RL for LLM Reasoning

汇报人：邓人嘉

目录

1. Deepseek-R1
2. GRPO
3. DAPO

Deepseek-R1 三点贡献

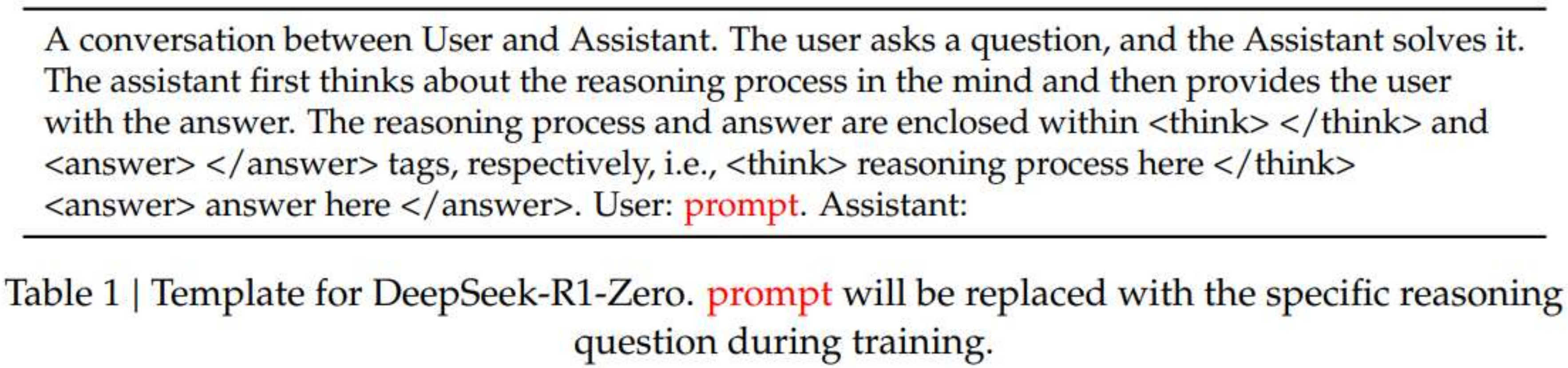
- DeepSeek-R1-Zero：纯强化学习
- DeepSeek-R1：冷启动SFT → RL → CoT/通用数据 SFT(80w条) → 全场景RL
- 蒸馏小模型：用第三阶段的80w条数据SFT小模型

Deepseek-R1-Zero

- 直接从Deepseek-v3-base做纯RL
- 数据：数学 + 代码(leetcode)
- 使用基于规则的奖励（准确率奖励+格式奖励）

$$r(o|q) = accuracy(o) + format\_reward(o)$$

- 优点：
  - 不需要存储Reward Model
  - 人工标注成本低

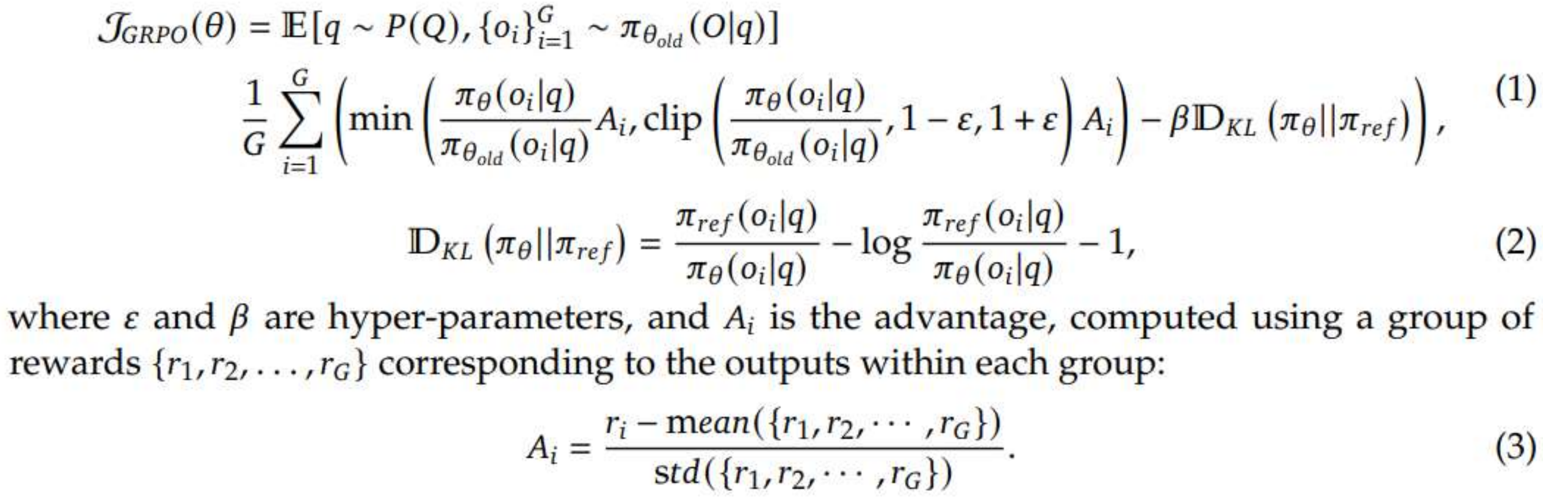


A conversation between User and Assistant. The user asks a question, and the Assistant solves it. The assistant first thinks about the reasoning process in the mind and then provides the user with the answer. The reasoning process and answer are enclosed within <think> </think> and <answer> </answer> tags, respectively, i.e., <think> reasoning process here </think> <answer> answer here </answer>. User: prompt. Assistant:

Table 1 | Template for DeepSeek-R1-Zero. prompt will be replaced with the specific reasoning question during training.

