

# DeepSeek-R1 and DAPO

Bae Sun Woo

January 12, 2026

# Table of Contents

1. Preliminary – PPO, GRPO
2. DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning
3. DAPO: An Open-Source LLM Reinforcement Learning System at Scale
4. GDPO: Group reward-Decoupled Normalization Policy Optimization for Multi-reward RL Optimization

# PPO: Proximal Policy Optimization

Actor

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

Critic

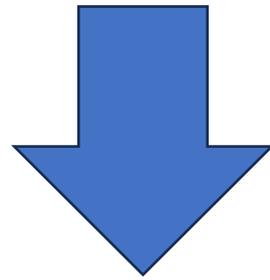
$$\hat{A}_t^{\text{GAE}(\gamma, \lambda)} = \sum_{l=0}^{\infty} (\gamma \lambda)^l \delta_{t+l},$$

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

# PPO: Proximal Policy Optimization

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

High resource demands of maintaining a separate value function (Critic)



**GRPO**

# GRPO: Group Relative Policy Optimization

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

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

No Critic, instead

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

# DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning

**DeepSeek-R1-Zero**

**DeepSeek-R1**

# DeepSeek-R1-Zero

DeepSeek-V3 Base



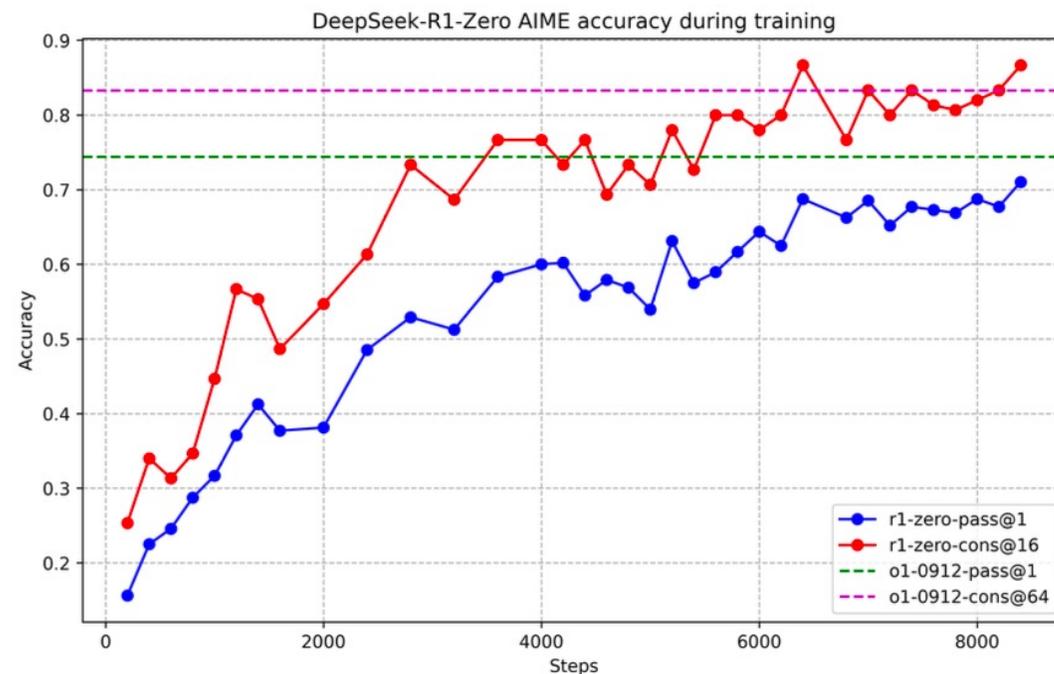
Large Scale RL (GRPO, Rule-based Reward Model)



**DeepSeek-R1-Zero**

# Performance of DeepSeek-R1-Zero

Model	AIME 2024		MATH-500 pass@1	GPQA Diamond pass@1	LiveCode Bench pass@1	CodeForces rating
	pass@1	cons@64				
OpenAI-o1-mini	63.6	80.0	90.0	60.0	53.8	1820
OpenAI-o1-0912	74.4	83.3	94.8	77.3	63.4	1843
DeepSeek-R1-Zero	71.0	86.7	95.9	73.3	50.0	1444



# “AHA moment” of DeepSeek-R1-Zero

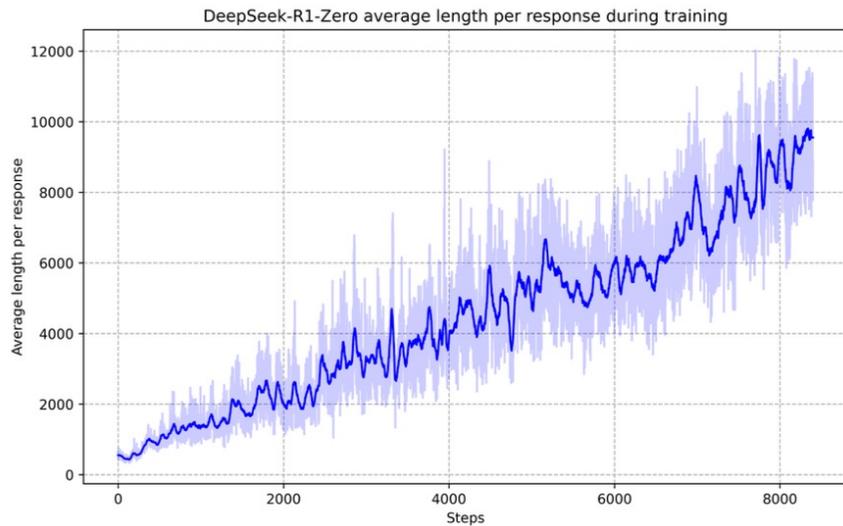


Figure 3 | The average response length of DeepSeek-R1-Zero on the training set during the RL process. DeepSeek-R1-Zero naturally learns to solve reasoning tasks with more thinking time.

