Stabilizing Off-policy LLM Optimization with Prefix Importance Ratio

Shiye Lei [1] **Zhihao Cheng** [1*] **Dacheng Tao** [2]

[1] ByteDance BandAI
[2] College of Computing & Data Science, Nanyang Technological University

Abstract

Reinforcement learning (RL) post-training has increasingly demonstrated strong ability to elicit reasoning behaviors in large language models (LLMs). For training efficiency, rollouts are typically generated in an off-policy manner using an older sampling policy and then used to update the current target policy. To correct the resulting discrepancy between the sampling and target policies, most existing RL objectives rely on a token-level importance sampling ratio. However, we observe unstable training dynamics in existing importance-weighted objectives under large policy lag. In this paper, we revisit LLM policy optimization under off-policy conditions and show that the exact correction for the prefix-marginal policy gradient is the cumulative prefix importance ratio. Token-level weighting does not explicitly correct for this prefix distribution shift. To stabilize LLM optimization under large off-policy drift, we propose a simple yet effective objective, *Minimum Prefix Ratio* (**MinPRO**). MinPRO replaces the unstable cumulative prefix ratio with a biased, non-cumulative surrogate based on the minimum token-level ratio observed in the preceding prefix. Extensive experiments on both dense and mixture-of-experts LLMs, across multiple mathematical reasoning benchmarks, demonstrate that MinPRO substantially improves training stability and peak performance in off-policy regimes.

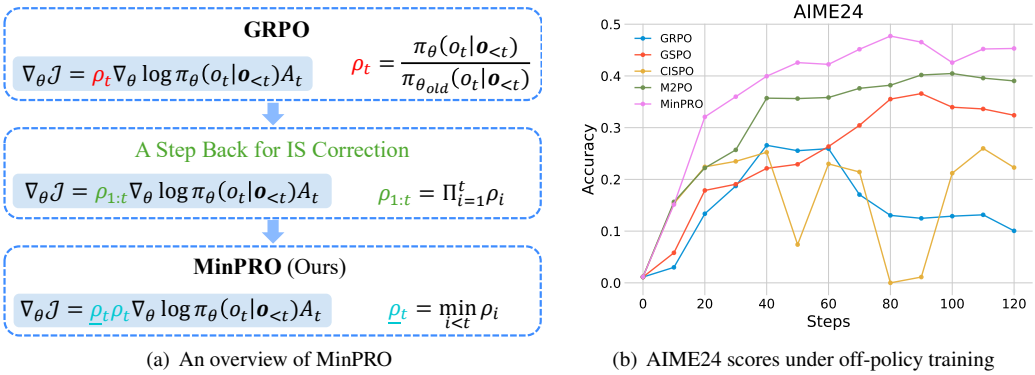
1 Introduction

Post-training has become an essential stage in modern large language model (LLM) development, complementing base model pre-training by aligning model behavior with human objectives and improving reasoning quality [Ouyang et al., 2022, Jaech et al., 2024]. In many practical scenarios, only final ground truth answers are available, and collecting detailed chain-of-thought annotations is expensive and often impractical. Reinforcement learning (RL) has therefore emerged as the most effective and widely used framework for LLM post-training [Guo et al., 2025].

When optimizing LLMs, an off-policy workflow is typically adopted: rollouts are generated by an older behavior policy π_{old}, while the optimization target is a newer policy π. This design is primarily driven by system efficiency considerations. In practice, rollouts are often generated in large batches, which helps alleviate GPU idle time caused by response-length imbalance and improves infrastructure throughput [Kimi et al., 2025, Gao et al., 2025]. These rollout batches are then divided into multiple mini-batches for gradient updates. Moreover, recent asynchronous RL frameworks further decouple rollout generation from policy gradient updates, introducing additional policy lag and amplifying off-policyness [Noukhovitch et al., 2024, Roux et al., 2025]. Although operating in a large off-policy regime can substantially improve rollout efficiency, the resulting rollout distribution shift poses severe

*Corresponding author: zhihao.cheng@bytedance.com

40th Conference on Neural Information Processing Systems (NeurIPS 2026).



(a) An overview of MinPRO

(b) AIME24 scores under off-policy training

Figure 1: (a) An overview of MinPRO. $\nabla_\theta \mathcal{J}$ denotes the policy gradient and ρ_t is the token-level importance sampling (IS) ratio. We take a step back to derive the full IS ratio $\rho_{1:t}$, referred to as the prefix importance ratio. MinPRO is then developed by relaxing $\rho_{1:t} = \rho_{1:t-1} \cdot \rho_t$ to a non-cumulative proxy $\underline{\rho_t}\rho_t$. (b) AIME24 scores as functions of training steps for Qwen3-30B-A3B-Base under off-policy training.

challenges to optimization stability and often leads to training collapse under high off-policyness [Xi et al., 2026, Zheng et al., 2026].

To correct the rollout distribution discrepancy that arises in off-policy training, a standard approach is to apply the importance sampling ratio π/π_{old}. Most LLM RL objectives instantiate this correction at the token level. However, by revisiting the policy gradient in the off-policy setting, we show that the theoretically justified correction is the prefix importance ratio, rather than the token-level ratio. Although the token-level relaxation can be effective in near-on-policy regimes, it does not accurately capture policy mismatch for LLMs, where rollouts are typically long. Consequently, this approximation can induce unstable optimization and lead to inferior performance, or even training collapse, under substantial off-policy drift.

Motivated by this insight, we seek to incorporate prefix-level information to stabilize policy optimization. In autoregressive LLMs, the prefix importance ratio is defined as a cumulative product, which is often numerically unstable and therefore impractical to use directly. To address this challenge, we introduce *Minimum Prefix Ratio* (**MinPRO**), a simple yet effective objective to stabilize LLM post-training. MinPRO replaces the cumulative prefix ratio with a simple surrogate: the current token ratio multiplied by the minimum token ratio observed in the preceding prefix (see Figure 1(a)). This formulation preserves essential prefix-level information while mitigating numerical instability and length bias by avoiding explicit cumulative products.

To comprehensively evaluate the stability of MinPRO, we conduct RL post-training under a large off-policy regime and compare it against several widely used and strong baselines, including GRPO [Shao et al., 2024, Yu et al., 2025], GSPO [Zheng et al., 2025], CISPO [Chen et al., 2025], and M2PO [Zheng et al., 2026]. As pre-trained models, we consider two dense LLMs, Qwen3-8B-Base and Qwen3-14B-Base, as well as the mixture-of-experts (MoE) model Qwen3-30B-A3B-Base [Yang et al., 2025a]. We assess performance on a suite of mathematical reasoning benchmarks, including AMC23, AIME24, AIME25, MATH500, Olympiad, Minerva, and GSM8K. From the training reward curves, which play a role analogous to training loss curves in supervised learning, MinPRO consistently exhibits higher rewards and more stable optimization dynamics throughout training. Evaluation on downstream benchmarks further shows that MinPRO achieves the highest peak performance among all compared methods, demonstrating strong stability under off-policy conditions.

2 Related Work

LLM Policy Optimization Pre-training equips LLMs with broad factual and linguistic knowledge, yet additional techniques are required to strengthen their reasoning ability. Chain-of-thought (CoT) prompting, which encourages models to decompose problems into intermediate steps, has emerged as an effective approach for improving reasoning performance [Wei et al., 2022]. More recently,

RL-based post-training has been shown to induce CoT behaviors without requiring explicit step-by-step annotations. Seminal systems such as OpenAI-o1 [Jaech et al., 2024] and DeepSeek-R1 [Guo et al., 2025] leverage RL to incentivize deliberate multi-step reasoning, leading to substantial gains on challenging tasks including mathematical problem solving and code generation. Many RL objectives have been developed from different perspectives, including clip-range design [Yu et al., 2025, Yang et al., 2025b, Sheng et al., 2025a], token ratio [Wang et al., 2025a], and token entropy [Cui et al., 2025, Wang et al., 2025b, Lei et al., 2025]. In contrast to hard clipping, which discards all tokens whose importance ratios fall outside the trust region, CISPO [Chen et al., 2025] introduces a soft-clipping mechanism that preserves all token-level gradients while constraining only the magnitude of the importance-weighting factor. In this paper, we develop an RL objective grounded in the prefix importance ratio. While GSPO [Zheng et al., 2025] replaces the token-level ratio with a full-sequence importance ratio, our approach neither discards token-level information nor relies on the complete sequence ratio. Instead, we focus on the prefix ratio, which provides a more flexible and fine-grained correction signal.