Deepseek-R1-Zero

- GRPO



$$\mathcal{J}_{GRPO}(\theta) = \mathbb{E}[q \sim P(Q), \{o_i\}_{i=1}^{G} \sim \pi_{\theta_{old}}(O|q)]$$

$$\frac{1}{G}\sum_{i=1}^{G}\left(\min\left(\frac{\pi_\theta(o_i|q)}{\pi_{\theta_{old}}(o_i|q)}A_i, \text{clip}\left(\frac{\pi_\theta(o_i|q)}{\pi_{\theta_{old}}(o_i|q)}, 1-\varepsilon, 1+\varepsilon\right)A_i\right) - \beta\mathbb{D}_{KL}\left(\pi_\theta||\pi_{ref}\right)\right), \tag{1}$$

$$\mathbb{D}_{KL}\left(\pi_\theta||\pi_{ref}\right) = \frac{\pi_{ref}(o_i|q)}{\pi_\theta(o_i|q)} - \log\frac{\pi_{ref}(o_i|q)}{\pi_\theta(o_i|q)} - 1, \tag{2}$$

where ε and β are hyper-parameters, and A_i is the advantage, computed using a group of rewards $\{r_1, r_2, \ldots, r_G\}$ corresponding to the outputs within each group:

$$A_i = \frac{r_i - \text{mean}(\{r_1, r_2, \cdots, r_G\})}{\text{std}(\{r_1, r_2, \cdots, r_G\})}. \tag{3}$$

PPO Memory = Actor Model + Critic Model + Reference Model + Reward Model

R1 GRPO Memory = Actor Model + Reference Model

红色表示训练，蓝色表示冻结

GRPO相较于PPO节省了约50%内存

Deepseek-R1-Zero

- self-evolution

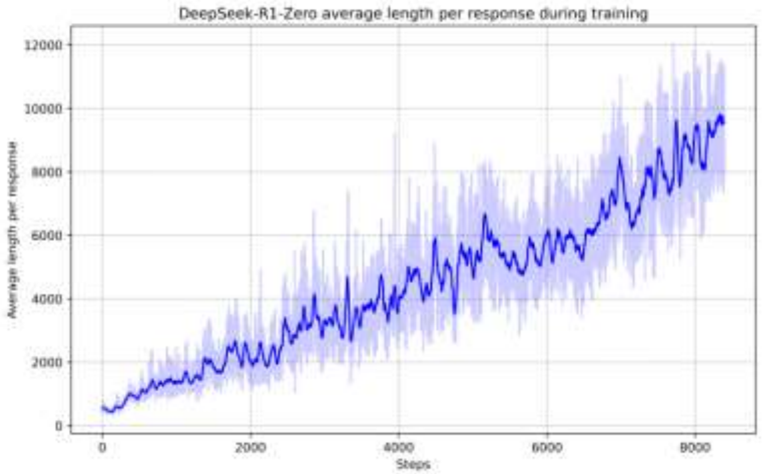


Figure 3 | The average response length of DeepSeek-R1-Zero on the training set during the RL process. DeepSeek-R1-Zero naturally learns to solve reasoning tasks with more thinking time.

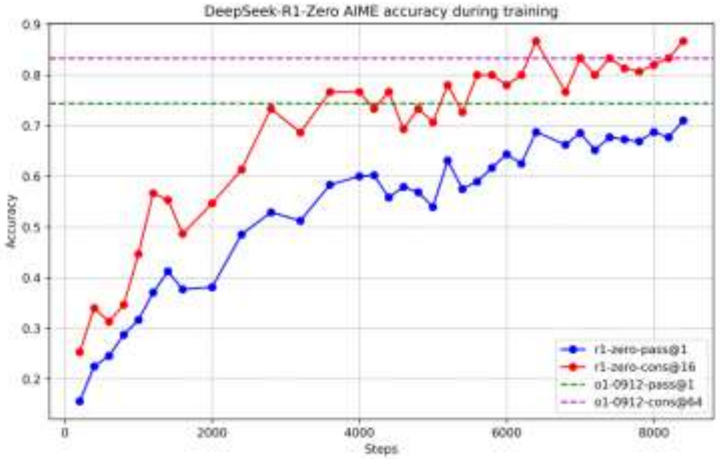
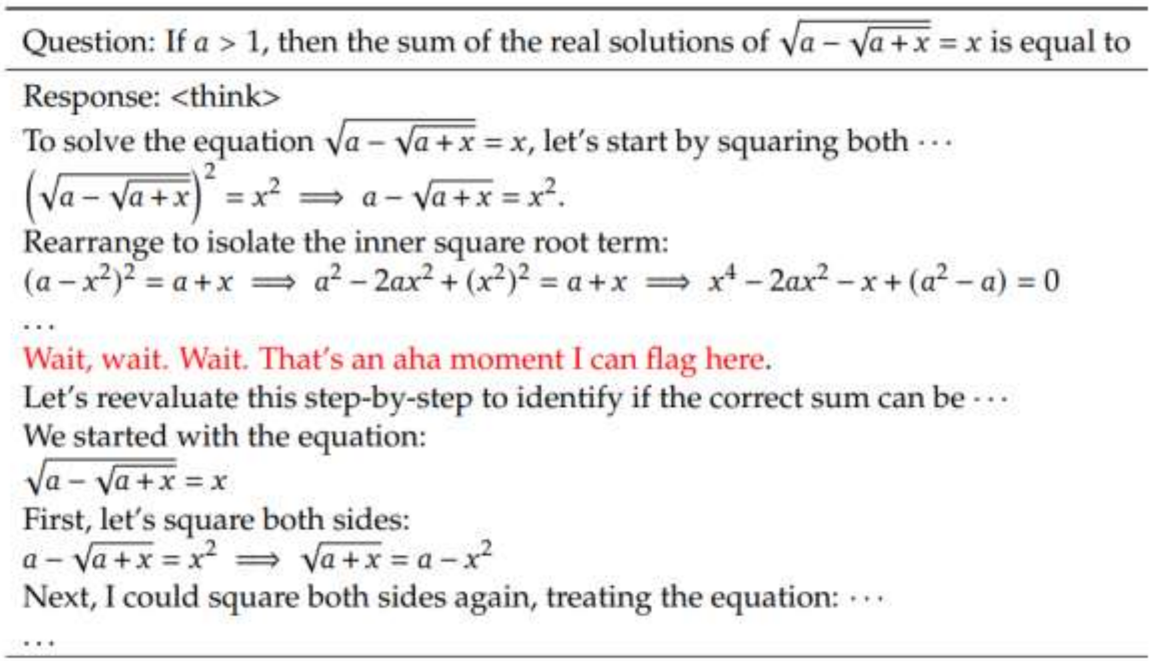


Figure 2 | AIME accuracy of DeepSeek-R1-Zero during training. For each question, we sample 16 responses and calculate the overall average accuracy to ensure a stable evaluation.