---

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 ...

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

Rearrange to isolate the inner square root term:

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

...

**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 ...

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: ...

...

---

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.

# “AHA moment” of DeepSeek-R1-Zero (v2)

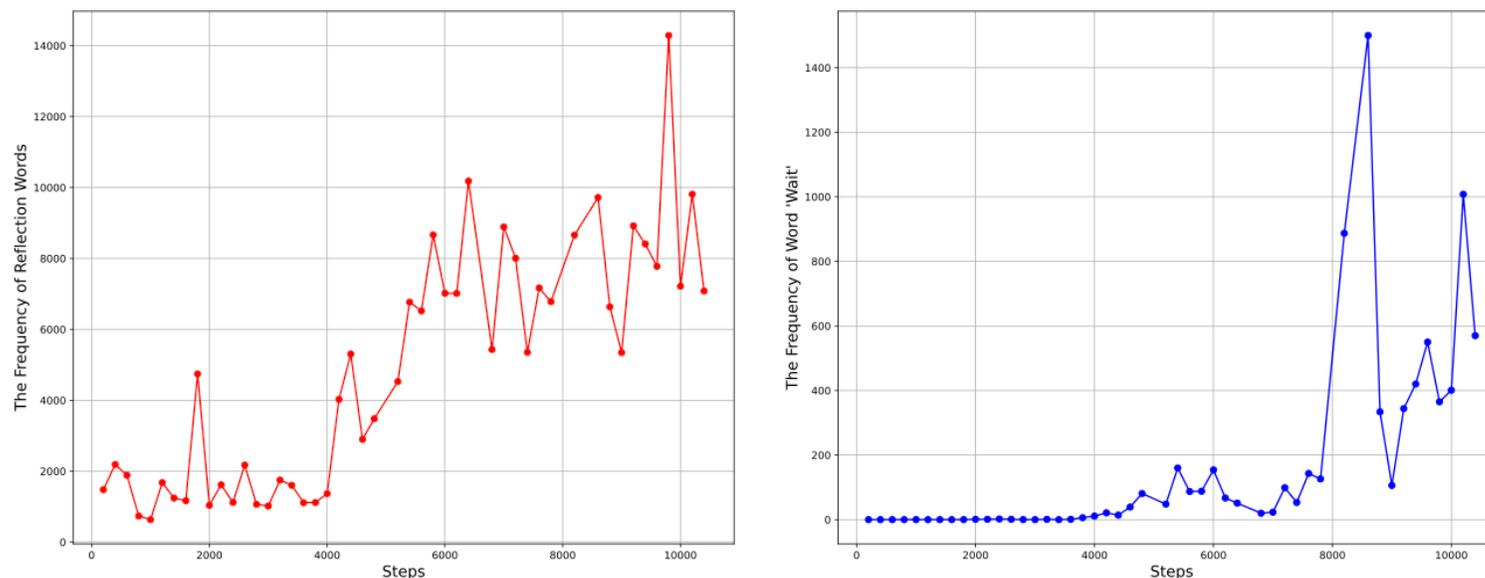
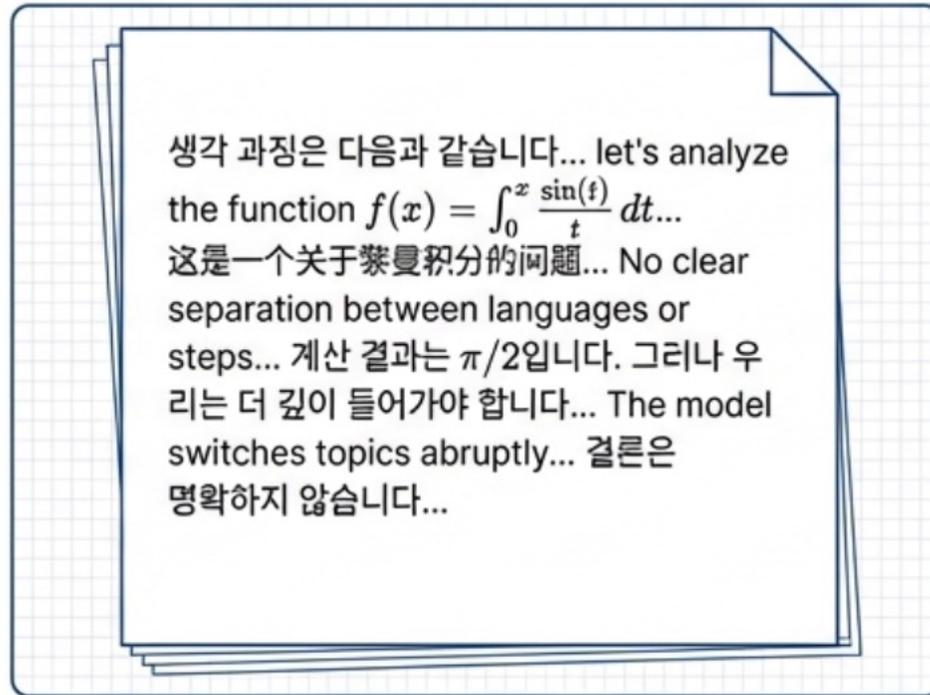