Off-Policy RL for LLMs Off-policy training, where rollouts are sampled using an older policy version and then used to update a newer policy, is common in LLM post-training. Using stale rollouts brings notable efficiency benefits. First, generating a large batch of rollouts at once reduces GPU idle time compared with repeatedly generating many smaller batches [Kimi et al., 2025, Gao et al., 2025]. Second, asynchronous execution between inference (rollout generation) and training (gradient updates) further improves overall throughput [Noukhovitch et al., 2024, Roux et al., 2025, Fu et al., 2025, Sheng et al., 2025b]. However, excessive data staleness can cause severe instability or even full training collapse. To address this challenge, several recent methods aim to stabilize training under large off-policy drift. BAPO [Xi et al., 2026] and M2PO [Zheng et al., 2026] instead modify the clipping mechanism to dynamically filter unstable tokens. AsymRE [Arnal et al., 2025] stabilizes off-policy RL by asymmetrically balancing positive and negative rewards. Its key idea is to lower the reward baseline, so that positive samples obtain larger relative advantages and therefore contribute more strongly to the policy-gradient update. In contrast, our work adopts a principled approach grounded in theoretical analysis and leverages prefix importance ratio information to mitigate instability under off-policy optimization.

3 Preliminaries

Given a question or prompt $\boldsymbol{q}$, an LLM π_θ, parameterized by θ, generates a response $\boldsymbol{o} = (o_1, o_2, \ldots, o_T)$ in an autoregressive manner, where each token o_t is sampled according to $\pi_\theta(o_t \mid \boldsymbol{q}, \boldsymbol{o}_{<t})$. The training dataset is given by $\mathcal{S} = \{(\boldsymbol{q}_i, \boldsymbol{a}_i)\}_{i=1}^m$, where $\boldsymbol{a}_i$ denotes the ground truth final answer to $\boldsymbol{q}_i$ without any intermediate reasoning steps. Although $\boldsymbol{a}_i$ does not include a full reasoning chain, it still provides a reliable supervision signal because the correctness of a generated response can be verified by comparing its final answer with $\boldsymbol{a}_i$. Many contemporary LLM RL post-training methods are built upon the PPO clip objective [Schulman et al., 2017], which applies token-level clipping to control updates according to positive and negative advantages. PPO relies on a learned critic to estimate token advantages, while recently developed critic-free algorithms such as GRPO [Shao et al., 2024] adopt multi-sample Monte Carlo estimation instead of value function learning. In this paper, we focus on the critic-free paradigm and present a detailed exposition of two representative RL objectives within this family, namely the hard-clipping method GRPO and the soft-clipping approach CISPO.

GRPO estimates token advantages by sampling multiple rollouts per prompt. Its objective is defined as

$$\mathcal{J}_{\mathrm{GRPO}}(\theta) = \mathbb{E}_{\boldsymbol{q}\sim\mathcal{S},\{\boldsymbol{o}^i\}_{i=1}^G\sim\pi_{\theta_{\mathrm{old}}}(\cdot|\boldsymbol{q})}\left[\frac{1}{G}\sum_{i=1}^{G}\frac{1}{|\boldsymbol{o}^i|}\sum_{t=1}^{|\boldsymbol{o}^i|}\min\left(\rho_t^i(\theta)\hat{A}_t^i, \mathrm{clip}\left(\rho_t^i(\theta), 1-\varepsilon, 1+\varepsilon\right)\hat{A}_t^i\right)\right],$$

where $\pi_{\theta_{\mathrm{old}}}$ is the sampling policy and $\rho_t^i(\theta) = \frac{\pi_\theta\left(o_t^i|\boldsymbol{q},\boldsymbol{o}_{<t}^i\right)}{\pi_{\theta_{\mathrm{old}}}\left(o_t^i|\boldsymbol{q},\boldsymbol{o}_{<t}^i\right)}$ denotes the importance sampling ratio. After generating G responses for each prompt $\boldsymbol{q}$, the token advantage is estimated as $\hat{A}_t^i =$

$\frac{R^i - \text{mean}\left(\{R^j\}_{j=1}^G\right)}{\text{std}\left(\{R^j\}_{j=1}^G\right)}$, where R^i is the reward assigned to the response $\boldsymbol{o}^i$ and is typically computed by comparing the predicted final answer with the ground truth.

CISPO In contrast to hard-clipping methods such as PPO and GRPO, which eliminate tokens whose importance ratios fall outside the trust region, CISPO [Chen et al., 2025] adopts a soft-clipping strategy that retains all token-level gradients and constrains only extreme ratio values, thereby controlling gradient magnitudes rather than masking gradients. The CISPO objective is given by

$$\mathcal{J}_{\text{CISPO}}(\theta) = \mathbb{E}_{q\sim\mathcal{S},\{\boldsymbol{o}^i\}_{i=1}^G\sim\pi_{\theta_{\text{old}}}(\cdot|q)} \left[\frac{1}{\sum_{i=1}^G |\boldsymbol{o}^i|} \sum_{i=1}^G \sum_{t=1}^{|\boldsymbol{o}^i|} \text{sg}\left(\text{clip}\left(\rho_t^i(\theta), 1-\varepsilon_{\text{low}}, 1+\varepsilon_{\text{high}}\right)\right) \hat{A}_t^i \log \pi_\theta\left(o_t^i \mid \boldsymbol{q}, \boldsymbol{o}_{<t}^i\right) \right],$$

where $\text{sg}(\cdot)$ denotes the stop gradient operator, ensuring that clipping affects only the gradient magnitude.

4 Method

Most LLM RL post-training methods follow the PPO formulation and therefore use the token-level importance ratio $\rho_t = \frac{\pi_\theta(o_t|\boldsymbol{o}_{<t})}{\pi_{\theta_{\text{old}}}(o_t|\boldsymbol{o}_{<t})}$, where we omit the prompt $\boldsymbol{q}$ for simplicity. In this section, *we take a step back and revisit how importance sampling ratios arise in the RL objective*, showing that the commonly used token-level ratio is an inaccurate approximation that leads to unstable optimization when training LLMs under large off-policy conditions.

4.1 Theoretical Analysis

For an LLM π_θ, recall the generated response $\boldsymbol{o} = (o_1, o_2, ..., o_T)$. Ignoring the prompt distribution for simplicity, the standard RL objective is given by the expected cumulative reward $\mathcal{J}(\theta) = \mathbb{E}_{\boldsymbol{o}\sim\pi_\theta}[R(\boldsymbol{o})]$. Then the policy gradient $\nabla_\theta \mathcal{J}(\theta)$ is obtained via the following theorem.

Lemma 1 (Policy Gradient Theorem [Sutton et al., 1999])**.** *Let $\boldsymbol{o} = (o_1, o_2, \ldots, o_T)$ denote a trajectory generated by π_θ. The gradient of the RL objective satisfies*

$$\nabla_\theta \mathcal{J}(\theta) = \sum_{t=1}^T \mathbb{E}_{(o_1,\ldots,o_t)\sim\pi_\theta}\Big[\nabla_\theta \log \pi_\theta(o_t \mid \boldsymbol{o}_{<t})\, A^\pi(o_t; \boldsymbol{o}_{<t})\Big],$$

where $\boldsymbol{o}_{<t} = (o_1, ..., o_{t-1})$ is the prefix prior to step t and $A^\pi(o_t; \boldsymbol{o}_{<t})$ is the advantage under π_θ.

We provide a proof of this lemma in Appendix A under an LLM-specific autoregressive setting. Based on Lemma 1, we can directly derive the policy gradient in the off-policy setting, where the response is generated by a separate sampling policy $\pi_{\theta_{\text{old}}}$.

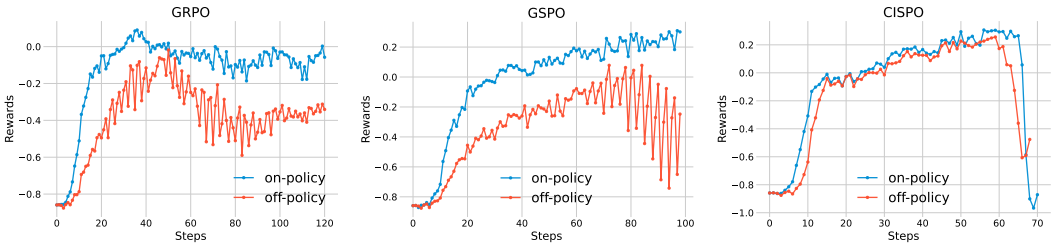
Theorem 1 (Policy Gradient under Off-policy Conditions)**.** *Suppose trajectories are sampled from an older policy $\pi_{\theta_{\text{old}}}$ and used to optimize the current policy π_θ. Let the prefix importance ratio be*

$$\rho_{1:t} = \frac{\mathbb{P}_\theta(o_1, \ldots, o_t)}{\mathbb{P}_{\theta_{\text{old}}}(o_1, \ldots, o_t)} = \prod_{i=1}^t \frac{\pi_\theta(o_i \mid \boldsymbol{o}_{<i})}{\pi_{\theta_{\text{old}}}(o_i \mid \boldsymbol{o}_{<i})} = \prod_{i=1}^t \rho_i,$$

