhao et al., 2025) introduce alternative optimization objectives. Besides, another line of work (Dai et al., 2025; Yue et al., 2025; Liu et al., 2025b) proposes more complex pipelines, such as value models or prompt refinement.

Data Augmentation. A parallel line of work improves mathematical reasoning from a data-centric perspective. One strategy involves generating entirely new, high-quality problem-solution pairs using powerful teacher models, showing that synthetic data can rival human-curated datasets (Luo et al., 2023; Li et al., 2024b;a). Another strategy, more aligned with our work, focuses on reformulating existing questions while preserving the original answer. Approaches like MetaMath (Yu et al., 2024) and PersonaMath (Luo et al., 2024) achieve this by rephrasing problems or adopting specific personas. Moreover, an advanced approach employs self-play, where the model generates its own challenging questions from solutions, fostering continuous self-improvement (Liang et al., 2025).

6 Conclusion

In this paper, we propose MathForge, a comprehensive framework designed to enhance mathematical reasoning by targeting harder problems from both algorithmic and data perspectives. MathForge is two-fold: the Difficulty-Aware Group Policy Optimization (DGPO) algorithm rectifies the update magnitude imbalance and prioritizes challenging questions, while the Multi-Aspect Question Reformulation (MQR) strategy augments training data with more difficult, yet answer-preserving, question variants from multiple aspects. Extensive experiments demonstrate that this synergistic combination significantly outperforms existing methods across various models and benchmarks, underscoring our core principle that "harder is better" in mathematical reasoning.

Ethics Statement

This work adheres to the ICLR Code of Ethics, ensuring ethical compliance throughout all stages of the research. The MQR-augmented data was constructed by reformulating problems from the public

MATH dataset. This process and the source data do not involve any personally identifiable information or sensitive content, thereby mitigating privacy concerns. The primary goal of our research is to enhance the mathematical reasoning capabilities of AI models, a pursuit with significant potential benefits for scientific research, engineering, and education.

REPRODUCIBILITY STATEMENT

To ensure the full reproducibility of our research, we will make our code and the MQR-augmented dataset publicly available. Our implementation is built upon the Open-R1 codebase (Hugging Face, 2025). Comprehensive details regarding the experimental setup, including model configurations and all hyperparameters, are described in Section 4.1 and further elaborated in Appendix F. For the MQR strategy, the exact prompts used for generating the augmented data are provided in Appendix C, and illustrative examples of the reformulated questions are presented in Appendix D.

REFERENCES

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, et al. Qwen2.5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025.

Jiaqi Chen, Jianheng Tang, Jinghui Qin, Xiaodan Liang, Lingbo Liu, Eric P Xing, and Liang Lin. Geoqa: A geometric question answering benchmark towards multimodal numerical reasoning. *arXiv preprint arXiv:2105.14517*, 2021.

Liang Chen, Lei Li, Haozhe Zhao, Yifan Song, and Vinci. R1-v: Reinforcing super generalization ability in vision-language models with less than $3. `https://github.com/Deep-Agent/R1-V`, 2025. Accessed: 2025-02-02.

Xiangxiang Chu, Hailang Huang, Xiao Zhang, Fei Wei, and Yong Wang. Gpg: A simple and strong reinforcement learning baseline for model reasoning. *arXiv preprint arXiv:2504.02546*, 2025.

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, et al. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*, 2025.

Muzhi Dai, Chenxu Yang, and Qingyi Si. S-grpo: Early exit via reinforcement learning in reasoning models. *arXiv preprint arXiv:2505.07686*, 2025.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

Chaoqun He, Renjie Luo, Yuzhuo Bai, Shengding Hu, Zhen Leng Thai, Junhao Shen, Jinyi Hu, Xu Han, Yujie Huang, Yuxiang Zhang, et al. Olympiadbench: A challenging benchmark for promoting agi with olympiad-level bilingual multimodal scientific problems. *arXiv preprint arXiv:2402.14008*, 2024.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the math dataset. *arXiv preprint arXiv:2103.03874*, 2021.

Hugging Face. Open r1: A fully open reproduction of deepseek-r1, January 2025. URL `https://github.com/huggingface/open-r1`.

Aaron Jaech, Adam Kalai, Adam Lerer, Adam Richardson, Ahmed El-Kishky, Aiden Low, Alec Helyar, Aleksander Madry, Alex Beutel, Alex Carney, et al. Openai o1 system card. *arXiv preprint arXiv:2412.16720*, 2024.