Figure 9 | Evolution of reasoning behaviors during training. (a) Frequency of representative reflective words during the training process; (b) Specific occurrence patterns of the word “wait” throughout the training process.

“wait” depicts the self-correction of model

# Drawback of DeepSeek-R1-Zero



Poor readability, Language Mixing

# DeepSeek-R1

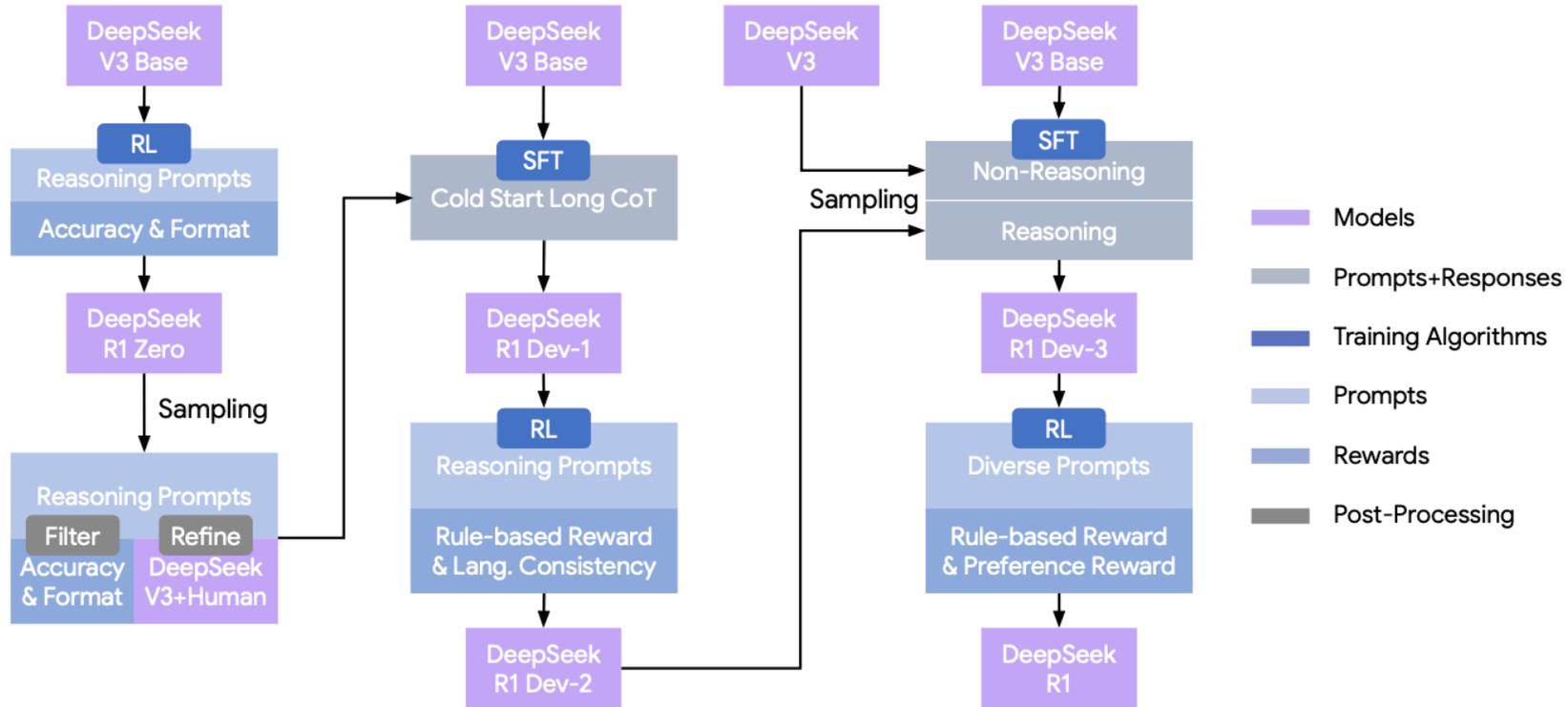
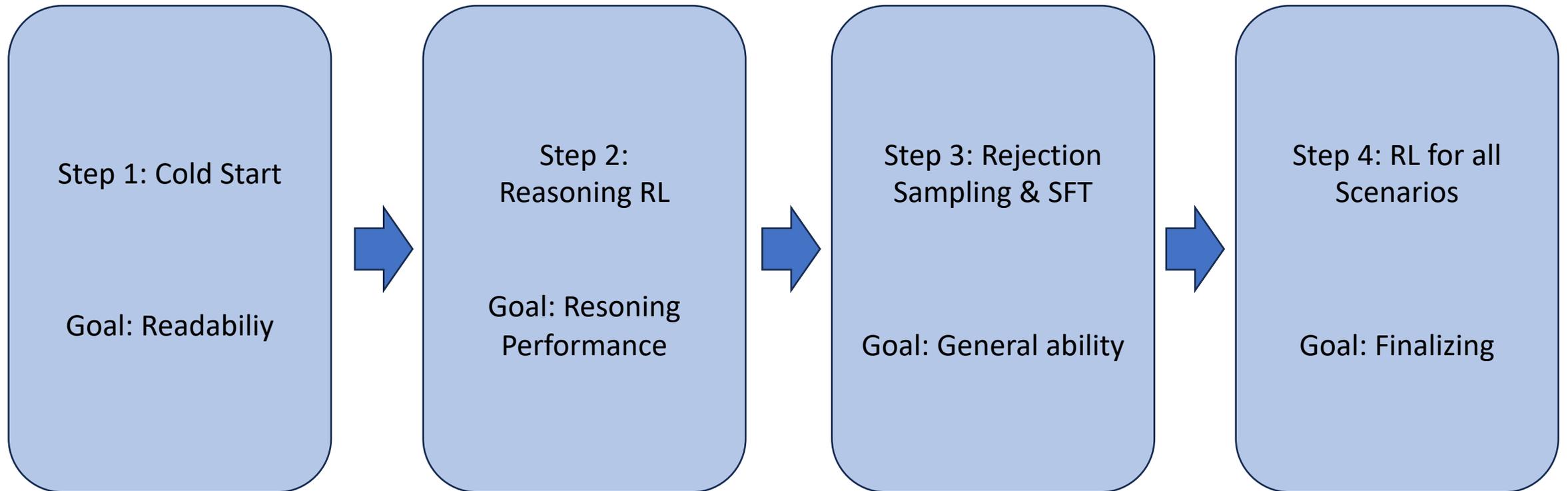