Deepseek-R1-Zero

- Aha Moment



Question: If $a > 1$, then the sum of the real solutions of $\sqrt{a - \sqrt{a + x}} = x$ is equal to

Response: <think>

To solve the equation $\sqrt{a - \sqrt{a + x}} = x$, let's start by squaring both $\cdots$

$\left(\sqrt{a - \sqrt{a + x}}\right)^2 = x^2 \implies a - \sqrt{a + x} = x^2.$

Rearrange to isolate the inner square root term:

$(a - x^2)^2 = a + x \implies a^2 - 2ax^2 + (x^2)^2 = a + x \implies x^4 - 2ax^2 - x + (a^2 - a) = 0$

$\ldots$

Wait, wait. Wait. That's an aha moment I can flag here.

Let's reevaluate this step-by-step to identify if the correct sum can be $\cdots$

We started with the equation:

$\sqrt{a - \sqrt{a + x}} = x$

First, let's square both sides:

$a - \sqrt{a + x} = x^2 \implies \sqrt{a + x} = a - x^2$

Next, I could square both sides again, treating the equation: $\cdots$

$\ldots$

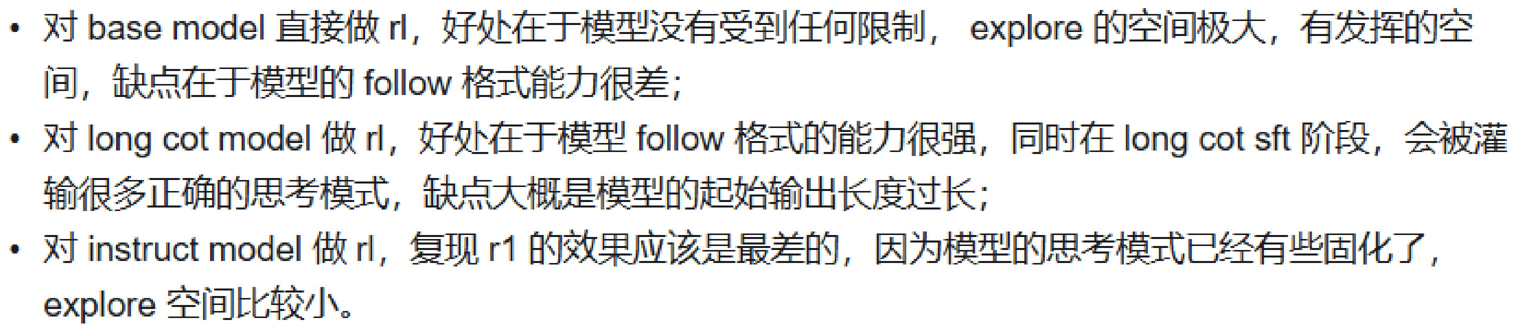
Table 3 | An interesting "aha moment" of an intermediate version of DeepSeek-R1-Zero. The model learns to rethink using an anthropomorphic tone. This is also an aha moment for us, allowing us to witness the power and beauty of reinforcement learning.

R1复现经验：

- 模型回复长度增加不代表成功复现R1，还需要观察到准确率的上升
- 有Aha Moment 不代表成功复现R1，还需要观察到有Aha Moment的数据比没有Aha Moment的数据准确率高

Deepseek-R1-Zero

Q：对base模型做RL 和 对SFT后的模型做RL有什么区别?



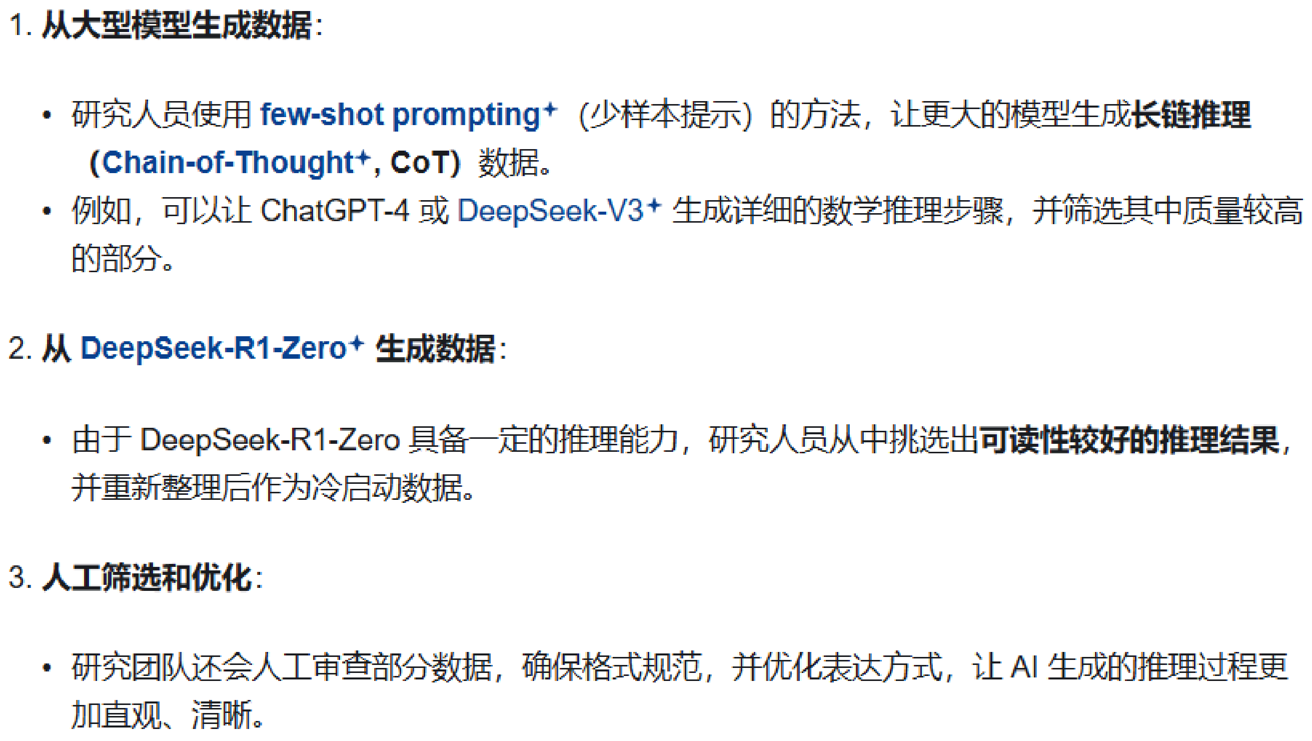
- 对 base model 直接做 rl，好处在于模型没有受到任何限制， explore 的空间极大，有发挥的空间，缺点在于模型的 follow 格式能力很差；
- 对 long cot model 做 rl，好处在于模型 follow 格式的能力很强，同时在 long cot sft 阶段，会被灌输很多正确的思考模式，缺点大概是模型的起始输出长度过长；
- 对 instruct model 做 rl，复现 r1 的效果应该是最差的，因为模型的思考模式已经有些固化了，explore 空间比较小。