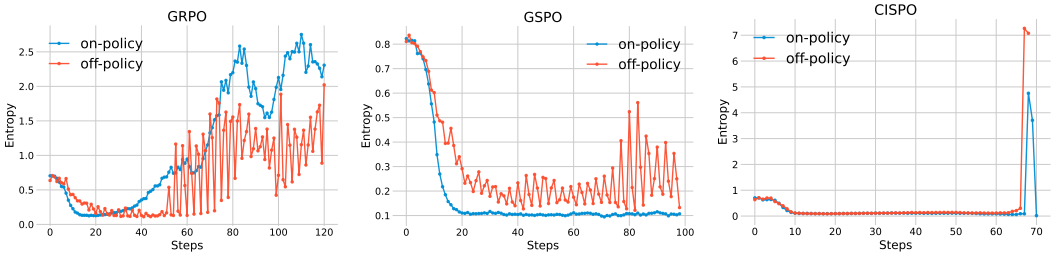
where $\mathbb{P}_\theta(o_1, ..., o_t)$ denotes the probability of generating the sequence $(o_1, ..., o_t)$ under π_θ. Then the policy gradient under off-policy sampling satisfies

$$\nabla_\theta \mathcal{J}(\theta) = \sum_{t=1}^T \mathbb{E}_{(o_1,\ldots,o_t)\sim\pi_{\theta_{\text{old}}}}\Big[\rho_{1:t}\, \nabla_\theta \log \pi_\theta(o_t \mid \boldsymbol{o}_{<t})\, A^\pi(o_t; \boldsymbol{o}_{<t})\Big].$$

As shown in Theorem 1, the prefix importance ratio $\rho_{1:t}$ provides a theoretically rigorous correction for distribution shift under off-policy regimes, whereas existing approaches rely on the relaxed token-level ratio ρ_t. To illustrate the consequences of this approximation, we compare the training dynamics



(a) Rewards vs. Steps



(b) Entropy vs. Steps

Figure 2: Plots of (a) reward and (b) entropy as functions of training steps for Qwen3-8B-Base under on-policy and off-policy regimes. In the off-policy setting, each batch of sampled rollouts is stored in a buffer and used for training after a delay of 2 global steps, whereas the on-policy setting applies no delay between sampling and optimization.

of three widely used methods, GRPO, GSPO, and CISPO, under both on-policy and off-policy settings in Figure 2. For hard-clipping methods such as GRPO and GSPO, operating in an off-policy regime results in substantially lower rewards with severe oscillations, accompanied by similarly unstable entropy dynamics, ultimately leading to degraded performance. In contrast, the soft-clipping method CISPO exhibits an even more pronounced failure mode, with training collapsing in both on-policy and off-policy settings as rewards abruptly drop and entropy explodes to extremely large values. In sum, while relaxing the prefix importance ratio $\rho_{1:t}$ to its token-level approximation ρ_t is often workable in on-policy regimes, relying on the token-level ratio leads to unstable optimization and inferior or even collapsed training under off-policy conditions.

By contrasting the relaxed token-level ratio ρ_t with the full prefix ratio $\rho_{1:t}$, we observe that in autoregressive LLM environments, where $\rho_{1:t} = \prod_{i=1}^{t} \rho_i$ grows multiplicatively with sequence length, the token-level approximation rapidly diverges from the true prefix ratio as off-policyness increases. Moreover, LLM-generated rollouts are often very long, frequently exceeding $10{,}000$ tokens, which further amplifies this divergence. This growing mismatch ultimately leads to unstable optimization dynamics and training collapse in large off-policy regimes. As a result, the commonly used token-level ratio ρ_t becomes increasingly unreliable under significant off-policy drift.

4.2 MinPRO

To achieve stable LLM optimization, a more principled treatment should explicitly incorporates the information carried by the prefix ratio $\rho_{1:t}$. A naive approach would be to use $\rho_{1:t}$ directly, possibly with heuristics such as upper clipping to prevent numerical explosion. However, since the prefix ratio is a cumulative product, it suffers from two fundamental limitations. First, it is prone to extreme values, resulting in large variance. Second, these extreme values tend to arise near the end of the generated sequence, introducing a strong length bias. This bias leads to inconsistent updates across token positions and ultimately undermines effective policy optimization.

To incorporate prefix ratio information while eliminating the large variance and length bias induced by cumulative token ratio products, we propose a simple yet stable proxy termed *Minimum Prefix Ratio* (MinPRO). Instead of using the full prefix ratio $\rho_{1:t} = \rho_{1:t-1} \cdot \rho_t$, we consider only the smallest

token ratio that appears prior to step t:

$$\underline{\rho}_t = \min_{i<t} \rho_i.$$

We then approximate the prefix ratio as $\rho_{1:t} \approx \underline{\rho}_t \cdot \rho_t$. This formulation replaces the unstable cumulative product with a non-cumulative and numerically stable surrogate that preserves essential prefix-level correction signals while mitigating variance and sequence-length-induced artifacts. We therefore formulate the MinPRO objective as follows:

$$\mathcal{J}_{\text{MinPRO}}(\theta) = \mathbb{E}_{q\sim\mathcal{S},\{\boldsymbol{o}^i\}_{i=1}^{G}\sim\pi_{\theta_{\text{old}}}(\cdot|q)} \left[\frac{1}{\sum_{i=1}^{G}|\boldsymbol{o}^i|} \sum_{i=1}^{G}\sum_{t=1}^{|\boldsymbol{o}^i|} \text{sg}\left(\text{clip}\left(\underline{\rho}_t^i \rho_t^i, 1-\varepsilon_{\text{low}}, 1+\varepsilon_{\text{high}}\right)\right) \hat{A}_t^i \log \pi_\theta\left(o_t^i \mid \boldsymbol{q}, \boldsymbol{o}_{<t}^i\right) \right]$$

As shown in the MinPRO formulation, the key modification is to multiply the token ratio ρ_t by the minimum prefix ratio $\underline{\rho}_t$. Intuitively, when the prefix ratio $\rho_{1:t-1}$ becomes extremely small, it indicates that the prefix fragment is unlikely to be sampled under the current policy. In this case, the gradient associated with token o_t should not exert a strong influence on policy optimization, even if the current token ratio ρ_t remains within a normal range. However, prior approaches ignore prefix-level information and still apply this gradient, which can lead to unstable updates under large off-policy drift. MinPRO alleviates this issue by incorporating the simple yet effective factor $\underline{\rho}_t$, which suppresses the contribution of such tokens when the prefix suggests that the trajectory is unlikely under the current policy, thereby stabilizing optimization in off-policy regimes.

5 Experiments

Datasets and Models We perform RL post-training on the DAPO-Math-17K dataset [Yu et al., 2025], which consists of 17,000 mathematical questions, each paired with a final integer answer. As base models, we use two dense LLMs, Qwen3-8B-Base and Qwen3-14B-Base, as well as an MoE model, Qwen3-30B-A3B-Base [Yang et al., 2025a]. These models are widely adopted pre-trained LLMs that have not undergone instruction tuning or reasoning-specific training, making them well-suited testbeds for evaluating post-training algorithms designed to elicit reasoning capabilities. During post-training, we evaluate mathematical reasoning performance on seven benchmarks of AMC23, AIME24, AIME25, MATH500, Olympiad, Minerva, and GSM8K.

Setup We conduct LLM RL post-training using the VeRL framework [Sheng et al., 2025c] and evaluate performance with `pass@k`, which measures the probability of obtaining at least one correct response among k sampled attempts. We adopt a critic-free setup, computing token-level advantages from normalized rollout rewards. The maximum response length is 20,480 tokens, and KL regularization is disabled. For Qwen3-8B-Base and Qwen3-30B-A3B-Base, we use a batch size of 512 and a mini-batch size of 32. For Qwen3-14B-Base, we reduce these sizes to 256 and 16, respectively, to avoid out-of-memory issues. Both configurations yield 16 parameter update steps per global training step. Qwen3-8B-Base, Qwen3-14B-Base, and Qwen3-30B-A3B-Base are trained for 120, 160, and 120 global steps, respectively, which is sufficient for convergence. Checkpoints are saved every 10 global steps, and we select the checkpoint with the highest average score across the evaluation benchmarks. Additional implementation details are provided in Appendix B.

To evaluate training stability, we explicitly introduce a large off-policy regime by controlling policy lag through a rollout buffer. Specifically, each batch of sampled rollouts is stored in the buffer and used for training after a delay of n global steps. Under this setup, the number of parameter updates separating the behavior policy $\pi_{\theta_{\text{old}}}$ and the current policy π_θ ranges from $16n$ to $16(n+1)-1$. In our experiments, we set the data staleness to $n = 2$ unless otherwise specified, corresponding to a highly off-policy regime that induces unstable optimization for many existing algorithms.

Baselines We evaluate several representative baseline methods along with their corresponding hyperparameters: (1) GRPO [Shao et al., 2024, Yu et al., 2025]: we set $\epsilon_{\text{low}} = 0.2$ and $\epsilon_{\text{high}} = 0.28$; (2) GSPO [Zheng et al., 2025]: we set $\epsilon_{\text{low}} = \epsilon_{\text{high}} = 2e-3$; (3) CISPO [Chen et al., 2025]: we use $\epsilon_{\text{low}} = 1$ and $\epsilon_{\text{high}} = 4$; (4) M2PO [Zheng et al., 2026]: a GRPO-style hard-clipping method designed for large off-policy regimes, with budget $M_2 = 0.04$ following the original paper; (5) MinPRO: our proposed method, which adopts the same settings $\epsilon_{\text{low}} = 1$ and $\epsilon_{\text{high}} = 4$ as CISPO.