Figure 2 | The multi-stage pipeline of DeepSeek-R1. A detailed background on DeepSeek-V3 Base and DeepSeek-V3 is provided in Supplementary [A.1](#). The models DeepSeek-R1 Dev1, Dev2, and Dev3 represent intermediate checkpoints within this pipeline.

# DeepSeek-R1, Simplified Pipeline



# R1 performance

Benchmark (Metric)	Claude-3.5- Sonnet-1022	GPT-4o 0513	DeepSeek V3	OpenAI o1-mini	OpenAI o1-1217	DeepSeek R1	
Architecture	-	-	MoE	-	-	MoE	
# Activated Params	-	-	37B	-	-	37B	
# Total Params	-	-	671B	-	-	671B	
English	MMLU (Pass@1)	88.3	87.2	88.5	85.2	<b>91.8</b>	90.8
	MMLU-Redux (EM)	88.9	88.0	89.1	86.7	-	<b>92.9</b>
	MMLU-Pro (EM)	78.0	72.6	75.9	80.3	-	<b>84.0</b>
	DROP (3-shot F1)	88.3	83.7	91.6	83.9	90.2	<b>92.2</b>
	IF-Eval (Prompt Strict)	<b>86.5</b>	84.3	86.1	84.8	-	83.3
	GPQA Diamond (Pass@1)	65.0	49.9	59.1	60.0	<b>75.7</b>	71.5
	SimpleQA (Correct)	28.4	38.2	24.9	7.0	<b>47.0</b>	30.1
	FRAMES (Acc.)	72.5	80.5	73.3	76.9	-	<b>82.5</b>
	AlpacaEval2.0 (LC-winrate)	52.0	51.1	70.0	57.8	-	<b>87.6</b>
	ArenaHard (GPT-4-1106)	85.2	80.4	85.5	92.0	-	<b>92.3</b>
Code	LiveCodeBench (Pass@1-COT)	38.9	32.9	36.2	53.8	63.4	<b>65.9</b>
	Codeforces (Percentile)	20.3	23.6	58.7	93.4	<b>96.6</b>	96.3
	Codeforces (Rating)	717	759	1134	1820	<b>2061</b>	2029
	SWE Verified (Resolved)	<b>50.8</b>	38.8	42.0	41.6	48.9	49.2
	Aider-Polyglot (Acc.)	45.3	16.0	49.6	32.9	<b>61.7</b>	53.3
Math	AIME 2024 (Pass@1)	16.0	9.3	39.2	63.6	79.2	<b>79.8</b>
	MATH-500 (Pass@1)	78.3	74.6	90.2	90.0	96.4	<b>97.3</b>
	CNMO 2024 (Pass@1)	13.1	10.8	43.2	67.6	-	<b>78.8</b>
Chinese	CLUEWSC (EM)	85.4	87.9	90.9	89.9	-	<b>92.8</b>
	C-Eval (EM)	76.7	76.0	86.5	68.9	-	<b>91.8</b>
	C-SimpleQA (Correct)	55.4	58.7	<b>68.0</b>	40.3	-	63.7

Table 4 | Comparison between DeepSeek-R1 and other representative models.