Q：R1-Zero效果已经这么好了，为什么不能直接使用?

A：R1-Zero的推理过程质量很差，存在不可读、多语言混合的缺点。需要使用高质量数据先做冷启动再RL能解决这些缺点。

R1 第一阶段：冷启动SFT

- 训练模型：DeepSeek-v3-base
- 数据：数千条高质量、可读性强的数据



1. **从大型模型生成数据**：
   - 研究人员使用 **few-shot prompting**+（少样本提示）的方法，让更大的模型生成**长链推理（Chain-of-Thought+，CoT）**数据。
   - 例如，可以让 ChatGPT-4 或 DeepSeek-V3+ 生成详细的数学推理步骤，并筛选其中质量较高的部分。
2. **从 DeepSeek-R1-Zero+ 生成数据**：
   - 由于 DeepSeek-R1-Zero 具备一定的推理能力，研究人员从中挑选出**可读性较好的推理结果**，并重新整理后作为冷启动数据。
3. **人工筛选和优化**：
   - 研究团队还会人工审查部分数据，确保格式规范，并优化表达方式，让 AI 生成的推理过程更加直观、清晰。

- 冷启动好处：让模型在后续RL中生成更有意义的推理过程。

R1 第二阶段：推理导向的RL

- 训练模型：冷启动后的R1-sft1
- 数据：R1-zero训练的数据
- Reward = 准确率奖励 + 语言一致性奖励

- 语言一致性奖励即CoT中目标语言单词的比例，减少推理链中的语言混合问题。

R1 第三阶段：拒绝采样+SFT

- 训练模型：DeepSeek-v3-base
- 推理数据（600K）
  - 能使用准确率奖励的数据：提供一些问题，使用上一阶段的R1-rl1生成k个结果，使用规则奖励挑选奖励最大的结果。
  - 不能使用准确率奖励的数据：提供一些问题，使用R1-rl1生成k个结果，把模型生成结果和 ground-truth同时喂给DeepSeek-v3来对比两者的差异。
  - 过滤含有混合语言、长段落、有代码块的数据
- 非推理数据（200K）
  - 部分v3使用过的sft数据
  - 提供提示词，让DeepSeek-v3生成回答。对于简单问题直接生成回答，困难问题生成CoT+回答。