Aitor Lewkowycz, Anders Andreassen, David Dohan, Ethan Dyer, Henryk Michalewski, Vinay Ramasesh, Ambrose Slone, Cem Anil, Imanol Schlag, Theo Gutman-Solo, et al. Solving quantitative reasoning problems with language models. *Advances in neural information processing systems*, 35:3843–3857, 2022.

Chen Li, Weiqi Wang, Jingcheng Hu, Yixuan Wei, Nanning Zheng, Han Hu, Zheng Zhang, and Houwen Peng. Common 7b language models already possess strong math capabilities. *arXiv preprint arXiv:2403.04706*, 2024a.

Chengpeng Li, Zheng Yuan, Hongyi Yuan, Guanting Dong, Keming Lu, Jiancan Wu, Chuanqi Tan, Xiang Wang, and Chang Zhou. Mugglemath: Assessing the impact of query and response augmentation on math reasoning. *arXiv preprint arXiv:2310.05506*, 2023.

Chengpeng Li, Zheng Yuan, Hongyi Yuan, Guanting Dong, Keming Lu, Jiancan Wu, Chuanqi Tan, Xiang Wang, and Chang Zhou. Mugglemath: Assessing the impact of query and response augmentation on math reasoning. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 10230–10258, 2024b.

Jia Li, Edward Beeching, Lewis Tunstall, Ben Lipkin, Roman Soletskyi, Shengyi Huang, Kashif Rasul, Longhui Yu, Albert Q Jiang, Ziju Shen, et al. Numinamath: The largest public dataset in ai4maths with 860k pairs of competition math problems and solutions. *Hugging Face repository*, 13(9):9, 2024c.

Xiao Liang, Zhongzhi Li, Yeyun Gong, Yelong Shen, Ying Nian Wu, Zhijiang Guo, and Weizhu Chen. Beyond pass@ 1: Self-play with variational problem synthesis sustains rlvr. *arXiv preprint arXiv:2508.14029*, 2025.

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

Ziru Liu, Cheng Gong, Xinyu Fu, Yaofang Liu, Ran Chen, Shoubo Hu, Suiyun Zhang, Rui Liu, Qingfu Zhang, and Dandan Tu. Ghpo: Adaptive guidance for stable and efficient llm reinforcement learning. *arXiv preprint arXiv:2507.10628*, 2025b.

Haipeng Luo, Qingfeng Sun, Can Xu, Pu Zhao, Jianguang Lou, Chongyang Tao, Xiubo Geng, Qingwei Lin, Shifeng Chen, and Dongmei Zhang. Wizardmath: Empowering mathematical reasoning for large language models via reinforced evol-instruct. *arXiv preprint arXiv:2308.09583*, 2023.

Jing Luo, Longze Chen, Run Luo, Liang Zhu, Chang Ao, Jiaming Li, Yukun Chen, Xin Cheng, Wen Yang, Jiayuan Su, et al. Personamath: Boosting mathematical reasoning via persona-driven data augmentation. *arXiv preprint arXiv:2410.01504*, 2024.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Kimi Team, Yifan Bai, Yiping Bao, Guanduo Chen, Jiahao Chen, Ningxin Chen, Ruijue Chen, Yanru Chen, Yuankun Chen, Yutian Chen, et al. Kimi k2: Open agentic intelligence. *arXiv preprint arXiv:2507.20534*, 2025.

Qwen Team. Qwen2.5 technical report, 2025.

Leandro von Werra, Younes Belkada, Lewis Tunstall, Edward Beeching, Tristan Thrush, Nathan Lambert, Shengyi Huang, Kashif Rasul, and Quentin Gallouédec. Trl: Transformer reinforcement learning. https://github.com/huggingface/trl, 2020.

Xumeng Wen, Zihan Liu, Shun Zheng, Zhijian Xu, Shengyu Ye, Zhirong Wu, Xiao Liang, Yang Wang, Junjie Li, Ziming Miao, et al. Reinforcement learning with verifiable rewards implicitly incentivizes correct reasoning in base llms. *arXiv preprint arXiv:2506.14245*, 2025.