# R1 distillation

Model	AIME 2024		MATH-500	GPQA Diamond	LiveCode Bench	CodeForces
	pass@1	cons@64	pass@1	pass@1	pass@1	rating
GPT-4o-0513	9.3	13.4	74.6	49.9	32.9	759
Claude-3.5-Sonnet-1022	16.0	26.7	78.3	65.0	38.9	717
OpenAI-o1-mini	63.6	80.0	90.0	60.0	53.8	<b>1820</b>
QwQ-32B-Preview	50.0	60.0	90.6	54.5	41.9	1316
DeepSeek-R1-Distill-Qwen-1.5B	28.9	52.7	83.9	33.8	16.9	954
DeepSeek-R1-Distill-Qwen-7B	55.5	83.3	92.8	49.1	37.6	1189
DeepSeek-R1-Distill-Qwen-14B	69.7	80.0	93.9	59.1	53.1	1481
DeepSeek-R1-Distill-Qwen-32B	<b>72.6</b>	83.3	94.3	62.1	57.2	1691
DeepSeek-R1-Distill-Llama-8B	50.4	80.0	89.1	49.0	39.6	1205
DeepSeek-R1-Distill-Llama-70B	70.0	<b>86.7</b>	<b>94.5</b>	<b>65.2</b>	<b>57.5</b>	1633

Table 5 | Comparison of DeepSeek-R1 distilled models and other comparable models on reasoning-related benchmarks.

# Distillation vs Reinforcement Learning

Model	AIME 2024		MATH-500	GPQA Diamond	LiveCodeBench
	pass@1	cons@64	pass@1	pass@1	pass@1
<b>QwQ-32B-Preview</b>	50.0	60.0	90.6	54.5	41.9
<b>DeepSeek-R1-Zero-Qwen-32B</b>	47.0	60.0	91.6	55.0	40.2
<b>DeepSeek-R1-Distill-Qwen-32B</b>	<b>72.6</b>	<b>83.3</b>	<b>94.3</b>	<b>62.1</b>	<b>57.2</b>

Table 6 | Comparison of distilled and RL Models on Reasoning-Related Benchmarks.

1. Distillation(powerful model->small model) > RL(small model)
2. Advancing beyond the boundaries of intelligence may still require more powerful base models and larger-scale reinforcement learning

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

Limitation in naive GRPO: entropy collapse, reward noise, training instability.

1. Clip-Higher: promote the diversity of the system and avoids entropy collapse
2. Dynamic Sampling: improve training efficiency and stability
3. Token-Level Policy Gradient Loss: critical in long-CoT RL scenarios
4. Overlong Reward Shaping: reduce reward noise and stabilizes training

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

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

where

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

# Remove KL divergence

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

KL divergence: divergence between the online policy and the frozen reference policy

Useful in the RLHF scenario, but during training the long-CoT reasoning model, the model distribution can diverge significantly from the initial model

=> Exclude KL divergence

# Key Technique 1: Clip-Higher

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

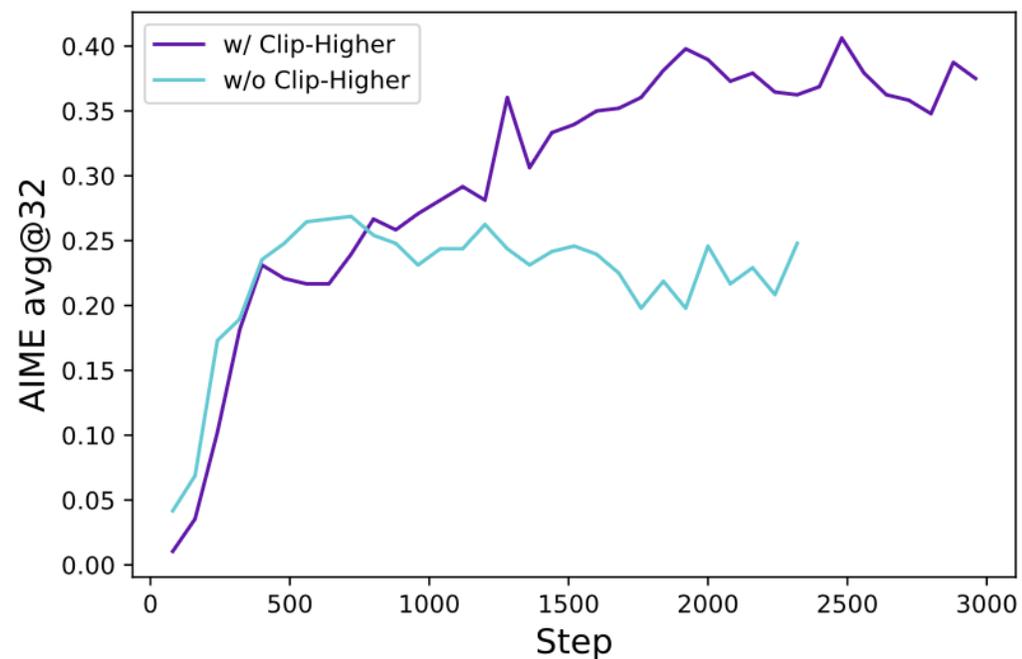


Entropy Collapse phenomenon.  
Upper clip can restrict exploration of the policy,  
Making exploitation more probable