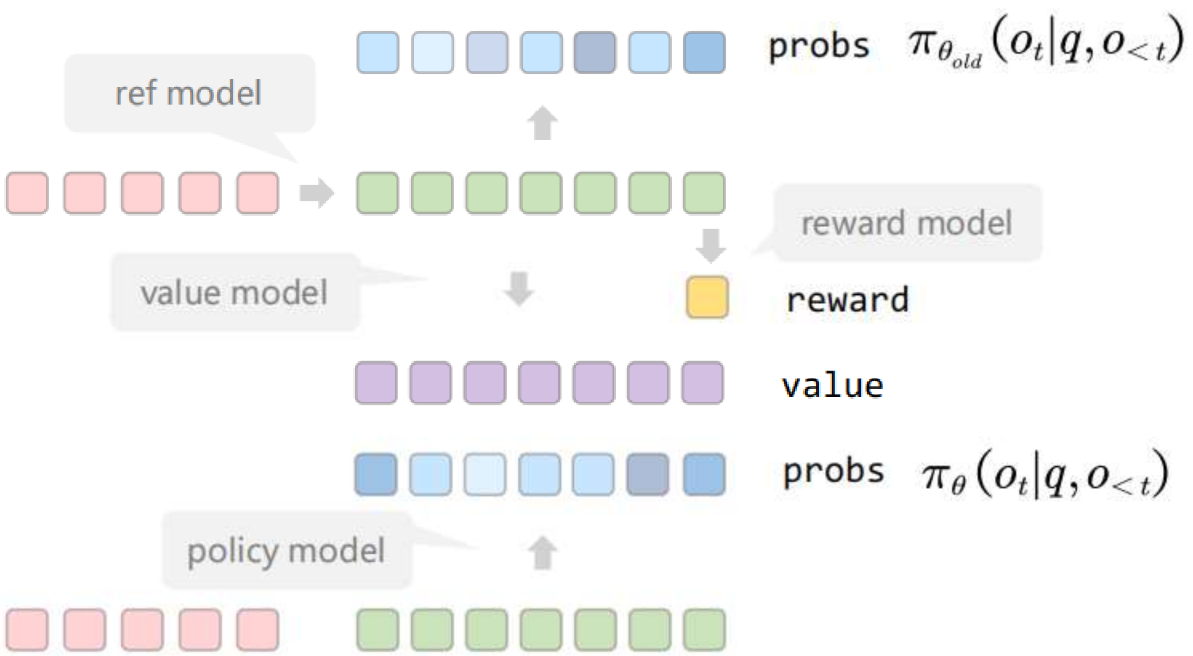
- 在DeepSeek-v3-base上微调共800K条数据

R1 第四阶段：全场景强化学习

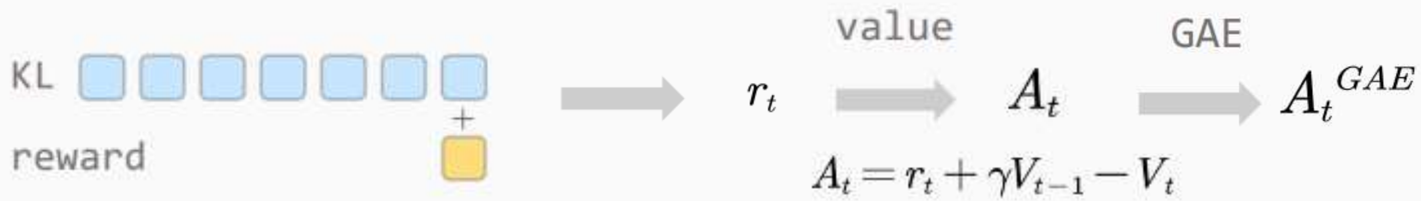
- 训练模型：第三阶段得到的R1-sft2
- 推理数据 - 提升推理能力：
  - 使用R1-Zero的RL方法（基于规则的奖励+GRPO）
- 通用数据 - 对齐人类偏好：
  - 使用标准的RL方法（正常奖励模型捕捉人类偏好+GRPO）

PPO回顾

$$\mathcal{J}_{PPO}(\theta) = \mathbb{E}[q \sim P(Q), o \sim \pi_{\theta_{old}}(O|q)] \frac{1}{o} \sum_{t=1}^{|o|} \min\left[\frac{\pi_\theta(o_t|q,o_{<t})}{\pi_{\theta_{old}}(o_t|q,o_{<t})} A_t, clip\left(\frac{\pi_\theta(o_t|q,o_{<t})}{\pi_{\theta_{old}}(o_t|q,o_{<t})}, 1-\epsilon, 1+\epsilon \right) A_t \right]$$


probs $\pi_{\theta_{old}}(o_t|q,o_{<t})$
ref model
reward model
value model
reward
value
probs $\pi_\theta(o_t|q,o_{<t})$
policy model

$$r_t = \underbrace{r_\varphi(q,o_{\leqslant}t)}_{\text{reward}} - \underbrace{\beta \ln \frac{\pi_\theta(o_t|q,o_{<t})}{\pi_{\theta_{old}}(o_t|q,o_{<t})}}_{\text{KL}}$$



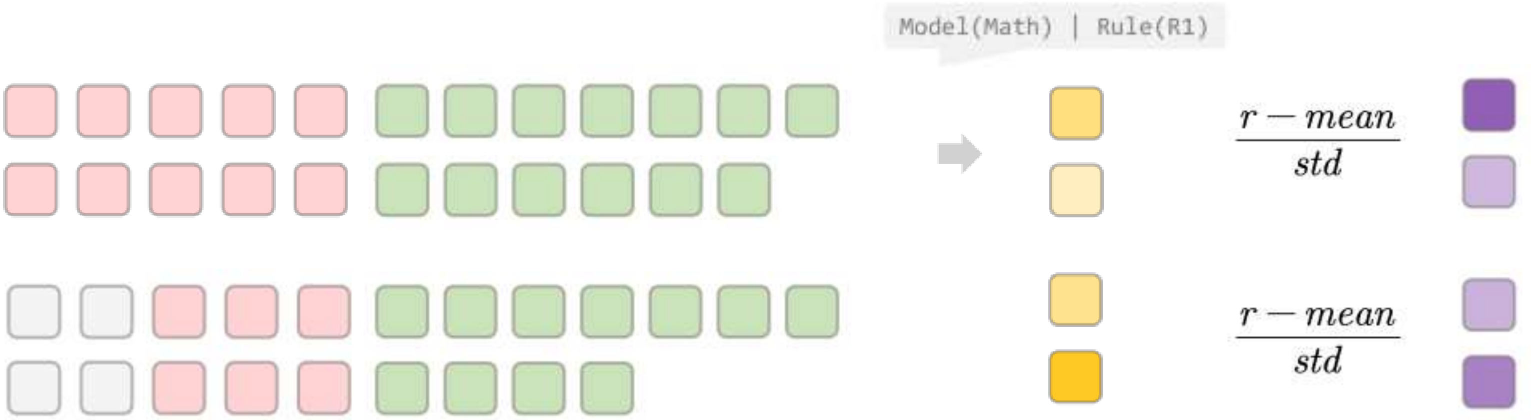
$$A_t = r_t + \gamma V_{t-1} - V_t$$

GRPO

$$\mathcal{J}_{GRPO}(\theta) = \mathbb{E}[q \sim P(Q), \{o_i\}_{i=1}^{G} \sim \pi_{\theta_{old}}(O|q)]$$

$$\frac{1}{G}\sum_{i=1}^{G}\left(\min\left(\frac{\pi_\theta(o_i|q)}{\pi_{\theta_{old}}(o_i|q)}A_i, \text{clip}\left(\frac{\pi_\theta(o_i|q)}{\pi_{\theta_{old}}(o_i|q)}, 1-\varepsilon, 1+\varepsilon\right)A_i\right) - \beta\mathbb{D}_{KL}\left(\pi_\theta||\pi_{ref}\right)\right)$$

$$\mathbb{D}_{KL}\left(\pi_\theta||\pi_{ref}\right) = \frac{\pi_{ref}(o_i|q)}{\pi_\theta(o_i|q)} - \log\frac{\pi_{ref}(o_i|q)}{\pi_\theta(o_i|q)} - 1,$$



$$A_i = \frac{r_i - \text{mean}(\{r_1, r_2, \cdots, r_G\})}{\text{std}(\{r_1, r_2, \cdots, r_G\})}.$$

prompt completions rewards advanteges

为什么不用DPO?