An Yang, Beichen Zhang, Binyuan Hui, Bofei Gao, Bowen Yu, Chengpeng Li, Dayiheng Liu, Jianhong Tu, Jingren Zhou, Junyang Lin, et al. Qwen2.5-math technical report: Toward mathematical expert model via self-improvement. *arXiv preprint arXiv:2409.12122*, 2024.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

Longhui Yu, Weisen Jiang, Han Shi, Jincheng Yu, Zhengying Liu, Yu Zhang, James T Kwok, Zhenguo Li, Adrian Weller, and Weiyang Liu. Metamath: Bootstrap your own mathematical questions for large language models. In *International Conference on Learning Representations*, 2024.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Lingjun Liu, et al. Dapo: An open-source llm reinforcement learning system at scale. *arXiv preprint arXiv:2503.14476*, 2025.

Yu Yue, Yufeng Yuan, Qiying Yu, Xiaochen Zuo, Ruofei Zhu, Wenyuan Xu, Jiaze Chen, Chengyi Wang, TianTian Fan, Zhengyin Du, et al. Vapo: Efficient and reliable reinforcement learning for advanced reasoning tasks. *arXiv preprint arXiv:2504.05118*, 2025.

Jixiao Zhang and Chunsheng Zuo. Grpo-lead: A difficulty-aware reinforcement learning approach for concise mathematical reasoning in language models. *arXiv preprint arXiv:2504.09696*, 2025.

Yuzhong Zhao, Yue Liu, Junpeng Liu, Jingye Chen, Xun Wu, Yaru Hao, Tengchao Lv, Shaohan Huang, Lei Cui, Qixiang Ye, et al. Geometric-mean policy optimization. *arXiv preprint arXiv:2507.20673*, 2025.

Chujie Zheng, Shixuan Liu, Mingze Li, Xiong-Hui Chen, Bowen Yu, Chang Gao, Kai Dang, Yuqiong Liu, Rui Men, An Yang, et al. Group sequence policy optimization. *arXiv preprint arXiv:2507.18071*, 2025.

A The Use of Large Language Models (LLMs)

We used LLMs to assist in polishing the writing of this paper. Its use was limited to improving grammar, clarity, and style. All core intellectual contributions, including the proposed methods, experimental design, and analysis, were conceived and executed by the human authors.

B Proofs

B.1 Full Derivation for Gradient of GRPO

Consider a single question q and its corresponding responses $\{o_i\}_{i=1}^{G}$, the unclipped policy gradient calculated in GRPO is as follows:

$$
\begin{aligned}
g_{\text{GRPO}} &= \frac{1}{\sum_{i=1}^{G}|o_i|}\sum_{i=1}^{G}\sum_{t=1}^{|o_i|}\hat{A}_{\text{GR},i}\nabla_\theta I_{it}(\theta) \\
&= \frac{1}{\sum_{i=1}^{G}|o_i|}\sum_{i=1}^{G}\sum_{t=1}^{|o_i|}\hat{A}_{\text{GR},i}\nabla_\theta \frac{\pi_\theta\left(o_{i,t}\mid q, o_{i,<t}\right)}{\pi_{\theta_{\text{old}}}\left(o_{i,t}\mid q, o_{i,<t}\right)} \\
&= \frac{1}{\sum_{i=1}^{G}|o_i|}\sum_{i=1}^{G}\sum_{t=1}^{|o_i|}\hat{A}_{\text{GR},i}\frac{\operatorname{detach}\left(\pi_\theta\left(o_{i,t}\mid q, o_{i,<t}\right)\right)}{\pi_{\theta_{\text{old}}}\left(o_{i,t}\mid q, o_{i,<t}\right)}\nabla_\theta \frac{\pi_\theta\left(o_{i,t}\mid q, o_{i,<t}\right)}{\operatorname{detach}\left(\pi_\theta\left(o_{i,t}\mid q, o_{i,<t}\right)\right)} \\
&= \frac{1}{\sum_{i=1}^{G}|o_i|}\sum_{i=1}^{G}\sum_{t=1}^{|o_i|}\hat{A}_{\text{GR},i}\operatorname{detach}\left(I_{it}(\theta)\right)\nabla_\theta \log\left(\pi_\theta\left(o_{i,t}\mid q, o_{i,<t}\right)\right) \\
&= \frac{1}{\sum_{i=1}^{G}|o_i|}\sum_{i=1}^{G}\sum_{t=1}^{|o_i|}\operatorname{sgn}\left(\hat{A}_{\text{GR},i}\right)\left|\hat{A}_{\text{GR},i}\right|\operatorname{detach}\left(I_{it}(\theta)\right)\nabla_\theta \log\left(\pi_\theta\left(o_{i,t}\mid q, o_{i,<t}\right)\right),
\end{aligned}
\tag{12}
$$

where $\mathrm{sgn}(\cdot)$ is the sign function and $\mathrm{detach}(\cdot)$ is the stop-gradient operator.

