ByteDance | Seed

清华大学 智能产业研究院 Institute for AI Industry Research, Tsinghua University

DAPO: An Open-Source LLM Reinforcement Learning System at Scale

[1]ByteDance Seed [2]Institute for AI Industry Research (AIR), Tsinghua University
[3]The University of Hong Kong
[4]SIA-Lab of Tsinghua AIR and ByteDance Seed

Full author list in Contributions

Abstract

Inference scaling empowers LLMs with unprecedented reasoning ability, with reinforcement learning as the core technique to elicit complex reasoning. However, key technical details of state-of-the-art reasoning LLMs are concealed (such as in OpenAI o1 blog and DeepSeek R1 technical report), thus the community still struggles to reproduce their RL training results. We propose the **D**ecoupled Clip and **D**ynamic s**A**mpling **P**olicy **O**ptimization (**DAPO**) algorithm, and fully open-source a state-of-the-art large-scale RL system that achieves 50 points on AIME 2024 using Qwen2.5-32B base model. Unlike previous works that withhold training details, we introduce four key techniques of our algorithm that make large-scale LLM RL a success. In addition, we open-source our training code, which is built on the **verl** framework [a], along with a carefully curated and processed dataset. These components of our open-source system enhance reproducibility and support future research in large-scale LLM RL.

Date: March 17, 2025

Correspondence: zhouhao@air.tsinghua.edu.cn, wangmingxuan.89@bytedance.com

Project Page: https://dapo-sia.github.io/

[a] https://github.com/volcengine/verl

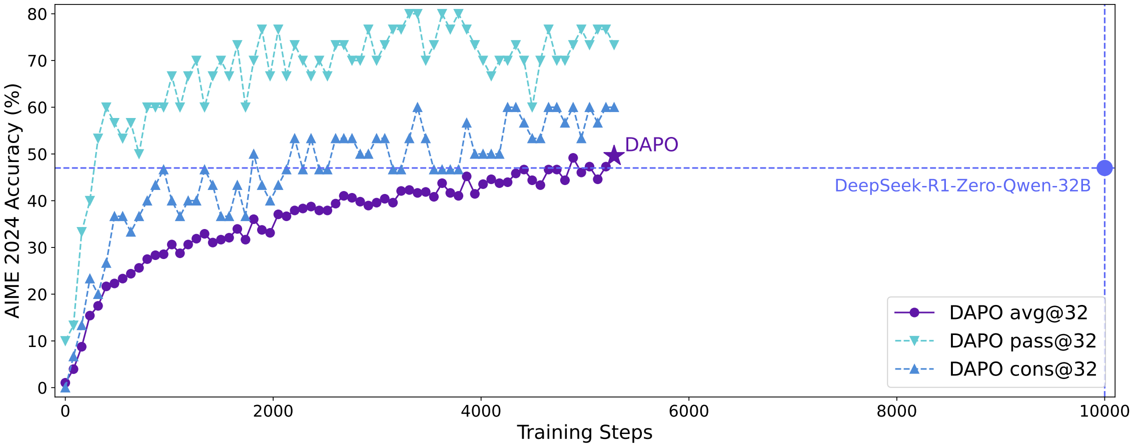


Figure 1 AIME 2024 scores of **DAPO** on the Qwen2.5-32B base model, outperforming the previous SoTA DeepSeek-R1-Zero-Qwen-32B using 50% training steps. The x-axis represents the gradient update steps.

1 Introduction

Test-time scaling such as OpenAI's o1 [1] and DeepSeek's R1 [2] brings a profound paradigm shift to Large Language Models (LLMs) [3–7]. Test-time scaling enables longer Chain-of-Thought thinking and induces sophisticated reasoning behaviors, which makes the models superior in competitive math and coding tasks like AIME and Codeforces.