DPO

$$\mathcal{L}_{DPO} = -\mathbb{E}\left[log\sigma\left(\beta log\frac{\pi_\theta(y_w|x)}{\pi_{ref}(y_w|x)} - \beta log\frac{\pi_\theta(y_l|x)}{\pi_{ref}(y_l|x)}\right)\right]$$

判别模型Loss

语言模型

$$\mathcal{L}_{lm} = -\frac{1}{T}\sum_{t=1}^{T} logP(x_t|x_{<t})$$

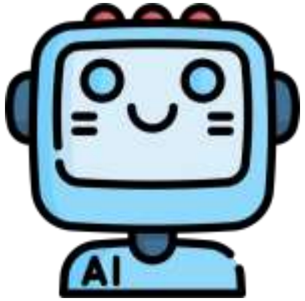
生成模型Loss

$$\mathcal{L}_{DPO} = -\mathbb{E}\left[log\sigma\left(\beta log\frac{\pi_\theta(y_w|x)}{\pi_{ref}(y_w|x)} - \beta log\frac{\pi_\theta(y_l|x)}{\pi_{ref}(y_l|x)}\right) + \alpha log\pi_\theta(y_w|x)\right]$$

Prompt：意大利面应该拌

模型输出 1：蕃茄肉酱

模型输出 2：油泼辣子



Ref Model：油泼辣子

好的DPO Model：蕃茄肉酱

坏的DPO Model：42号混凝土

为什么PPO/GRPO更能避免训崩?

$$\mathcal{L}_{DPO} = -\mathbb{E}\left[log\sigma\left(\beta log\frac{\pi_\theta(y_w|x)}{\pi_{ref}(y_w|x)} - \beta log\frac{\pi_\theta(y_l|x)}{\pi_{ref}(y_l|x)}\right)\right]$$

$$\mathcal{J}_{PPO}(\theta) = \mathbb{E}[q\sim P(Q), o\sim \pi_{\theta_{old}}(O|q)]\frac{1}{o}\sum_{t=1}^{|o|}\min\left[\frac{\pi_\theta(o_t|q,o_{<t})}{\pi_{\theta_{old}}(o_t|q,o_{<t})}A_t, clip\left(\frac{\pi_\theta(o_t|q,o_{<t})}{\pi_{\theta_{old}}(o_t|q,o_{<t})}, 1-\epsilon, 1+\epsilon\right)A_t\right]$$

DPO是offline的。仅从偏好数据中训练，ref模型不知道自己训崩了。
PPO是online的。模型每个epoch自己生成数据进行训练，假设训崩了，ref模型会有很大的KL约束。

DAPO

- **D**ecoupled Clip and Dynamic s**A**mling **P**olicy **O**ptmization
- 对GRPO进行了优化

- 贡献
  - Clip-Higher
  - Dynamic Sampling
  - Token-Level Policy Gradient Loss
  - Overlong Reward Shaping

DAPO：移除KL散度

$$
\begin{aligned}
\mathcal{J}_{\text{DAPO}}(\theta) = \quad & \mathbb{E}_{(q,a)\sim\mathcal{D},\{o_i\}_{i=1}^{G}\sim\pi_{\theta_{\text{old}}}(\cdot|q)} \\
& \left[\frac{1}{\sum_{i=1}^{G}|o_i|}\sum_{i=1}^{G}\sum_{t=1}^{|o_i|}\min\Big(r_{i,t}(\theta)\hat{A}_{i,t},\ \text{clip}\Big(r_{i,t}(\theta),1-\varepsilon_{\text{low}},1+\varepsilon_{\text{high}}\Big)\hat{A}_{i,t}\Big)\right] \\
\text{s.t.} \quad & 0 < \Big|\{o_i \mid \texttt{is\_equivalent}(a,o_i)\}\Big| < G.
\end{aligned}
$$

where

$$
r_{i,t}(\theta) = \frac{\pi_{\theta}(o_{i,t} \mid q, o_{i,<t})}{\pi_{\theta_{\text{old}}}(o_{i,t} \mid q, o_{i,<t})}, \quad \hat{A}_{i,t} = \frac{R_i - \text{mean}(\{R_i\}_{i=1}^{G})}{\text{std}(\{R_i\}_{i=1}^{G})}.
$$

- 移除KL散度的理由：在训练长思维链推理模型时，模型分布可能与初始模型显著偏离，因此这种限制不是必要的。

- 基于规则的奖励建模

$$
R(\hat{y}, y) = \begin{cases} 1, & \texttt{is\_equivalent}(\hat{y}, y) \\ -1, & \text{otherwise} \end{cases}
$$

DAPO：Clip-Higher

$$\mathcal{J}_{\text{DAPO}}(\theta) = \mathbb{E}_{(q,a)\sim\mathcal{D},\{o_i\}_{i=1}^{G}\sim\pi_{\theta_{\text{old}}}(\cdot|q)}$$

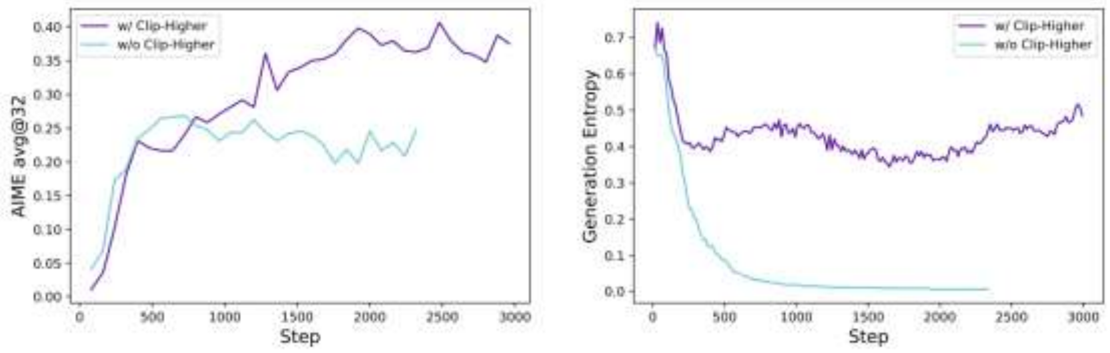
$$\left[\frac{1}{\sum_{i=1}^{G}|o_i|}\sum_{i=1}^{G}\sum_{t=1}^{|o_i|}\min\Big(r_{i,t}(\theta)\hat{A}_{i,t},\ \text{clip}\Big(r_{i,t}(\theta),1-\varepsilon_{\text{low}},1+\varepsilon_{\text{high}}\Big)\hat{A}_{i,t}\Big)\right]$$