B.2 Full Derivation for the Total Update Magnitude of GRPO

The PPO/GRPO-style gradient for a fixed question q can be written (ignoring token length difference, clipping and importance sampling terms) as:

$$g(q) = \frac{1}{G}\sum_{i=1}^{G} \hat{A}_i \nabla_\theta \log \pi_\theta (o_i \mid q) \triangleq \frac{1}{G}\sum_{i=1}^{G} \hat{A}_i g_i, \tag{13}$$

By the triangle inequality, the gradient norm satisfies:

$$\|g(q)\| = \left\| \frac{1}{G}\sum_{i=1}^{G} \hat{A}_i g_i \right\| \le \frac{1}{G}\sum_{i=1}^{G} |\hat{A}_i|\|g_i\|. \tag{14}$$

Since all gradients $\hat{A}_i g_i$ are generated from the same question and tend to together improve the policy on that specific query, their directions are positively correlated. Such directional alignment implies limited mutual cancellation, causing the triangle inequality to be nearly tight.

Moreover, as all responses in a batch are sampled from the same policy with the same temperature and the same or similar math prompt, the variation in $\|g_i\|$ is typically much smaller than the variation in $|\hat{A}_i|$. Under this mild assumption, $\sum_i |\hat{A}_i|$ serves as a tight upper bound and a faithful proxy for the question-level update strength, but is not an exact equality.

B.3 Proof of Theorem 1

We provide a proof of Theorem 1 (Update Magnitude for a Single Question using GRAE) below.

Proof. By definition, the total update magnitude is the sum of the absolute values of the advantages:

$$\sum_{i=1}^{G} \left|\hat{A}_{\mathrm{GR},i}\right| = \sum_{i=1}^{G} \left| \frac{r_i - \mathrm{mean}\left(\{r_i\}_{i=1}^{G}\right)}{\mathrm{std}\left(\{r_i\}_{i=1}^{G}\right)} \right| = \frac{\sum_{i=1}^{G} \left|r_i - \mathrm{mean}\left(\{r_i\}_{i=1}^{G}\right)\right|}{\mathrm{std}\left(\{r_i\}_{i=1}^{G}\right)}. \tag{15}$$

For binary rewards $r_i \in \{0, 1\}$, the mean value is the accuracy rate $p = \frac{1}{G}\sum_{i=1}^{G} r_i$, and the standard deviation is $\sqrt{p(1-p)}$. Substituting these gives:

$$\sum_{i=1}^{G} \left|\hat{A}_{\mathrm{GR},i}\right| = \frac{\sum_{i=1}^{G} |r_i - p|}{\sqrt{p(1-p)}}. \tag{16}$$

The numerator can be decomposed based on the reward values. There are Gp terms where $r_i = 1$ and $G(1-p)$ terms where $r_i = 0$. Therefore:

$$\begin{aligned}
\sum_{i=1}^{G} \left|\hat{A}_{\mathrm{GR},i}\right| &= \frac{Gp|1-p| + G(1-p)|0-p|}{\sqrt{p(1-p)}} \\
&= \frac{Gp(1-p) + G(1-p)p}{\sqrt{p(1-p)}} \quad (\text{since } p \in (0,1)) \\
&= \frac{2Gp(1-p)}{\sqrt{p(1-p)}} \\
&= 2G\sqrt{p(1-p)}. \qquad (17)
\end{aligned}$$

$\square$

B.4 Proof of Theorem 2

We provide a proof of Theorem 2 (Update Magnitude for a Single Question using DGAE) below.

