From PPO to GRPO: A Policy Gradient Approach within DeepSeek R1

Tianbing Xu

January 2025

1 Introduction

In early 2025, DeepSeek R1 ([3]), an LLM with remarkable reasoning capabilities, achieved state-of-the-art results compared to OpenAI's O1 ([2]). Most remarkably, DeepSeek-R1-Zero employs large-scale reinforcement learning (Figure 1) without requiring supervised fine-tuning (SFT) as a preliminary step. Therefore, it is necessary to introduce and explain the RL algorithm (GRPO) used in DeepSeek R1.

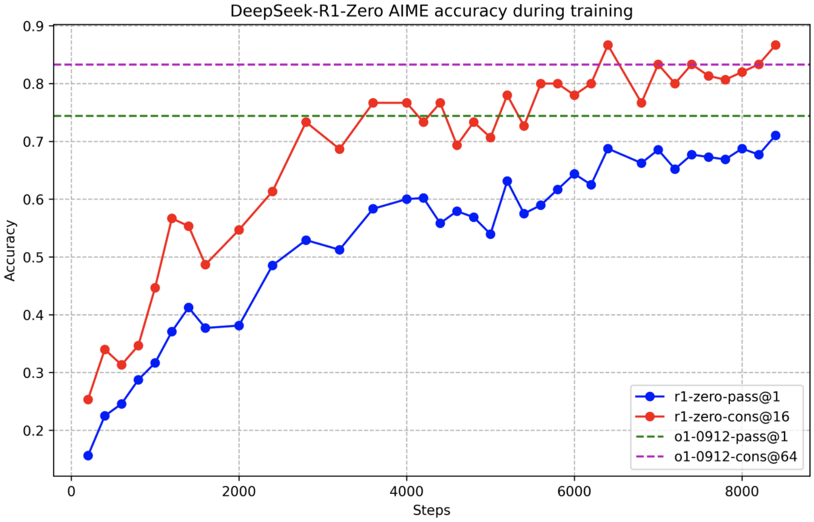

Figure 1: DeepSeek-R1-Zero RL learning curve for AIME

2 Discussion on GRPO

2.1 Vanilla Policy Gradient

Vanilla Policy Gradient is a simple on-policy learning algorithm for policy optimization. The objective function is defined as:

$$L_{\mathrm{PG}}(\theta) = \mathbb{E}_{s\sim P(s),\, a\sim\pi_\theta}\big[A(s,a)\big], \tag{1}$$

where the advantage function $A(s,a)$ is given by:

$$A(s,a) = Q(s,a) - V(s)$$

.

The policy gradient is then computed as:

$$\nabla_\theta L_{\mathrm{PG}}(\theta) = \mathbb{E}_{\pi_\theta}\big[A(s,a)\nabla_\theta \log \pi_\theta(a|s)\big], \tag{2}$$

2.2 Proximal Policy Optimization(PPO, [1])

$$L_{\mathrm{PPO}}(\theta) = \mathbb{E}_{\pi_{\theta_{\mathrm{old}}}}\left\{\min\Big(r(\pi_\theta,\pi_{\theta_{\mathrm{old}}})A(s,a),\, \mathrm{clip}\big(r(\pi_\theta,\pi_{\theta_{\mathrm{old}}}),1-\epsilon,1+\epsilon\big)A(s,a)\Big)\right\}, \tag{3}$$

where, the ratio of two distributions is given by $r(p,q) = p/q$. The trajectory sequences $\{(s,a,r)\}$ are sampled from the distribution induced by the previous policy $\pi_{\theta_{old}}$. For simplicity, let $\pi_{\theta_{old}} = \pi_\theta$, which implies that the trajectories are sampled based on the current policy. In this case, PPO reduces to the vanilla policy gradient.

2.3 Group Relatively Policy Optimization(GRPO, [3])

$$\begin{aligned} L_{\mathrm{GRPO}}(\theta) =& \mathbb{E}_{\pi_{\theta_{\mathrm{old}}}}\Big\{\min\Big(r(\pi_\theta,\pi_{\theta_{\mathrm{old}}})A(s,a),\, \mathrm{clip}\big(r(\pi_\theta,\pi_{\theta_{\mathrm{old}}}),1-\epsilon,1+\epsilon\big)A(s,a)\Big)\Big\} \\ & - \beta\, \mathbb{E}_{\pi_{\theta_{\mathrm{old}}}}\big[D_{KL}(\pi_\theta\|\pi_{\mathrm{ref}})\big], \end{aligned} \tag{4}$$

GRPO is a variant of the PPO algorithm that incorporates a KL divergence term to penalize deviations from the reference policy. Notably, GRPO uses an unbiased estimate for the KL term $D_{KL}(\pi_\theta\|\pi_{\mathrm{ref}})$ with low variance, leveraging control variate techniques. It is straightforward to show that the estimation is unbiased:

$$D_{KL}(q\|p) = \mathbb{E}_q\big[r(p,q) - 1 - \log r(p,q)\big], \tag{5}$$

where the ratio of the two distributions is given by $r(p,q) = p/q$, and we have $\mathbb{E}_q\big[r(p,q)\big] = 1$.

To simplify the policy gradient computation, GRPO samples the responses based on the current policy, i.e., $\pi_{\theta_{old}} = \pi_\theta$. This reduces the algorithm further to Vanilla Policy Gradient with a KL divergence constraint. The simplified policy gradient is then:

$$\nabla_\theta L_{\text{GRPO}}(\theta) = \mathbb{E}_{\pi_\theta}\big\{[A(s,a) + \beta(\pi_{ref}/\pi_\theta - 1)]\nabla_\theta \log \pi_\theta(a|s)\big\}, \tag{6}$$

Another notable simplification is the removal of the value model used to estimate the advantage function $A(s,a)$ (e.g., in PPO), replacing it with a simple normalization method based on a batch of K reward samples from $\pi_\theta(\cdot \mid s)$

$$\hat{r} = \{r_j \mid j = 1, \ldots, K\}$$

to stabilize the RL training process.

$$\hat{A}(a_i|s_i) = \frac{r_i - \mu(\hat{r})}{\delta(\hat{r})}$$

In LLMs, training the value model incurs substantial memory and computational costs. More importantly, an inaccurate value model can hinder the convergence of the policy learning algorithm. Therefore, this simplification can actually improve RL training.

References

[1] John Schulman, et. al, OpenAI, *Proximal Policy Optimization Algorithms., 2017*

[2] OpenAI, *Learning to Reason with LLMs, 2024*

[3] DeepSeek-AI, *DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning, 2025*