$$\text{s.t.}\quad 0<\Big|\{o_i \mid \texttt{is\_equivalent}(a,o_i)\}\Big|<G.$$

where

$$r_{i,t}(\theta)=\frac{\pi_\theta(o_{i,t}\mid q,o_{i,<t})}{\pi_{\theta_{\text{old}}}(o_{i,t}\mid q,o_{i,<t})},\qquad \hat{A}_{i,t}=\frac{R_i-\text{mean}(\{R_i\}_{i=1}^{G})}{\text{std}(\{R_i\}_{i=1}^{G})}.$$

old模型中低概率token想要增加会很难。
假设epsilon为0.2, 旧模型概率为0.01和0.9时，新模型概率最多为0.012和1.08，因此低概率token上升很难



(a) Accuracies on AIME.

(b) Entropy of actor model.

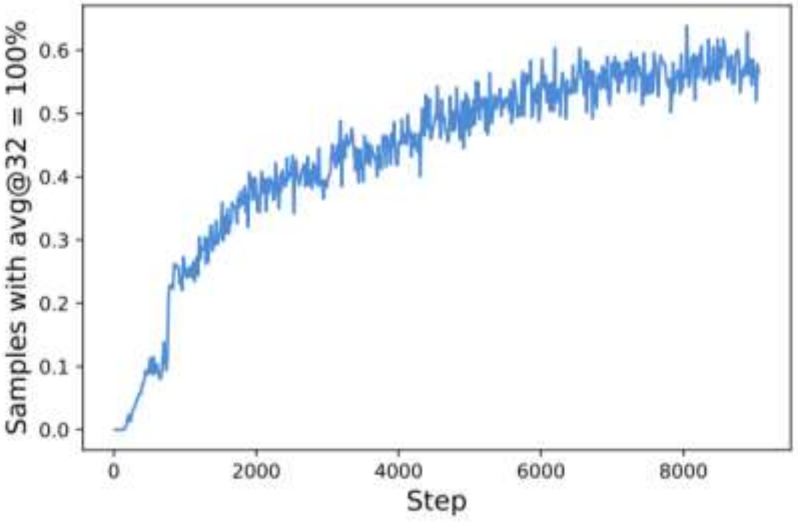
Figure 2 The accuracy on the AIME test set and the entropy of the actor model's generated probabilities during the RL training process, both before and after applying **Clip-Higher** strategy.

DAPO：Dynamic Sampling

$$\mathcal{J}_{\text{DAPO}}(\theta) = \mathbb{E}_{(q,a)\sim\mathcal{D},\{o_i\}_{i=1}^{G}\sim\pi_{\theta_{\text{old}}}(\cdot|q)} \left[\frac{1}{\sum_{i=1}^{G}|o_i|}\sum_{i=1}^{G}\sum_{t=1}^{|o_i|}\min\left(r_{i,t}(\theta)\hat{A}_{i,t},\ \text{clip}\Big(r_{i,t}(\theta),1-\varepsilon_{\text{low}},1+\varepsilon_{\text{high}}\Big)\hat{A}_{i,t}\right)\right]$$

$$\text{s.t.}\quad 0 < \Big|\{o_i \mid \texttt{is\_equivalent}(a,o_i)\}\Big| \quad r_{i,t}(\theta)=\frac{\pi_\theta(o_{i,t}\mid q,o_{i,<t})}{\pi_{\theta_{\text{old}}}(o_{i,t}\mid q,o_{i,<t})} \quad \hat{A}_{i,t}=\frac{R_i-\text{mean}(\{R_i\}_{i=1}^{G})}{\text{std}(\{R_i\}_{i=1}^{G})}.$$

假如对于某个问题，多次采样的结果都正确，会导致所有输出的奖励相同，优势函数等于0，梯度不更新。动态采样机制是指对样本持续采样直到 既有不正确的样本也有正确的样本



(b) The proportion of samples with an accuracy of 1.

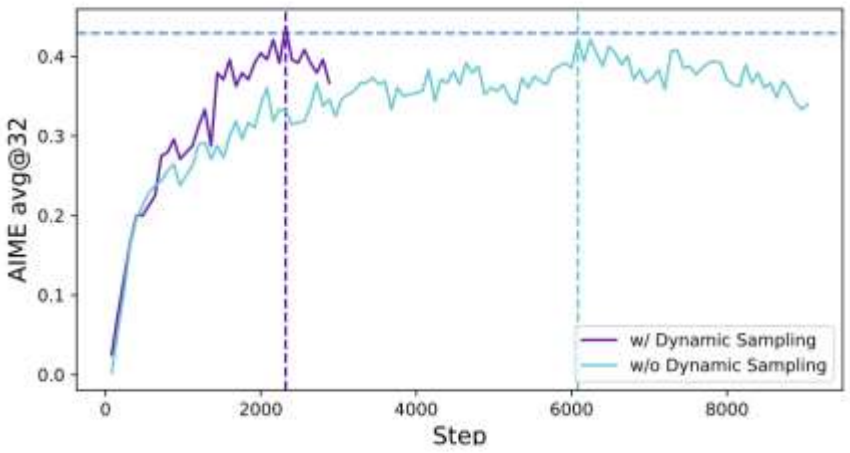
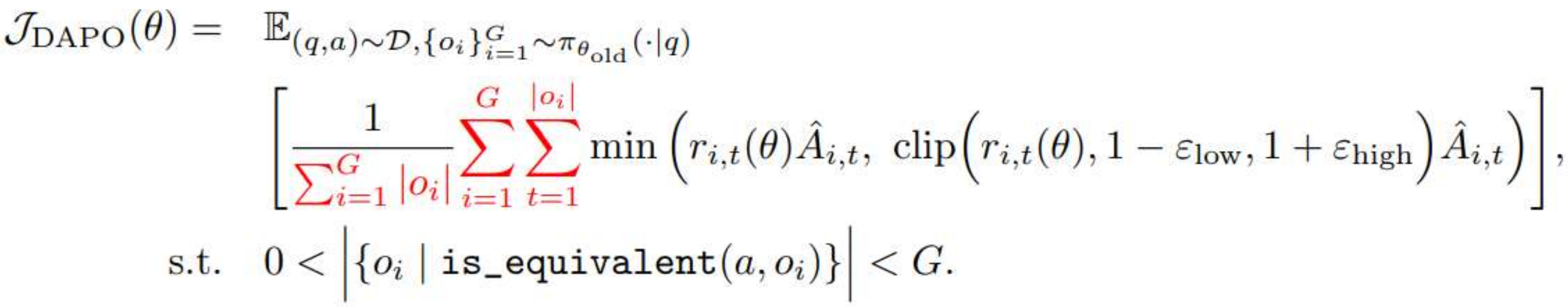


Figure 6 The training progress before and after applying dynamic sampling on a baseline setting.