Proof. By definition, the total update magnitude is the sum of the absolute values of the advantages:

$$\sum_{i=1}^{G} \left|\hat{A}_{\mathrm{DG},i}\right| = \sum_{i=1}^{G} \left| \frac{r_i - \mathrm{mean}\left(\{r_i\}_{i=1}^{G}\right)}{\frac{1}{G}\sum_{i=1}^{G}\left|r_i - \mathrm{mean}\left(\{r_i\}_{i=1}^{G}\right)\right|} \right|. \tag{18}$$

Since the denominator, $\frac{1}{G}\sum_{j=1}^{G}\left|r_j - \mathrm{mean}\left(\{r_i\}_{i=1}^{G}\right)\right|$, is constant with respect to the summation index i and non-negative, we can move it outside the outer summation:

$$\sum_{i=1}^{G} \left|\hat{A}_{\mathrm{DG},i}\right| = \frac{\sum_{i=1}^{G}\left|r_i - \mathrm{mean}(\{r_i\}_{i=1}^{G})\right|}{\frac{1}{G}\sum_{i=1}^{G}\left|r_i - \mathrm{mean}(\{r_i\}_{i=1}^{G})\right|} = G. \tag{19}$$

□

C Prompts for MQR

We provide the detailed prompts for MQR below.

General Prompt for Question Reformulation

I want you to act as an expert Math Question Rephraser.

Your goal is to rephrase a given math question so it becomes more challenging for large AI models while remaining logically sound and fully comprehensible to humans. The rephrased question MUST yield exactly the same final answer as the original.

You should complicate the given question using the following method:
{instruction}

You must strictly adhere to the following constraints:
- The final answer MUST remain unchanged.
- The rephrased question should be no more than 100 words longer than the given question.
- Preserve the original interrogative verb (e.g., “find”, “determine”, “compute…”, “evaluate”).
- Use LaTeX for all mathematical expressions.
- Output only the rephrased question (no hints, solutions, explanation, or commentary).

#Given Question Start#
{question}
#Given Question End#

Specific Instruction for Background Question

- Add a story background that is not related to the core mathematical content of the given question, but seems to be related to the question.
- If the given question already has such a background, change it to a new, complexer background.
- Possible background themes include, but are not limited to, the following: history, culture, geography, nature, occupation, daily life, sports, art, science fiction, and adventure. Astronomy is explicitly excluded.
- The background should be presented as natural parts of the question statement, ensuring the rephrased question is coherent and self-contained.

Specific Instruction for Term Question

- Invent a new, abstract mathematical term to define a concept that is central to the given question, and restate the entire question using this term.
- The term should be presented as natural parts of the question statement, ensuring the rephrased question is coherent and self-contained.

Specific Instruction for Sub-Problem Question

- Convert a key numerical condition of the given question which have a definite value into an independent sub-problem.
- The sub-problem may belong to any branch of mathematics (e.g., algebra, geometry, number theory, combinatorics).
- The sub-problem must be self-contained, have a unique solution, and its solution must yield exactly the value required for the original question.
- The sub-problem should be presented as natural parts of the question statement, ensuring the rephrased question is coherent and self-contained.

D Augmented Data of MQR

We provide examples of questions generated by MQR below, with the highlighted parts representing the main modifications made according to the reformulation strategies.

Original Question

Berengere and her American foreign-exchange student Emily are at a bakery in Paris that accepts both euros and American dollars. They want to buy a cake, but neither of them has enough money. If the cake costs 6 euros and Emily has an American five-dollar bill, how many euros does Berengere need to contribute to the cost of the cake if 1 euro = 1.25 USD?

Question using Background Reformulation

In the bustling Montmartre district of Paris, Berengere—a culinary historian compiling notes on classic French desserts—and her visiting American friend Emily, an anthropology student documenting European food customs, wander into the venerable pâtisserie "Le Temps Sucré." They decide to purchase a famed gâteau Saint-Honoré that the proprietor has priced at 6 euros. Emily searches her travel wallet and discovers only a single crisp five-dollar bill, while Berengere carries euros exclusively. A sign by the register lists the day's exchange rate as 1 euro $= 1.25$ USD. To complete the purchase, how many euros must Berengere contribute?

Question using Term Reformulation

Define the "euro-gap" ϵ of a prospective purchase as the non-negative difference, measured in euros, between an item's listed euro price and the euro-denominated value of the funds already on hand to pay for it. Berengere and her American foreign-exchange student Emily visit a Parisian bakery. The cake they wish to buy is priced at 6 euros. Emily can contribute only an American five-dollar bill, and the prevailing conversion rate is 1 euro $= 1.25$ USD. Determine, in euros, the euro-gap ϵ that Berengere must cover to complete the purchase.