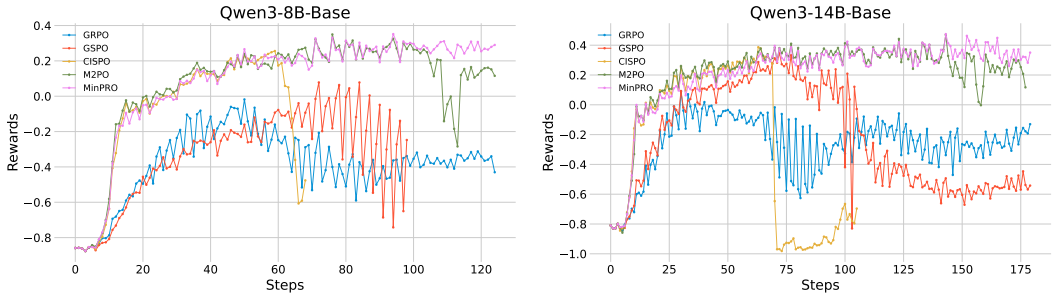


Figure 3: Training reward curves as functions of training steps for Qwen3-8B-Base and Qwen3-14B-Base under off-policy optimization.

Table 1: `pass@1` scores under large off-policyness ($n = 2$).

	Method	AMC23	AIME24	AIME25	MATH500	Olympiad	Minerva	GSM8K	Avg
8B-Base	Base	21.6	3.5	3.5	44.5	20.2	20.6	70.3	26.3
	GRPO	63.3	23.1	18.5	70.2	40.4	29.0	80.4	46.4
	GSPO	64.4	25.9	18.3	78.4	44.4	31.4	87.3	50.0
	CISPO	78.9	35.9	25.1	84.5	54.3	32.2	86.7	56.8
	M2PO	82.2	33.9	25.8	85.8	**56.5**	**37.7**	88.7	58.7
	MinPRO	**82.5**	**36.1**	**30.3**	**86.5**	**56.5**	33.6	**90.2**	**59.4**
14B-Base	Base	22.9	2.8	2.9	68.8	35.6	24.6	68.4	32.3
	GRPO	67.0	26.2	23.1	80.0	47.8	31.1	86.2	51.6
	GSPO	80.8	44.1	31.8	85.4	54.6	33.5	**92.4**	60.4
	CISPO	85.7	45.3	32.7	87.4	58.0	**35.7**	87.0	61.7
	M2PO	85.7	46.3	32.0	87.7	57.3	33.8	91.8	62.1
	MinPRO	**87.5**	**47.0**	**33.4**	**88.1**	**58.8**	33.3	90.1	**62.6**

5.1 Main Results

Optimization Stability To show the training stability of different LLM RL algorithms, we plot their training reward curves, where the reward at each step is computed as the average reward over the training batch. This quantity plays a role analogous to the (negative) training loss in supervised learning, and a steadily increasing reward curve therefore indicates a stable optimization process. As shown in Figure 3, we report training reward curves under the large off-policy regime for both 8B and 14B LLMs. In this setting, GRPO, GSPO, and CISPO exhibit highly unstable reward dynamics, while MinPRO maintains consistently higher and more stable rewards throughout training than all other baselines, including the recent M2PO method. This suggests that MinPRO can support stable long-horizon training even under severe off-policy conditions.

Pass@1 Results For AMC23, AIME24, and AIME25, `pass@1` is estimated using 128 sampled generations per prompt, while the other four larger benchmarks are evaluated with 2 generations per prompt. The results are summarized in Table 1. As shown, GRPO suffers from inferior performance under large off-policy drift, likely due to unstable training dynamics. In contrast, MinPRO achieves the highest average `pass@1` scores, outperforming all baselines by at least 0.5 points on both the 8B and 14B models. These results demonstrate the effectiveness of MinPRO for off-policy optimization.

We further evaluate MinPRO under different degrees of off-policyness, including a more off-policy setting ($n = 4$) and the near-on-policy setting ($n = 0$); results are reported in Table 2. When $n = 4$, MinPRO again achieves the best average performance, outperforming the strong off-policy baseline M2PO by 1.4 points. Moreover, when $n = 0$, MinPRO also attains the best average `pass@1` score, indicating that it remains highly competitive in the near-on-policy regime without sacrificing performance relative to standard on-policy-style optimization. Overall, these results suggest that MinPRO is effective across a broad range of off-policy regimes.

Table 2: pass@1 on Qwen3-8B-Base under more off-policy ($n = 4$) and near-on-policy ($n = 0$).

	Method	AMC23	AIME24	AIME25	MATH500	Olympiad	Minerva	GSM8K	Avg
$n = 4$	GRPO	60.5	19.1	15.2	74.4	43.6	34.2	82.2	47.0
	GSPO	60.8	20.7	15.8	73.6	44.2	34.2	88.2	48.2
	CISPO	78.6	34.3	25.4	84.3	53.0	33.0	87.3	56.6
	M2PO	77.9	34.8	24.5	83.8	54.0	**36.0**	87.6	56.9
	MinPRO	**79.3**	**38.2**	**28.3**	**84.8**	**54.2**	35.3	**88.3**	**58.3**
$n = 0$	GRPO	80.7	33.1	26.2	84.8	53.8	**36.0**	90.4	57.8
	GSPO	82.4	**37.9**	29.0	**86.3**	53.8	35.3	89.7	59.2
	CISPO	**82.5**	36.1	27.4	85.5	**55.3**	34.9	88.9	58.7
	M2PO	81.2	37.8	25.4	85.3	53.4	**36.0**	**90.6**	58.5
	MinPRO	**82.5**	**37.9**	**29.5**	85.5	54.4	35.1	90.2	**59.3**

Table 3: Mean-dataset pass@k scores averaged over AMC23, AIME24, and AIME25 under large off-policyness ($n = 2$). Boldface indicates the highest score within each model.

	Method	Pass@k								Avg
		1	2	4	8	16	32	64	128	
8B-Base	GRPO	35.0	43.2	51.1	57.8	63.4	67.7	71.3	73.9	58.0
	GSPO	36.2	44.0	51.4	58.0	63.5	68.0	71.2	73.5	58.2
	CISPO	46.6	54.3	60.5	65.2	69.2	72.6	**75.7**	**78.3**	65.3
	M2PO	47.3	54.4	60.7	65.7	69.5	72.8	75.6	77.5	65.4
	MinPRO	**49.6**	**56.7**	**62.5**	**67.0**	**70.5**	**73.1**	75.3	77.2	**66.5**
14B-Base	GRPO	38.8	47.3	54.7	60.7	65.7	69.6	72.8	75.2	60.6
	GSPO	52.2	61.5	**68.0**	**72.2**	75.2	77.6	79.5	81.0	70.9
	CISPO	54.6	62.0	67.0	71.2	75.0	78.3	81.2	83.4	71.6
	M2PO	54.7	61.1	65.7	69.8	73.5	76.7	79.6	82.1	70.4
	MinPRO	**56.0**	**63.0**	**68.0**	72.0	**75.5**	**78.9**	**82.4**	**85.4**	**72.6**

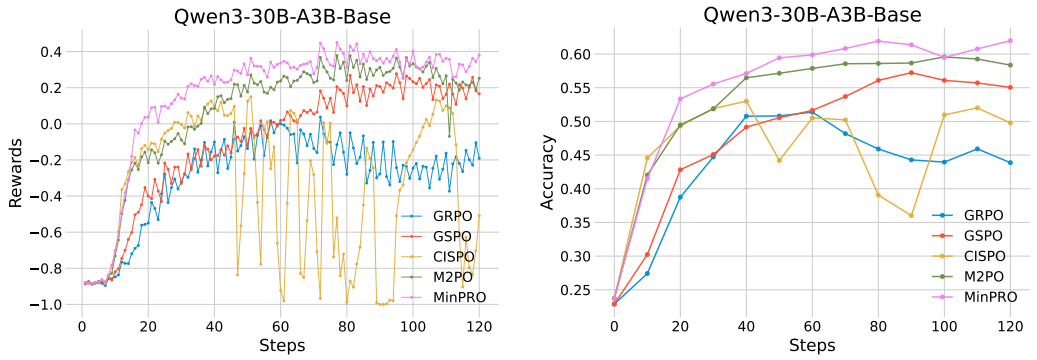
Pass@k Results To further assess performance beyond pass@1, we report average pass@k scores on AMC23, AIME24, and AIME25 in Table 3. MinPRO achieves the best overall pass@k performance on both Qwen3-8B-Base and Qwen3-14B-Base, outperforming all baselines by at least 1 point.