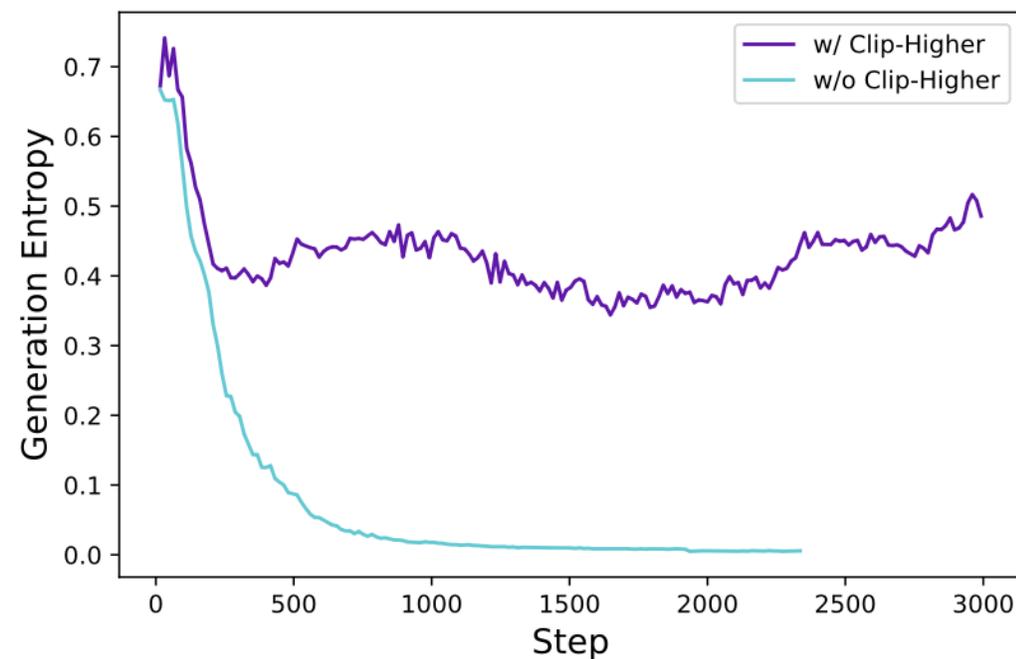
$$\mathcal{J}_{\text{DAPO}}(\theta) = \mathbb{E}_{(q,a) \sim \mathcal{D}, \{o_i\}_{i=1}^G \sim \pi_{\theta_{\text{old}}}(\cdot|q)} \left[ \frac{1}{\sum_{i=1}^G |o_i|} \sum_{i=1}^G \sum_{t=1}^{|o_i|} \min \left( r_{i,t}(\theta) \hat{A}_{i,t}, \text{clip} \left( r_{i,t}(\theta), 1 - \varepsilon_{\text{low}},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

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

# Key Technique 1: Clip-Higher



**(a)** Accuracies on AIME.



**(b)** Entropy of actor model.

# Key Technique 2: Dynamic Sampling

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

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

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

Advantage becomes 0 if all outputs are correct or incorrect.