Question using Sub-Problem Reformulation

Berengere and her American foreign-exchange student Emily are at a Paris bakery that accepts both euros and U.S. dollars, but neither of them alone can pay for the desired cake. Before the exchange rate is revealed, solve this independent task: Find positive integers x

and y that satisfy

$$x + y = 9 \quad \text{and} \quad x^2 + y^2 = 41.$$

Let r be the ratio of the larger of x and y to the smaller.The cashier states that €1 is worth exactly r U.S. dollars. The cake costs €6, and Emily offers a single $5 bill. Using the exchange rate r defined above, how many euros must Berengere contribute so that together they can pay for the cake?

E DATA AUGMENTATION COST OF MQR

The average token usage per question is 255.05 input tokens, 820.27 output reasoning tokens, and 138.33 output reformulated question tokens. Therefore, the total cost for generating 22.5k reformulated questions of the MATH dataset is approximately $184.

F IMPLEMENTATION DETAILS

This section provides detailed information on the training and evaluation configurations used in our experiments.

For all reinforcement learning experiments, responses were generated with a temperature of 1.0 and a maximum completion length of 1024 tokens. During evaluation, we used a generation temperature of 0.6, a top-p value of 0.95, and set the maximum new tokens to 4096.

F.1 MATH

For experiments trained on the MATH dataset, we used the following system prompt to guide the model's reasoning process: "Please reason step by step, and put your final answer within \boxed{}." The maximum prompt length was set to 512 tokens. For each prompt, we generated 8 responses and used a training batch size of 32. The reward was based on binary accuracy, where a correct final answer yielded a reward of 1 and an incorrect one yielded 0.

Model-specific hyperparameters, including learning rate, number of epochs, gradient accumulation steps, and total training steps, are detailed in Table 9. The table specifies configurations for training on both the original 7.5k MATH dataset and the 30k MQR-augmented dataset.

Table 9: Hyperparameter settings trained on the MATH dataset using varying base models.

Models	Learning Rate	Epochs	Gradient Accumulation	Training Steps
Qwen2.5-Math-7B	5e-7	1	1	230
+MQR	1e-6	1	4	230
Qwen2.5-Math-1.5B	5e-7	1	1	230
+MQR	1e-6	1	4	230
Qwen2.5-3B	5e-7	1	1	230
+MQR	1e-6	1	4	230
DeepSeek-Math-7B	1e-6	2	1	468
+MQR	1e-6	1	1	937

For the cold start of DeepSeek-Math-7B, we sampled 80k data from NuminaMath-CoT to fine-tune it with a learning rate of $2e{-}6$, a batch size of 32, and gradient accumulation steps of 8, resulting in a total of 40 training steps.

F.2 GEOQA-8K

For the multimodal experiments on the GEOQA-8k dataset using Qwen2.5-VL-3B-Instruct, we performed a preprocessing step to remove non-standard units from the gold answers to facilitate consistent reward calculation. Consequently, the system prompt was adjusted to: “Please reason step by step, and put your final answer without units in \boxed{}.”

The training was configured with a maximum prompt length of 2048 tokens and 8 generated responses per question. The model was trained for 2 epochs using a learning rate of $1e{-}6$ and a batch size of 32. We set gradient accumulation steps to 1, resulting in a total of 480 training steps. The reward mechanism was the same binary accuracy metric used in the text-only experiments.

G Combination Forms of DGPO and Other Methods

This section details how DGPO is integrated with other policy optimization methods.

G.1 GPG

The integration with GPG involves replacing its original advantage formulation with our DGAE and incorporating the DQW scheme. Specifically, the policy gradient objective of GPG is retained, but the update for each token is now scaled by the difficulty-balanced advantage $\hat{A}_{\text{DG},si}$. Furthermore, the loss contribution of each question is modulated by the difficulty-aware weight λ_s. The normalization is also adjusted to average over valid tokens. The optimization objective is as follows:

$$\mathcal{J}_{\text{GPG+DGPO}}(\theta) = \mathbb{E}\left[\{q_s\}_{s=1}^{B} \sim \mathcal{D}, \{o_{si}\}_{i=1}^{G} \sim \pi_\theta(\cdot \mid q_s)\right]$$
$$\frac{1}{\sum_{s=1}^{B_\text{v}} \sum_{i=1}^{G} |o_{si}|} \sum_{s=1}^{B_\text{v}} \lambda_s \sum_{i=1}^{G} \sum_{t=1}^{|o_{si}|} \left[-\log \pi_\theta\left(o_{i,t} \mid q, o_{i,<t}\right) \hat{A}_{\text{DG},si}\right], \tag{20}$$