5.2 Scaling to Large MoE LLMs

To further assess the effectiveness of MinPRO, we additionally conduct post-training on the widely used MoE LLM Qwen3-30B-A3B-Base, which contains 30 billion total parameters while activating 3 billion parameters during inference. We also perform off-policy training by setting the data staleness to 2. The results are summarized in Figure 4. As shown in Figure 4(a), MinPRO maintains consistently higher and more stable training rewards than all other baselines throughout training, consistent with the trends observed on the 8B and 14B dense models. For evaluation, we report the averaged pass@1 accuracy across the seven benchmarks over the course of training in Figure 4(b), with detailed per-dataset curves provided in Figure 5 in Appendix C. MinPRO again achieves superior and more stable performance compared to all baseline methods. These results demonstrate that MinPRO scales effectively to large MoE models and provides stable policy optimization.

5.3 Additional Comparisons and Ablations

Choice of Prefix Statistic To assess the choice of the minimum prefix statistic $\underline{\rho}_t \rho_t$ in MinPRO, we compare it with three alternative ways of aggregating prefix ratios: the arithmetic mean $\frac{1}{t}\sum_{i=1}^{t}\rho_i$, geometric mean $(\prod_{i=1}^{t}\rho_i)^{1/t}$, and cumulative product $\prod_{i=1}^{t}\rho_i$, while keeping all other settings fixed. As shown in Table 4(a), the minimum achieves the highest average pass@1 score (58.3),



(a) Rewards vs. Steps

(b) Average `pass@1` Scores vs. Steps

Figure 4: (a) Training reward curves and (b) average `pass@1` scores as functions of training steps for Qwen3-30B-A3B-Base under off-policy optimization.

Table 4: Additional `pass@1` comparisons (%) on Qwen3-8B-Base. Avg is the mean over seven benchmarks. The shared MinPRO reference and groups (a)–(b) use $n = 4$; group (c) varies staleness. The shared reference is reported once.

Method / Statistic	AMC23	AIME24	AIME25	MATH500	Olympiad	Minerva	GSM8K	Avg
MinPRO (minimum)	79.3	38.2	28.3	84.8	54.2	35.3	88.3	**58.3**
(a) Alternative prefix statistics								
Arithmetic mean	72.4	29.8	21.9	83.0	51.6	33.5	86.1	54.0
Geometric mean	78.9	36.5	27.1	85.4	51.8	34.9	86.4	57.3
Cumulative prod	68.6	27.7	21.6	82.6	50.0	32.4	85.7	52.7
(b) Recent baselines								
AsymRE	59.5	17.3	13.3	74.6	42.6	35.3	87.8	47.2
VESPO	74.6	31.1	24.1	82.0	53.1	36.0	85.1	55.1
SNIB	65.5	24.3	19.1	76.8	45.3	28.3	84.5	49.1
(c) MinPRO at larger staleness								
MinPRO ($n = 8$)	78.9	35.6	27.1	85.0	52.6	34.6	87.0	57.3
MinPRO ($n = 16$)	78.2	35.9	27.3	83.8	52.5	34.6	88.3	57.2

outperforming all three alternatives. Here we explain that the minimum preserves a bottleneck signal: a low preceding token ratio indicates a local mismatch between the current and behavior policies. Whereas averaging can dilute this signal, the minimum retains it to downweight subsequent updates. It also avoids cumulative multiplication, which can produce extreme ratios and cause clipping saturation. These properties offer an intuitive explanation for MinPRO's empirical advantage.

More Baselines We further compare MinPRO with recent baselines, including AsymRE [Arnal et al., 2025] (see Appendix D for adaptation details), VESPO [Shen et al., 2026], and SNIB [Shao et al., 2025], which use reward-baseline adjustment, variational sequence-level weighting, and self-normalized importance sampling, respectively. As shown in Table 4(b), on Qwen3-8B-Base at $n = 4$, MinPRO achieves an average `pass@1` score of 58.3, outperforming AsymRE (47.2), VESPO (55.1), and SNIB (49.1) by 11.1, 3.2, and 9.2 points, respectively. These results further demonstrate MinPRO's performance advantage over these baselines in the off-policy setting.

Larger Staleness We also evaluate MinPRO in more off-policy settings with $n = 8$ and $n = 16$, corresponding to policy lags of 128–143 and 256–271 parameter updates, respectively. As shown in Table 4(c), MinPRO achieves average `pass@1` scores of 57.3 and 57.2, only 1.0 and 1.1 points below its score of 58.3 at $n = 4$. These results demonstrate that MinPRO retains strong performance even under substantially larger policy lags.

6 Limitations and Conclusion

Limitations While our results demonstrate the effectiveness of MinPRO for off-policy LLM training, several limitations remain. First, our training and evaluation are primarily conducted on mathematical reasoning tasks, since there are currently no widely adopted public training datasets for coding or logical reasoning in this off-policy RL setting. Second, although we evaluate MinPRO on dense 8B and 14B models as well as a 30B MoE model, we have not yet tested it on models exceeding 100B parameters. Investigating this regime is non-trivial, as RL post-training for very large LLMs requires substantially greater computational resources, potentially more than an order of magnitude beyond our current setup.

Conclusion In this paper, we address the challenge of unstable optimization when training LLMs under off-policy regimes. By revisiting the policy gradient formulation under off-policy conditions, we show that the theoretically rigorous correction term is the prefix importance ratio rather than the commonly used token-level ratio. With this insight, we propose MinPRO, a simple yet effective objective that incorporates prefix-level information through a stable surrogate of cumulative token ratios. MinPRO avoids the numerical instability and length bias of full prefix cumulative products while preserving essential correction signals. Extensive experiments on multiple dense and MoE LLMs across a range of mathematical reasoning benchmarks demonstrate that MinPRO consistently improves both optimization stability and peak performance compared to strong baselines.

Acknowledgments

Dr Tao's research is supported by the NTU Research Support Resources Award Number #023791-00001 and NTU Start Up Grant.

References

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744, 2022.

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.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. Deepseek-r1 incentivizes reasoning in llms through reinforcement learning. *Nature*, 645(8081):633–638, 2025.

Team Kimi, Angang Du, Bofei Gao, Bowei Xing, Changjiu Jiang, Cheng Chen, Cheng Li, Chenjun Xiao, Chenzhuang Du, Chonghua Liao, et al. Kimi k1. 5: Scaling reinforcement learning with llms. *arXiv preprint arXiv:2501.12599*, 2025.

Wei Gao, Yuheng Zhao, Dakai An, Tianyuan Wu, Lunxi Cao, Shaopan Xiong, Ju Huang, Weixun Wang, Siran Yang, Wenbo Su, et al. Rollpacker: Mitigating long-tail rollouts for fast, synchronous rl post-training. *arXiv preprint arXiv:2509.21009*, 2025.

Michael Noukhovitch, Shengyi Huang, Sophie Xhonneux, Arian Hosseini, Rishabh Agarwal, and Aaron Courville. Asynchronous rlhf: Faster and more efficient off-policy rl for language models. *arXiv preprint arXiv:2410.18252*, 2024.

Nicolas Le Roux, Marc G Bellemare, Jonathan Lebensold, Arnaud Bergeron, Joshua Greaves, Alex Fréchette, Carolyne Pelletier, Eric Thibodeau-Laufer, Sándor Toth, and Sam Work. Tapered off-policy reinforce: Stable and efficient reinforcement learning for llms. *arXiv preprint arXiv:2503.14286*, 2025.

Zhiheng Xi, Xin Guo, Yang Nan, Enyu Zhou, Junrui Shen, Wenxiang Chen, Jiaqi Liu, Jixuan Huang, Xun Deng, Zhihao Zhang, Honglin Guo, Zhikai Lei, Miao Zheng, Guoteng Wang, Peng

Sun, Rui Zheng, Hang Yan, Tao Gui, Qi Zhang, and Xuanjing Huang. BAPO: Stabilizing off-policy reinforcement learning for LLMs via balanced policy optimization with adaptive clipping. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=jIeJJqG7dz.

Haizhong Zheng, Jiawei Zhao, and Beidi Chen. Prosperity before collapse: How far can off-policy RL reach with stale data on LLMs? In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=IIgl5MWelz.

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

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.

Aili Chen, Aonian Li, Bangwei Gong, Binyang Jiang, Bo Fei, Bo Yang, Boji Shan, Changqing Yu, Chao Wang, Cheng Zhu, et al. Minimax-m1: Scaling test-time compute efficiently with lightning attention. *arXiv preprint arXiv:2506.13585*, 2025.

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*, 2025a.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed H. Chi, Quoc V Le, and Denny Zhou. Chain of thought prompting elicits reasoning in large language models. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho, editors, *Advances in Neural Information Processing Systems*, 2022. URL https://openreview.net/forum?id=_VjQlMeSB_J.

Shihui Yang, Chengfeng Dou, Peidong Guo, Kai Lu, Qiang Ju, Fei Deng, and Rihui Xin. Dcpo: Dynamic clipping policy optimization. *arXiv preprint arXiv:2509.02333*, 2025b.

Yuepeng Sheng, Yuwei Huang, Shuman Liu, Anxiang Zeng, and Haibo Zhang. Espo: Entropy importance sampling policy optimization. *arXiv preprint arXiv:2512.00499*, 2025a.