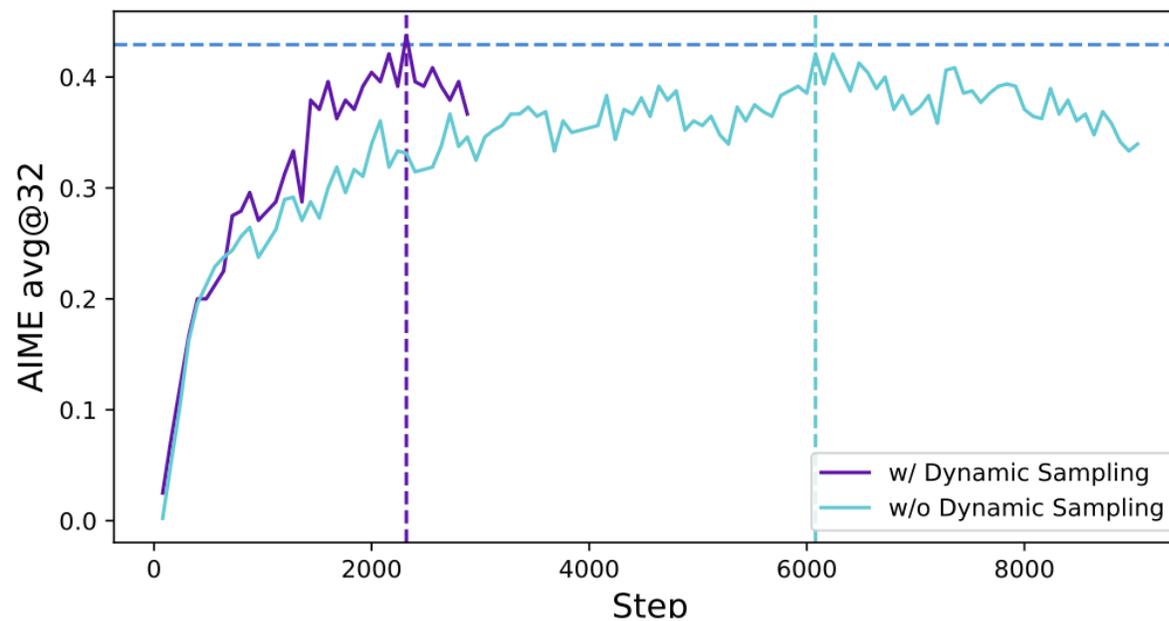


Zero policy gradient



Degrade Sample Efficiency

# Key Technique 2: Dynamic Sampling



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

# Key Technique 3: Token-Level Policy Gradient Loss

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

$$\left[ \frac{1}{G} \sum_{i=1}^G \frac{1}{|o_i|} \sum_{t=1}^{|o_i|} \left( \min \left( r_{i,t}(\theta) \hat{A}_{i,t}, \text{clip} \left( r_{i,t}(\theta), 1 - \varepsilon, 1 + \varepsilon \right) \hat{A}_{i,t} \right) - \beta D_{\text{KL}}(\pi_{\theta} \parallel \pi_{\text{ref}}) \right) \right]$$

Divided by response length



Same weight for all responses



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

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

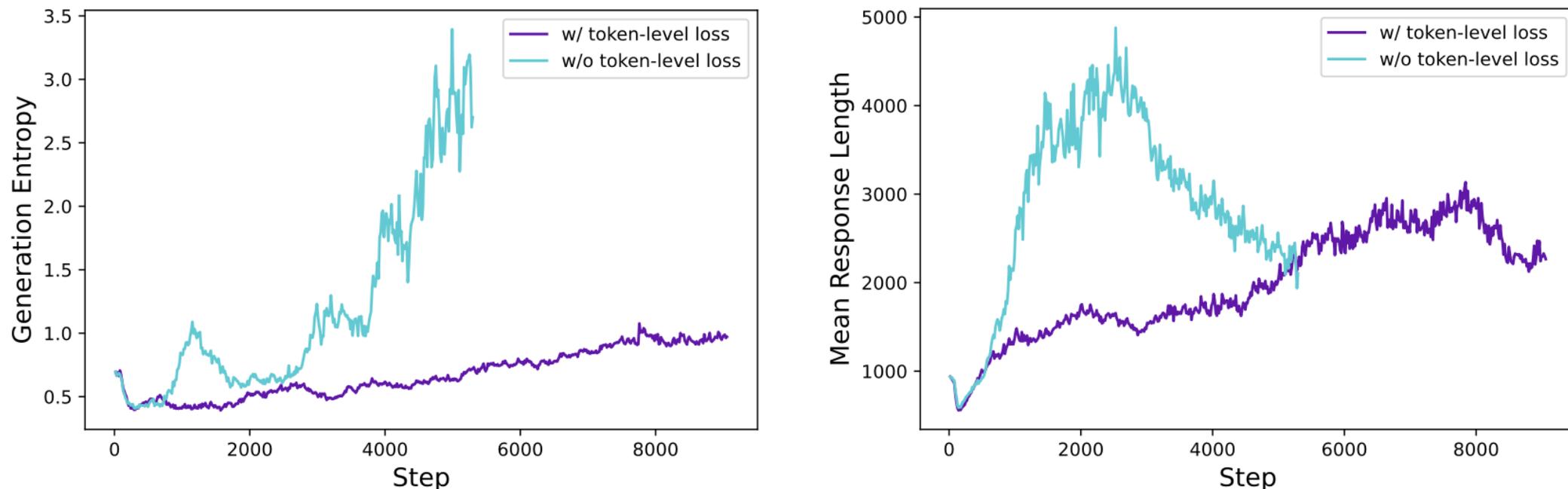
$$\text{s.t. } 0 < \left| \{o_i \mid \text{is\_equivalent}(a, o_i)\} \right| < G.$$

Divided by total token usages



More weight on longer response

# Key Technique 3: Token-Level Policy Gradient Loss



(a) Entropy of actor model's generation probabilities.

(b) Average length of actor model-generated responses

**Figure 4** The entropy of the probability distribution of the actor model, as well as the changes in response length.

# Key Technique 4: Overlong Reward Shaping

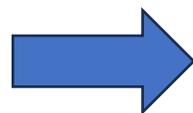
In RL training, we typically set a **maximum length** for generation, with **overlong samples truncated** accordingly

Default:

Truncated Samples



Punitive Rewards



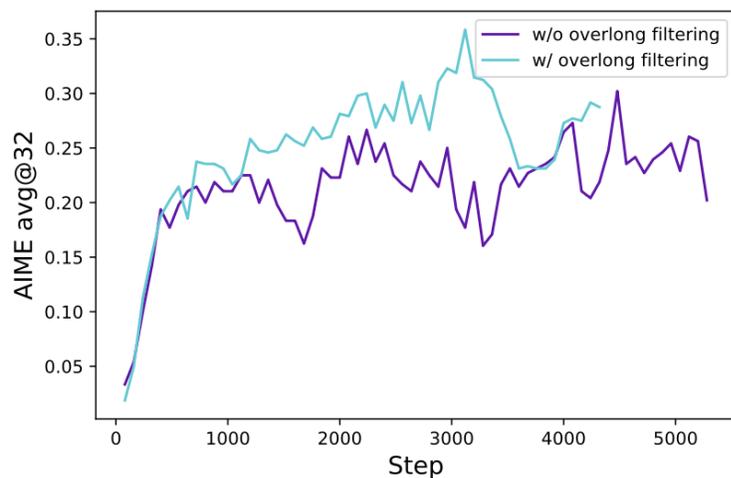
(Reward Noise)