where $\hat{A}_{\text{DG},si}$ is the advantage of the response o_i obtained by DGAE given by:

$$\hat{A}_{\text{DG},si} = G \cdot \frac{r_{si} - \text{mean}\left(\{r_{si}\}_{i=1}^{G}\right)}{\sum_{i=1}^{G}\left|r_{si} - \text{mean}\left(\{r_{si}\}_{i=1}^{G}\right)\right|}, \tag{21}$$

and λ_s is the difficulty-aware weight for the query q_s computed by DQW as follows:

$$\lambda_s = B_\text{v} \cdot \frac{\exp\left(D_s/T\right)}{\sum_{s=1}^{B_\text{v}} \exp\left(D_s/T\right)}, \quad \text{where } D_s = -\,\text{mean}\left(\{r_{si}\}_{i=1}^{G}\right). \tag{22}$$

G.2 DAPO

For DAPO, the combination preserves its core PPO-style clipped objective and its use of a composite reward signal (accuracy plus length penalty, i.e., $r_{si} = r_{\text{acc},si} + r_{\text{length},si}$). We replace DAPO's original advantage estimation with our DGAE ($\hat{A}_{\text{DG},si}$), which is calculated using this composite reward. Crucially, the difficulty score D_s for our DQW scheme is computed only using the accuracy component of the reward ($r_{\text{acc},si}$). This design choice ensures that the question weighting focuses purely on the logical difficulty of the question, rather than being conflated with the verbosity of the responses. The optimization objective is as follows:

$$\mathcal{J}_{\text{DAPO+DGPO}}(\theta) = \mathbb{E}\left[\{q_s\}_{s=1}^{B} \sim \mathcal{D}, \{o_{si}\}_{i=1}^{G} \sim \pi_\theta(\cdot \mid q_s)\right]$$
$$\frac{1}{\sum_{s=1}^{B_\text{v}} \sum_{i=1}^{G} |o_{si}|} \sum_{s=1}^{B_\text{v}} \lambda_s \sum_{i=1}^{G} \sum_{t=1}^{|o_{si}|} \left\{\min\left[I_{sit}(\theta)\hat{A}_{\text{DG},si}, \text{clip}\left(I_{sit}(\theta), 1-\varepsilon_{\text{low}}, 1+\varepsilon_{\text{high}}\right)\hat{A}_{\text{DG},si}\right]\right\}, \tag{23}$$

where $I_{sit}(\theta)$ is the importance sampling ratio of the token $o_{si,t}$, and $\hat{A}_{\text{DG},si}$ is the advantage of the response o_i obtained by DGAE, respectively given by:

$$I_{sit}(\theta) = \frac{\pi_\theta\left(o_{si,t} \mid q_s, o_{si,<t}\right)}{\pi_{\theta_{\text{old}}}\left(o_{si,t} \mid q_s, o_{si,<t}\right)}, \quad \hat{A}_{\text{DG},si} = G \cdot \frac{r_{si} - \text{mean}\left(\{r_{si}\}_{i=1}^{G}\right)}{\sum_{i=1}^{G}\left|r_{si} - \text{mean}\left(\{r_{si}\}_{i=1}^{G}\right)\right|}, \tag{24}$$

and λ_s is the difficulty-aware weight for the query q_s computed by DQW as follows:

$$\lambda_s = B_\text{v} \cdot \frac{\exp\left(D_s/T\right)}{\sum_{s=1}^{B_\text{v}} \exp\left(D_s/T\right)},$$
$$\text{where } D_s = \begin{cases} -\operatorname{mean}\left(\{r_{\text{acc},si}\}_{i=1}^{G}\right) & \text{if } \operatorname{mean}\left(\{r_{\text{acc},si}\}_{i=1}^{G}\right) \neq 0 \\ -1 & \text{if } \operatorname{mean}\left(\{r_{\text{acc},si}\}_{i=1}^{G}\right) = 0 \end{cases}. \tag{25}$$

Here, B_v signifies the number of valid queries in the batch. A query is considered valid if its rewards for G corresponding responses are not completely equal. For questions where all corresponding responses are incorrect (i.e., accuracy reward is 0), no positive learning signal is available in the current question. Consequently, we deliberately set its corresponding difficulty score, D_s, to its floor value of -1. This prevents the model from dedicating excessive attention to instances that offer no constructive gradient for policy improvement.