Jiakang Wang, Runze Liu, Lei Lin, Wenping Hu, Xiu Li, Fuzheng Zhang, Guorui Zhou, and Kun Gai. Aspo: Asymmetric importance sampling policy optimization. *arXiv preprint arXiv:2510.06062*, 2025a.

Ganqu Cui, Yuchen Zhang, Jiacheng Chen, Lifan Yuan, Zhi Wang, Yuxin Zuo, Haozhan Li, Yuchen Fan, Huayu Chen, Weize Chen, et al. The entropy mechanism of reinforcement learning for reasoning language models. *arXiv preprint arXiv:2505.22617*, 2025.

Shenzhi Wang, Le Yu, Chang Gao, Chujie Zheng, Shixuan Liu, Rui Lu, Kai Dang, Xiong-Hui Chen, Jianxin Yang, Zhenru Zhang, Yuqiong Liu, An Yang, Andrew Zhao, Yang Yue, Shiji Song, Bowen Yu, Gao Huang, and Junyang Lin. Beyond the 80/20 rule: High-entropy minority tokens drive effective reinforcement learning for LLM reasoning. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025b. URL https://openreview.net/forum?id=yfcpdY4gMP.

Shiye Lei, Zhihao Cheng, Kai Jia, and Dacheng Tao. Revisiting llm reasoning via information bottleneck. *arXiv preprint arXiv:2507.18391*, 2025.

Wei Fu, Jiaxuan Gao, Xujie Shen, Chen Zhu, Zhiyu Mei, Chuyi He, Shusheng Xu, Guo Wei, Jun Mei, WANG JIASHU, Tongkai Yang, Binhang Yuan, and Yi Wu. AREAL: A large-scale asynchronous reinforcement learning system for language reasoning. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=X9diEuva9R.

Guangming Sheng, Yuxuan Tong, Borui Wan, Wang Zhang, Chaobo Jia, Xibin Wu, Yuqi Wu, Xiang Li, Chi Zhang, Yanghua Peng, et al. Laminar: A scalable asynchronous rl post-training framework. *arXiv preprint arXiv:2510.12633*, 2025b.

Charles Arnal, Gaëtan Narozniak, Vivien Cabannes, Yunhao Tang, Julia Kempe, and Remi Munos. Asymmetric REINFORCE for off-policy reinforcement learning: Balancing positive and negative rewards. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=Ql3sENn0mi.

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

Richard S Sutton, David McAllester, Satinder Singh, and Yishay Mansour. Policy gradient methods for reinforcement learning with function approximation. *Advances in neural information processing systems*, 12, 1999.

Guangming Sheng, Chi Zhang, Zilingfeng Ye, Xibin Wu, Wang Zhang, Ru Zhang, Yanghua Peng, Haibin Lin, and Chuan Wu. Hybridflow: A flexible and efficient rlhf framework. In *Proceedings of the Twentieth European Conference on Computer Systems*, pages 1279–1297, 2025c.

Guobin Shen, Chenxiao Zhao, Xiang Cheng, Lei Huang, and Xing Yu. Vespo: Variational sequence-level soft policy optimization for stable off-policy llm training. *arXiv preprint arXiv:2602.10693*, 2026.

Huiyang Shao, Xintong Zhang, and Junyi Liu. Principled policy optimization for LLMs via self-normalized importance sampling, 2025. URL https://openreview.net/forum?id=HVciz8hi1c.

Yanli Zhao, Andrew Gu, Rohan Varma, Liang Luo, Chien-Chin Huang, Min Xu, Less Wright, Hamid Shojanazeri, Myle Ott, Sam Shleifer, Alban Desmaison, Can Balioglu, Pritam Damania, Bernard Nguyen, Geeta Chauhan, Yuchen Hao, Ajit Mathews, and Shen Li. Pytorch fsdp: Experiences on scaling fully sharded data parallel. *Proc. VLDB Endow.*, 16(12):3848–3860, August 2023. ISSN 2150-8097. doi: 10.14778/3611540.3611569. URL https://doi.org/10.14778/3611540.3611569.

Mohammad Shoeybi, Mostofa Patwary, Raul Puri, Patrick LeGresley, Jared Casper, and Bryan Catanzaro. Megatron-lm: Training multi-billion parameter language models using model parallelism. *arXiv preprint arXiv:1909.08053*, 2019.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the 29th symposium on operating systems principles*, pages 611–626, 2023.

A Proof

A.1 Proof of Lemma 1

We consider an autoregressive policy π_θ over responses (trajectories) $\boldsymbol{o} = (o_1, \ldots, o_T)$ with total return $R(\boldsymbol{o}) = \sum_{t=1}^{T} r_t$. The standard RL objective is the expected cumulative reward

$$\mathcal{J}(\theta) \;=\; \mathbb{E}_{\boldsymbol{o}\sim\pi_\theta}[R(\boldsymbol{o})].$$

Using the log–derivative trick (REINFORCE),

$$\begin{aligned}\nabla_\theta \mathcal{J} = \nabla_\theta \mathbb{E}_{\boldsymbol{o}\sim\pi_\theta}[R(\boldsymbol{o})] &= \mathbb{E}_{\boldsymbol{o}\sim\pi_\theta}\big[\nabla_\theta \log p_\theta(\boldsymbol{o})\, R(\boldsymbol{o})\big] \\ &= \mathbb{E}_{\boldsymbol{o}\sim\pi_\theta}\left[\sum_{t=1}^{T} \nabla_\theta \log \pi_\theta(o_t \mid \boldsymbol{o}_{<t})\, R(\boldsymbol{o})\right] \\ &= \sum_{t=1}^{T} \underbrace{\mathbb{E}_{\boldsymbol{o}\sim\pi_\theta}\left[\nabla_\theta \log \pi_\theta(o_t \mid \boldsymbol{o}_{<t})\, R(\boldsymbol{o})\right]}_{=:G_t}.\end{aligned}$$

Let $\boldsymbol{o}_{\le t} := (o_1, \ldots, o_t)$ denote the prefix up to step t. For a fixed t, we now rewrite G_t by conditioning on $\boldsymbol{o}_{\le t}$:

$$\begin{aligned}G_t &= \mathbb{E}_{\boldsymbol{o}_{\le t}\sim\pi_\theta}\Big[\mathbb{E}_{\boldsymbol{o}_{t+1:T}\sim\pi_\theta(\cdot|\boldsymbol{o}_{\le t})}\big[\nabla_\theta \log \pi_\theta(o_t \mid \boldsymbol{o}_{<t})\, R(\boldsymbol{o}) \,\big|\, \boldsymbol{o}_{\le t}\big]\Big] \qquad \text{(law of total expectation)} \\ &= \mathbb{E}_{\boldsymbol{o}_{\le t}\sim\pi_\theta}\Big[\nabla_\theta \log \pi_\theta(o_t \mid \boldsymbol{o}_{<t})\, \mathbb{E}_{\boldsymbol{o}_{t+1:T}\sim\pi_\theta(\cdot|\boldsymbol{o}_{\le t})}\big[R(\boldsymbol{o}) \,\big|\, \boldsymbol{o}_{\le t}\big]\Big].\end{aligned}$$

We decompose the return into past and future parts,

$$R(\boldsymbol{o}) = \underbrace{\sum_{i=1}^{t-1} r_i}_{R_{<t}(\boldsymbol{o})} + \underbrace{\sum_{i=t}^{T} r_i}_{R_{\ge t}(\boldsymbol{o})}.$$

Given $\boldsymbol{o}_{\le t}$, the past return $R_{<t}$ is deterministic, while the future return $R_{\ge t}$ is random. The state-action value is defined as

$$Q^\pi(o_t; \boldsymbol{o}_{<t}) := \mathbb{E}_{\boldsymbol{o}_{t+1:T}\sim\pi_\theta}\Big[R_{\ge t}(\boldsymbol{o}) \,\Big|\, \boldsymbol{o}_{\le t}\Big].$$

Then

$$\mathbb{E}_{\boldsymbol{o}_{t+1:T}\sim\pi_\theta(\cdot|\boldsymbol{o}_{\le t})}\big[R(\boldsymbol{o}) \mid \boldsymbol{o}_{\le t}\big] = R_{<t}(\boldsymbol{o}) + Q^\pi(o_t; \boldsymbol{o}_{<t}),$$

and hence

$$G_t = \mathbb{E}_{\boldsymbol{o}_{\le t}\sim\pi_\theta}\Big[\nabla_\theta \log \pi_\theta(o_t \mid \boldsymbol{o}_{<t})\, \big(R_{<t}(\boldsymbol{o}) + Q^\pi(o_t; \boldsymbol{o}_{<t})\big)\Big].$$

We invoke the following baseline invariance lemma to remove terms that do not depend on the action.