The central technique driving the revolution is large-scale Reinforcement Learning (RL), which elicits complex reasoning behaviors such as self-verification and iterative refinement. However, the actual algorithm and key recipe for scalable RL training remains a myth, hidden from technical reports of existing reasoning models [1, 2, 8–11]. In this paper, we reveal significant obstacles in large-scale RL training and open-source a scalable RL system with fully open-sourced algorithm, training code and dataset that provides democratized solutions with industry-level RL results.

We experiment over Qwen2.5-32B [12] as the pretrained model for RL. In our initial GRPO run, we achieved only 30 points on AIME — a performance significantly below DeepSeek's RL (47 points). A thorough analysis reveals that the naive GRPO baseline suffers from several key issues such as entropy collapse, reward noise, and training instability. The broader community has encountered similar challenges in reproducing DeepSeek's results [13–19] suggesting that critical training details may have been omitted in the R1 paper that are required to develop an industry-level, large-scale, and reproducible RL system.

To close this gap, we release an open-source state-of-the-art system for large-scale LLM RL, which achieves 50 points on AIME 2024 based on Qwen2.5-32B model, outperforming previous state-of-the-art results achieved by DeepSeek-R1-Zero-Qwen-32B [2] (47 points) using 50% training steps (Figure 1). We propose the **D**ecoupled Clip and **D**ynamic s**A**mpling **P**olicy **O**ptimization (**DAPO**) algorithm, and introduce 4 key techniques to make RL shine in the long-CoT RL scenario. Details are presented in Section 3.

1. **Clip-Higher**, which promotes the diversity of the system and avoids entropy collapse;
2. **Dynamic Sampling**, which improves training efficiency and stability;
3. **Token-Level Policy Gradient Loss**, which is critical in long-CoT RL scenarios;
4. **Overlong Reward Shaping**, which reduces reward noise and stabilizes training.

Our implementation is based on verl [20]. By fully releasing our state-of-the-art RL system including training code and data, we aim to reveal valuable insights to large-scale LLM RL that benefit the larger community.

2 Preliminary

2.1 Proximal Policy Optimization (PPO)

PPO [21] introduces a clipped surrogate objective for policy optimization. By constraining the policy updates within a proximal region of the previous policy using clip, PPO stabilizes training and improves sample efficiency. Specifically, PPO updates the policy by maximizing the following objective:

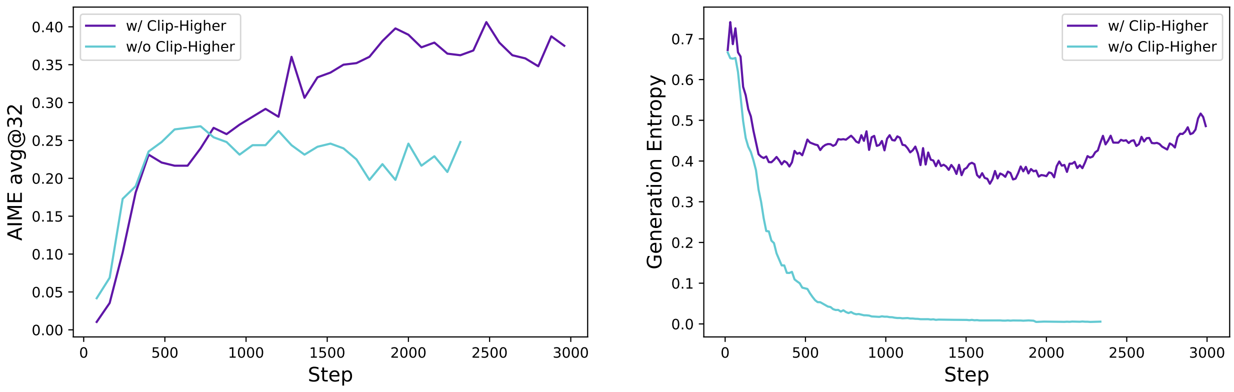
$$\mathcal{J}_{\text{PPO}}(\theta) = \mathbb{E}_{(q,a)\sim\mathcal{D},o_{\le t}\sim\pi_{\theta_{\text{old}}}(\cdot|q)}\left[\min\left(\frac{\pi_\theta(o_t \mid q, o_{<t})}{\pi_{\theta_{\text{old}}}(o_t \mid q, o_{<t})}\hat{A}_t,\ \text{clip}\left(\frac{\pi_\theta(o_t \mid q, o_{<t})}{\pi_{\theta_{\text{old}}}(o_t \mid q, o_{<t})}, 1-\varepsilon, 1+\varepsilon\right)\hat{A}_t\right)\right], \quad (1)$$