G.3 GSPO

The integration with GSPO is performed at the sequence level, aligning with GSPO's fundamental design. GSPO's sequence-level importance sampling ratio (S_{si}) is preserved. The update for each sequence is then driven by our DGAE, $\hat{A}_{\text{DG},si}$. The question-level weighting λ_s is also applied to modulate the influence of each question on the total loss. The loss is averaged over the number of valid questions, which aligns with the sequence-level nature of both GSPO and our DGPO. The optimization objective is as follows:

$$\mathcal{J}_{\text{GSPO+DGPO}}(\theta) = \mathbb{E}\left[\{q_s\}_{s=1}^{B} \sim \mathcal{D}, \{o_{si}\}_{i=1}^{G} \sim \pi_\theta(\cdot \mid q_s)\right]$$
$$\frac{1}{B_\text{v} \cdot G} \sum_{s=1}^{B_\text{v}} \lambda_s \sum_{i=1}^{G} \left\{ \min\left[S_{si}(\theta) \hat{A}_{\text{DG},si}, \operatorname{clip}\left(S_{si}(\theta), 1-\varepsilon, 1+\varepsilon\right) \hat{A}_{\text{DG},si} \right] \right\}, \tag{26}$$

where $S_{si}(\theta)$ is the sequence-level importance sampling ratio of the response o_{si}, and $\hat{A}_{\text{DG},si}$ is the advantage of the response o_i obtained by DGAE, respectively given by:

$$S_{si}(\theta) = \left(\prod_{t=1}^{|o_{si}|} \frac{\pi_\theta\left(o_{si,t} \mid q_s, o_{si,<t}\right)}{\pi_{\theta_\text{old}}\left(o_{si,t} \mid q_s, o_{si,<t}\right)} \right)^{\frac{1}{|o_{si}|}}, \quad \hat{A}_{\text{DG},si} = G \cdot \frac{r_{si} - \operatorname{mean}\left(\{r_{si}\}_{i=1}^{G}\right)}{\sum_{i=1}^{G} \left| r_{si} - \operatorname{mean}\left(\{r_{si}\}_{i=1}^{G}\right) \right|}, \tag{27}$$

and λ_s is the difficulty-aware weight for the query q_s computed by DQW as follows:

$$\lambda_s = B_\text{v} \cdot \frac{\exp\left(D_s/T\right)}{\sum_{s=1}^{B_\text{v}} \exp\left(D_s/T\right)}, \quad \text{where } D_s = -\operatorname{mean}\left(\{r_{si}\}_{i=1}^{G}\right). \tag{28}$$

H Quality Assessment of MQR

We utilized the OpenAI o3 model to determine whether a reformulated question is mathematically equivalent to the original question. In this context, mathematical equivalence is defined as the capacity to yield the same final answer. The specific prompt used for this evaluation is as follows:

Prompt for Quality Assessment of MQR

You are an expert in mathematics and logic.

Your task is to meticulously analyze and compare two versions of a mathematical problem: an "Original Question" and a "Rewritten Question". Your primary objective is to determine if these two questions are mathematically equivalent. For the purpose of this task, "mathematically equivalent" means that both questions, when solved correctly, will yield the identical final numerical answer or symbolic solution.

Please structure your response as follows: 1. **Equivalence Verdict:** Start with a clear and unambiguous "Yes" or "No". 2. **Detailed Justification:** If they are equivalent, explain why the changes in wording, structure, or given information do not alter the underlying mathematical operations or the final result. If they are not equivalent, pinpoint the specific change in the rewritten question that alters the problem's mathematical core. Explain how this change leads to a different solution or answer.

#Original Question Start#
{question}
#Original Question End#

#Rewritten Question Start#
{rewritten_question}
#Rewritten Question End#

We randomly sampled 100 questions from each of the three categories of reformulated questions, which yielded equivalence rates of 99% for Background, 97% for Term, and 97% for Sub-Problem, respectively.

In MQR, a failed reformulation means that the resulting question becomes unsolvable or has a new answer different from the original answer. In math reasoning RLVR, the answer space is open-ended, extremely large, and requires exact canonical matching (e.g., exact integers, simplified fractions, or normalized symbolic expressions). Therefore, it is highly improbable that the policy model would reason incorrectly and happen to provide the same answer as that of the original question. Therefore, the multiple responses to the corrupted question would be uniformly incorrect (i.e., all rewards = 0). Under GRPO and its variants (including our DGPO), such questions are invalid queries yielding no update gradients, thereby providing no harmful training signals.