Lemma 2 (Score-function baseline invariance)**.** *Let $b(\boldsymbol{o}_{\le t})$ be any function that depends on $\boldsymbol{o}_{<t}$ but not on the current action o_t. Then*

$$\mathbb{E}_{\boldsymbol{o}_{\le t}\sim\pi_\theta}\Big[\nabla_\theta \log \pi_\theta(o_t \mid \boldsymbol{o}_{<t})\, b(\boldsymbol{o}_{\le t})\Big] = 0.$$

Proof. By the tower rule and the definition of conditional expectation,

$$\mathbb{E}_{\boldsymbol{o}_{\le t}\sim\pi_\theta}\big[\nabla_\theta \log \pi_\theta(o_t \mid \boldsymbol{o}_{<t})\, b(\boldsymbol{o}_{\le t})\big] = \mathbb{E}_{\boldsymbol{o}_{<t}\sim\pi_\theta}\Big[b(\boldsymbol{o}_{\le t})\, \mathbb{E}_{o_t\sim\pi_\theta(\cdot|\boldsymbol{o}_{<t})}\big[\nabla_\theta \log \pi_\theta(o_t \mid \boldsymbol{o}_{<t})\big]\Big].$$

For any fixed $\boldsymbol{o}_{<t}$, the inner expectation is the score-function identity:

$$\mathbb{E}_{o_t\sim\pi_\theta(\cdot|\boldsymbol{o}_{<t})}\big[\nabla_\theta \log \pi_\theta(o_t \mid \boldsymbol{o}_{<t})\big] = \nabla_\theta \sum_{o_t} \pi_\theta(o_t \mid \boldsymbol{o}_{<t}) = \nabla_\theta 1 = 0.$$

Thus the outer expectation is also zero, which proves the lemma. □

Applying Lemma 2 with $b(\boldsymbol{o}_{\le t}) = R_{<t}(\boldsymbol{o})$ yields

$$G_t = \mathbb{E}_{\boldsymbol{o}_{\le t}\sim\pi_\theta}\Big[\nabla_\theta \log \pi_\theta(o_t \mid \boldsymbol{o}_{<t})\, Q^\pi(o_t; \boldsymbol{o}_{<t})\Big].$$

Recall that in RL the action value can be decomposed as $Q^\pi(o_t; \boldsymbol{o}_{<t}) = V^\pi(\boldsymbol{o}_{<t}) + A^\pi(o_t; \boldsymbol{o}_{<t})$, applying Lemma 2 once more with the baseline choice $b(\boldsymbol{o}_{\le t}) = V^\pi(\boldsymbol{o}_{<t})$, we obtain

$$G_t = \mathbb{E}_{\boldsymbol{o}_{\le t}\sim\pi_\theta}\Big[\nabla_\theta \log \pi_\theta(o_t \mid \boldsymbol{o}_{<t})\, A^\pi(o_t; \boldsymbol{o}_{<t})\Big].$$

Summing over t gives the on-policy policy gradient

$$\nabla_\theta \mathcal{J} = \sum_{t=1}^{T} \mathbb{E}_{\boldsymbol{o}_{\le t}\sim\pi_\theta}\Big[\nabla_\theta \log \pi_\theta(o_t \mid \boldsymbol{o}_{<t})\, A^\pi(o_t; \boldsymbol{o}_{<t})\Big].$$

A.2 Proof of Theorem 1

Proof. By Lemma 1, the policy gradient can be written as

$$\nabla_\theta \mathcal{J}(\theta) = \sum_{t=1}^{T} \mathbb{E}_{o_{1:t}\sim\pi_\theta}\left[\nabla_\theta \log \pi_\theta(o_t \mid o_{<t}) A^\pi(o_t; o_{<t})\right].$$

Fix any t. Applying the standard change-of-measure identity to the marginal distribution over the prefix $o_{1:t}$ gives

$$\begin{aligned}
&\mathbb{E}_{o_{1:t}\sim\pi_\theta}\left[\nabla_\theta \log \pi_\theta(o_t \mid o_{<t}) A^\pi(o_t; o_{<t})\right] \\
&\quad= \sum_{o_{1:t}} \mathbb{P}_\theta(o_{1:t}) \nabla_\theta \log \pi_\theta(o_t \mid o_{<t}) A^\pi(o_t; o_{<t}) \\
&\quad= \sum_{o_{1:t}} \mathbb{P}_{\theta_{\text{old}}}(o_{1:t}) \frac{\mathbb{P}_\theta(o_{1:t})}{\mathbb{P}_{\theta_{\text{old}}}(o_{1:t})} \nabla_\theta \log \pi_\theta(o_t \mid o_{<t}) A^\pi(o_t; o_{<t}) \\
&\quad= \mathbb{E}_{o_{1:t}\sim\pi_{\theta_{\text{old}}}}\left[\frac{\mathbb{P}_\theta(o_{1:t})}{\mathbb{P}_{\theta_{\text{old}}}(o_{1:t})} \nabla_\theta \log \pi_\theta(o_t \mid o_{<t}) A^\pi(o_t; o_{<t})\right].
\end{aligned}$$

For an autoregressive policy, the probability of a prefix factorizes as

$$\mathbb{P}_\theta(o_{1:t}) = \prod_{i=1}^{t} \pi_\theta(o_i \mid o_{<i}), \qquad \mathbb{P}_{\theta_{\text{old}}}(o_{1:t}) = \prod_{i=1}^{t} \pi_{\theta_{\text{old}}}(o_i \mid o_{<i}).$$

Therefore,

$$\frac{\mathbb{P}_\theta(o_{1:t})}{\mathbb{P}_{\theta_{\text{old}}}(o_{1:t})} = \prod_{i=1}^{t} \frac{\pi_\theta(o_i \mid o_{<i})}{\pi_{\theta_{\text{old}}}(o_i \mid o_{<i})} = \prod_{i=1}^{t} \rho_i =: \rho_{1:t}.$$

Substituting this identity back into the policy-gradient expression yields

$$\nabla_\theta \mathcal{J}(\theta) = \sum_{t=1}^{T} \mathbb{E}_{o_{1:t}\sim\pi_{\theta_{\text{old}}}}\left[\rho_{1:t} \nabla_\theta \log \pi_\theta(o_t \mid o_{<t}) A^\pi(o_t; o_{<t})\right],$$

which proves the theorem. □

B Implementation Details

The pre-trained LLMs can be downloaded from `https://huggingface.co/Qwen`. The training dataset DAPO-Math-17K is available at `https://huggingface.co/datasets/BytedTsinghua-SIA/DAPO-Math-17k`, and the evaluation datasets can be downloaded from `https://huggingface.co/math-ai`. Table 5 summarizes the key hyperparameters used in the RL algorithms. Experiments on Qwen3-8B-Base and Qwen3-14B-Base are conducted using FSDP training [Zhao et al., 2023], and experiments on Qwen3-30B-A3B-Base employ Megatron [Shoeybi et al., 2019] for distributed training. All experiments use vLLM [Kwon et al., 2023] as the inference engine.

Table 5: Key hyperparameters in RL algorithms. "—" denotes not used.

Category	GRPO	GSPO	CISPO	M2PO	MinPRO
Sampling and Validation					
Temperature	1.0	1.0	1.0	1.0	1.0
Top-p / Val Top-p	1.0 / 0.7	1.0 / 0.7	1.0 / 0.7	1.0 / 0.7	1.0 / 0.7
Clipping					
Clip ratio (low / high)	0.2 / 0.28	$2e-3$ / $2e-3$	1 / 4	—	1 / 4
Sequence Limits					
Max prompt / response len	2048 / 20480	2048 / 20480	2048 / 20480	2048 / 20480	2048 / 20480
Overlong buffer (on/off)	off	off	off	off	off
Batching					
Responses per prompt	8	8	8	8	8
Optimization					
Loss aggregation	`token-mean`	`seq-mean`	`token-mean`	`token-mean`	`token-mean`
Actor LR	$1e-6$	$1e-6$	$1e-6$	$1e-6$	$1e-6$
LR warmup steps	10	10	10	10	10

verl/trainer/ppo/core_algos.py

```python
@register_policy_loss("minpro")
def compute_policy_loss_minpro(old_log_prob, log_prob, advantages, response_mask,
                               loss_agg_mode="token-mean", config=None):
    assert config is not None
    low = config.clip_ratio_low
    high = config.clip_ratio_high
    low = config.clip_ratio if low is None else low
    high = config.clip_ratio if high is None else high

    log_ratio = (log_prob - old_log_prob).clamp(-20, 20)
    ratio = log_ratio.exp()
    valid = response_mask.bool()

    # Exclusive minimum over preceding valid response tokens.
    inclusive_min = ratio.masked_fill(~valid, float("inf")).cummin(-1).values
    prefix_min = torch.cat(
        [torch.ones_like(ratio[..., :1]), inclusive_min[..., :-1]], dim=-1
    )
    prev_valid = valid.long().cumsum(-1) - valid.long()
    prefix_min = torch.where(valid & (prev_valid > 0),
                             prefix_min, torch.ones_like(prefix_min))

    correction = ((ratio * prefix_min).clamp(1 - low, 1 + high)
                  / ratio).detach()
    pg_loss = agg_loss(-advantages * ratio * correction,
                       response_mask, loss_agg_mode)
    ppo_kl = verl_F.masked_mean(-log_ratio, response_mask)
    zero = log_prob.new_zeros(())
    return pg_loss, zero, ppo_kl, zero
```

Listing 1: MinPRO policy-loss implementation in VeRL.