where (q, a) is a question-answer pair from the data distribution $\mathcal{D}$, ε is the clipping range of importance sampling ratio, and $\hat{A}_t$ is an estimator of the advantage at time step t. Given the value function V and the reward function R, $\hat{A}_t$ is computed using the Generalized Advantage Estimation (GAE) [22]:

$$\hat{A}_t^{\text{GAE}(\gamma,\lambda)} = \sum_{l=0}^{\infty}(\gamma\lambda)^l\delta_{t+l}, \quad (2)$$

where

$$\delta_l = R_l + \gamma V(s_{l+1}) - V(s_l), \quad 0 \le \gamma, \lambda \le 1. \quad (3)$$



(a) Accuracies on AIME.

(b) Entropy of actor model.

Figure 2 The accuracy on the AIME test set and the entropy of the actor model's generated probabilities during the RL training process, both before and after applying **Clip-Higher** strategy.

2.2 Group Relative Policy Optimization (GRPO)

Compared to PPO, GRPO eliminates the value function and estimates the advantage in a group-relative manner. For a specific question-answer pair (q, a), the behavior policy $\pi_{\theta_{\text{old}}}$ samples a group of G individual responses $\{o_i\}_{i=1}^G$. Then, the advantage of the i-th response is calculated by normalizing the group-level rewards $\{R_i\}_{i=1}^G$:

$$\hat{A}_{i,t} = \frac{r_i - \text{mean}(\{R_i\}_{i=1}^G)}{\text{std}(\{R_i\}_{i=1}^G)}. \tag{4}$$

Similar to PPO, GRPO adopts a clipped objective, together with a directly imposed KL penalty term:

$$\begin{aligned}\mathcal{J}_{\text{GRPO}}(\theta) = \ & \mathbb{E}_{(q,a)\sim\mathcal{D},\{o_i\}_{i=1}^G\sim\pi_{\theta_{\text{old}}}(\cdot|q)} \\ & \left[\frac{1}{G}\sum_{i=1}^{G}\frac{1}{|o_i|}\sum_{t=1}^{|o_i|}\left(\min\Big(r_{i,t}(\theta)\hat{A}_{i,t},\ \text{clip}\Big(r_{i,t}(\theta), 1-\varepsilon, 1+\varepsilon\Big)\hat{A}_{i,t}\Big) - \beta D_{\text{KL}}(\pi_\theta||\pi_{\text{ref}})\right)\right],\end{aligned} \tag{5}$$

where

$$r_{i,t}(\theta) = \frac{\pi_\theta(o_{i,t} \mid q, o_{i,<t})}{\pi_{\theta_{\text{old}}}(o_{i,t} \mid q, o_{i,<t})}. \tag{6}$$

It is also worth noting that GRPO computes the objective at the sample-level. To be exact, GRPO first calculates the mean loss within each generated sequence, before averaging the loss of different samples. As we will be discussing in Section 3.3, such difference may have an impact on the performance of the algorithm.

2.3 Removing KL Divergence

The KL penalty term is used to regulate the divergence between the online policy and the frozen reference policy. In the RLHF scenario [23], the goal of RL is to align the model behavior without diverging too far from the initial model. However, during training the long-CoT reasoning model, the model distribution can diverge significantly from the initial model, thus this restriction is not necessary. Therefore, we will exclude the KL term from our proposed algorithm.