DAPO：Token-Level Policy Gradient Loss



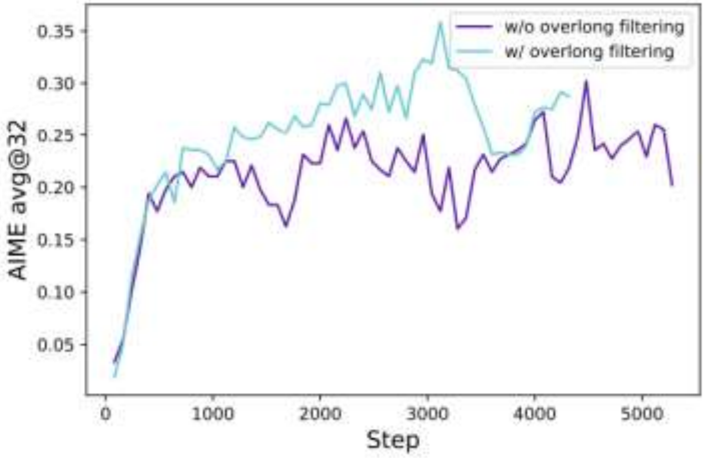
$$\begin{aligned}\mathcal{J}_{\text{DAPO}}(\theta) = \ & \mathbb{E}_{(q,a)\sim\mathcal{D},\{o_i\}_{i=1}^G\sim\pi_{\theta_{\text{old}}}(\cdot|q)} \\ & \left[\frac{1}{\sum_{i=1}^G |o_i|}\sum_{i=1}^G\sum_{t=1}^{|o_i|}\min\left(r_{i,t}(\theta)\hat{A}_{i,t},\ \text{clip}\Big(r_{i,t}(\theta),1-\varepsilon_{\text{low}},1+\varepsilon_{\text{high}}\Big)\hat{A}_{i,t}\right)\right], \\ \text{s.t.} \quad & 0 < \Big|\{o_i \mid \texttt{is\_equivalent}(a,o_i)\}\Big| < G.\end{aligned}$$

使长序列对梯度的影响比短序列更大

DAPO：Overlong Reward Shaping

Soft Overlong Punishment

$$R_{\text{length}}(y) = \begin{cases} 0, & |y| \le L_{\max} - L_{\text{cache}} \\ \frac{(L_{\max} - L_{\text{cache}}) - |y|}{L_{\text{cache}}}, & L_{\max} - L_{\text{cache}} < |y| \le L_{\max} \\ -1, & L_{\max} < |y| \end{cases}$$



(a) Performance on AIME.

DAPO

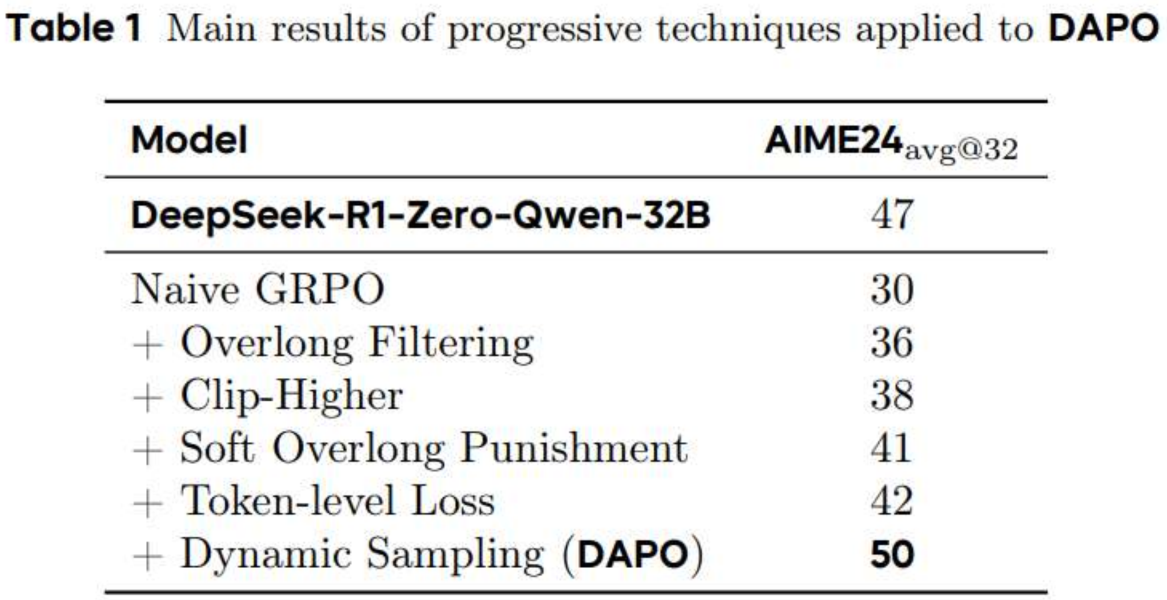


Table 1 Main results of progressive techniques applied to **DAPO**

Model	**AIME24**$_{\text{avg@32}}$
DeepSeek-R1-Zero-Qwen-32B	47
Naive GRPO	30
+ Overlong Filtering	36
+ Clip-Higher	38
+ Soft Overlong Punishment	41
+ Token-level Loss	42
+ Dynamic Sampling (**DAPO**)	**50**