B.1 MinPRO Implementation

Our implementation builds on VeRL[2] [Sheng et al., 2025c]. The MinPRO loss in Listing 1 is added to `verl/trainer/ppo/core_algos.py`, registered under `minpro`, and enabled by setting `actor_rollout_ref.actor.policy_loss.loss_mode=minpro`. We adapt the stale-rollout FIFO mechanism and queue checkpoint utilities from the M2PO[3] [Zheng et al., 2026].

[2] https://github.com/verl-project/verl
[3] https://github.com/Infini-AI-Lab/M2PO

C Additional Qwen3-30B-A3B-Base Results

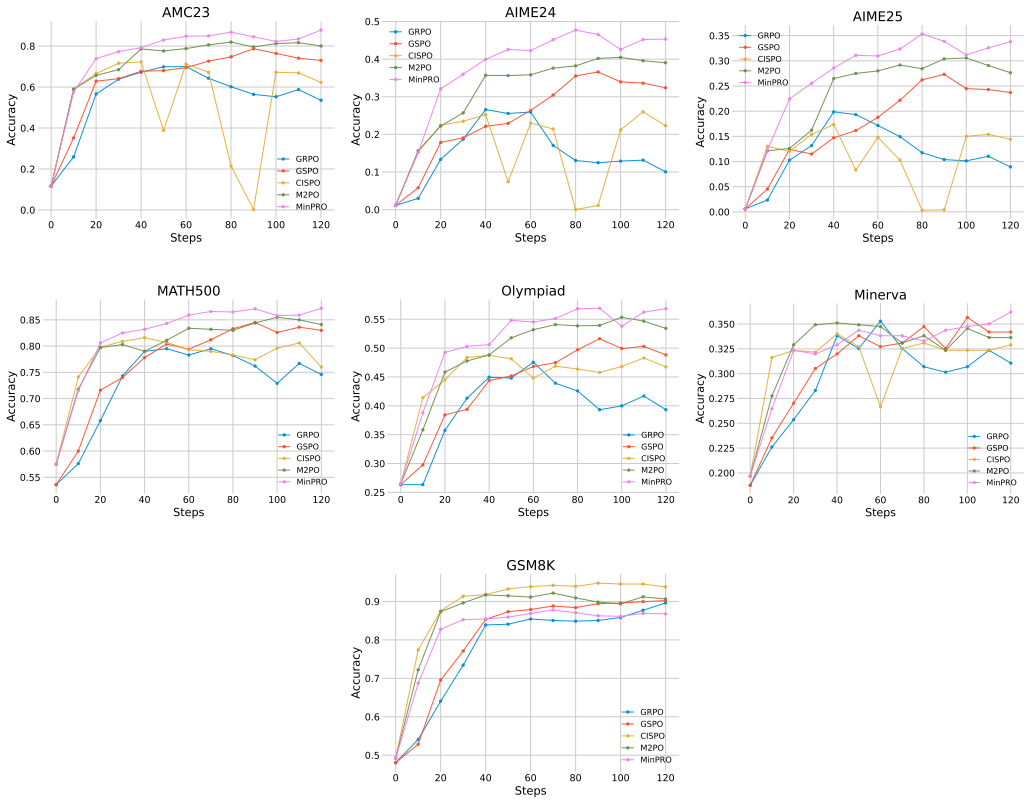


Figure 5: Per-dataset `pass@1` scores with Qwen3-30B-A3B-Base under off-policy training.

D Comparison with AsymRE

Table 6: `pass@1` score comparison with AsymRE at $n = 2$. The $n = 4$ results are in Table 4(b).

Setting	Method	AMC23	AIME24	AIME25	MATH500	Olympiad	Minerva	GSM8K	Avg
$n = 2$	GRPO	63.3	23.1	18.5	70.2	40.4	29.0	80.4	46.4
	AsymRE	58.8	22.0	14.2	79.2	43.8	**34.9**	85.8	48.4
	MinPRO	**82.5**	**36.1**	**30.3**	**86.5**	**56.5**	33.6	**90.2**	**59.4**

AsymRE [Arnal et al., 2025] is also designed for stabilizing off-policy RL by asymmetrically balancing positive and negative rewards. Its key idea is to reduce the reward baseline, so that positive samples receive larger relative advantages and therefore contribute more strongly to the policy-gradient update. To compare with this direction under our setting, we implement an AsymRE-style baseline by modifying the group-normalized advantage computation. Specifically, for a group of responses with rewards $\{R_j\}$, we compute

$$\hat{A}_i = \frac{R_i - \big(\mathrm{mean}\{R_j\} - \alpha\big(\mathrm{mean}\{R_j\} - \min\{R_j\}\big)\big)}{\mathrm{std}\{R_j\}},$$

where α controls the amount of baseline reduction. We set $\alpha = 0.5$ in our experiments.

We evaluate AsymRE on Qwen3-8B-Base under $n = 2$ and $n = 4$. Table 6 reports $n = 2$, and Table 4(b) in Section 5.3 reports $n = 4$. When $n = 2$, AsymRE improves the average pass@1 score from 46.4 to 48.5, but its gains are inconsistent across benchmarks. In contrast, MinPRO consistently achieves substantially larger gains, improving the average score to 59.4.

E Position-Conditioned Prefix Factors and Effective Weights

The minimum-prefix factor can attenuate the updates of later tokens after encountering a small ratio in the preceding prefix. To examine this behavior, we report aggregate diagnostics for Qwen3-8B-Base at $n = 4$. Let $m_t = \underline{\rho}_t$ denote the minimum-prefix factor applied at position t. Table 7 summarizes its distribution over all tokens and within the first and last deciles of the response.

Table 7: Position-conditioned distribution of the minimum-prefix factor for Qwen3-8B-Base at $n = 4$. Threshold columns give the percentage of tokens whose factor exceeds the stated value.

Token position	Mean	Median	$m > 0.5$	$m > 0.2$	$m > 0.1$	$m > 0.05$
All tokens	0.301	0.258	22.00%	58.55%	76.05%	86.64%
First 10%	0.357	0.292	29.11%	62.70%	78.39%	87.64%
Last 10%	0.281	0.239	19.23%	56.05%	74.39%	85.68%

The mean and median are lower in the last decile than in the first decile, consistent with stronger attenuation at later positions. A nonzero mean does not exclude a substantial lower tail: overall, 23.95% of factors are at most 0.1, and 13.36% are at most 0.05. In the last decile, 14.32% are at most 0.05. Conversely, 76.05% of all factors exceed 0.1, and 22.00% exceed 0.5. These statistics quantify the extent of attenuation, but do not determine whether the downweighted tokens correspond to useful exploratory reasoning.

We also compare the average effective token weights assigned by MinPRO and CISPO in Table 8. The CISPO weight is approximately 1 across positions, whereas the MinPRO weight decreases from 0.2502 in the first decile to 0.2036 in the last decile, an approximately 18.6% relative decrease. Thus, the reported averages show substantial attenuation relative to CISPO, with additional attenuation at later positions.

Table 8: Average effective token weights, grouped by relative position in the response, for Qwen3-8B-Base at $n = 4$.

Relative token position	MinPRO	CISPO
0–10%	0.2502	0.9960
10–20%	0.2232	1.0007
20–30%	0.2185	1.0005
30–40%	0.2155	1.0007
40–50%	0.2140	1.0006
50–60%	0.2129	1.0008
60–70%	0.2109	1.0006
70–80%	0.2095	1.0002
80–90%	0.2087	1.0006
90–100%	0.2036	1.0005

These aggregate diagnostics do not specify the token denominator, exact checkpoint, or aggregation procedure, and do not include uncertainty estimates or a breakdown by training step. They therefore describe average weighting behavior without establishing that useful exploration is preserved or that attenuation is uniform throughout training. In particular, a positive sequence-level advantage does not identify the causal contribution of each token, and a nonzero average weight is not a direct measure of successful learning from rare reasoning steps.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: Please see Abstract and Section 1.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Please see Section 6.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

Justification: Please see Section 4.1 and Appendix A.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: Please see Section 5 and Appendix B for implementation details.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [No]

Justification: We use only public datasets for both training and evaluation, please refer to Section 5.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Please see Section 5 and Appendix B for implementation details.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: The main experiments focus on RL post-training of LLMs, which is computationally intensive. Due to limited computational resources, we conducted the LLM training with a single fixed random seed rather than multiple independent runs.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).

- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: All 8B and 14B model experiments were conducted on 4×8 NVIDIA H20 GPUs, while the 30B MoE model experiments were conducted on 8×8 NVIDIA H20 GPUs.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: This research strictly adheres to the NeurIPS Code of Ethics.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: We study general machine learning problem of LLM RL post-training. There is no obvious negative societal impact.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.

- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: This research poses no such risks.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We have cited the original paper that produced the dataset; please refer to Section 5.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.

- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification: The paper does not release new assets.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: This paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: This paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.

- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: The core method development in this research does not involve LLMs as any important, original, or non-standard components.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.