Sound reasoning process  
might be penalized due to  
excessive length

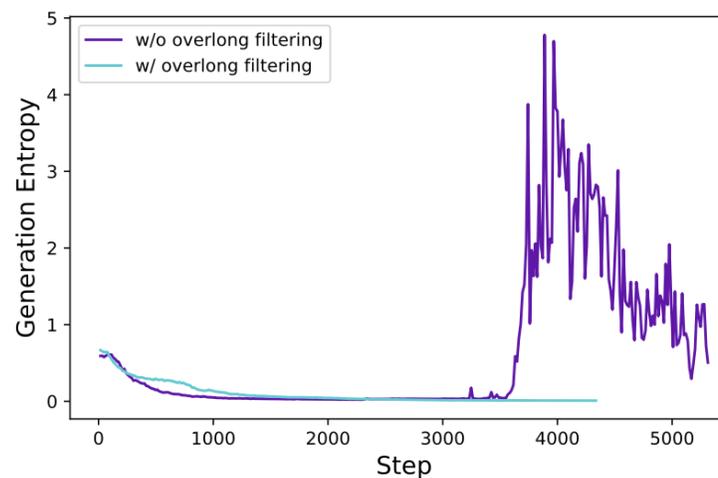
1. Overlong Filtering
2. Soft Overlong Punishment

# Key Technique 4-1: Overlong Filtering

“Masks the loss of truncated samples”



(a) Performance on AIME.



(b) Entropy of actor model.

**Figure 5** The accuracy of the actor model on AIME and the entropy of its generation probabilities, both before and after applying **Overlong Reward Shaping** strategy.

# Key Technique 4-2: Soft Overlong Punishment

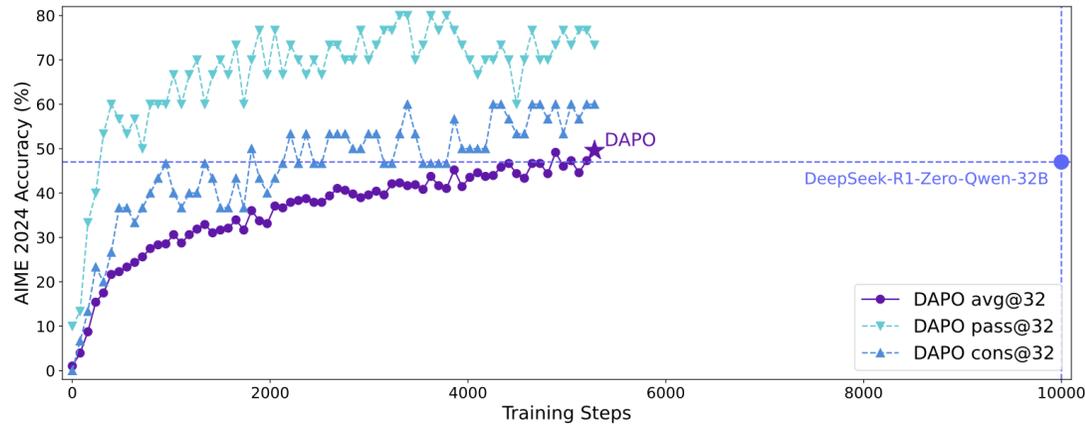
Length-aware penalty mechanism

$$R_{\text{length}}(y) = \begin{cases} 0, & |y| \leq L_{\text{max}} - L_{\text{cache}} \\ \frac{(L_{\text{max}} - L_{\text{cache}}) - |y|}{L_{\text{cache}}}, & L_{\text{max}} - L_{\text{cache}} < |y| \leq L_{\text{max}} \\ -1, & L_{\text{max}} < |y| \end{cases} \quad \rightarrow \text{Punishment increases linearly in this interval}$$

$L_{\text{max}}$ : 16,384 tokens,  $L_{\text{cache}}$ : 4,096 tokens

$$R(y) = R_{\text{acc}}(y) + R_{\text{length}}(y)$$

# Main Results of DAPO



**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.

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

Model	AIME24 <sub>avg@32</sub>
<b>DeepSeek-R1-Zero-Qwen-32B</b>	47
Naive GRPO	30
+ Overlong Filtering	36
+ Clip-Higher	38
+ Soft Overlong Punishment	41
+ Token-level Loss	42
+ Dynamic Sampling ( <b>DAPO</b> )	<b>50</b>

# Main Results of DAPO

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

Model	AIME24 <sub>avg@32</sub>
<b>DeepSeek-R1-Zero-Qwen-32B</b>	47
Naive GRPO	30
+ Overlong Filtering	36
+ Clip-Higher	38
+ Soft Overlong Punishment	41
+ Token-level Loss	42
+ Dynamic Sampling ( <b>DAPO</b> )	<b>50</b>

Model	AIME 2024		MATH-500	GPQA Diamond	LiveCodeBench
	pass@1	cons@64	pass@1	pass@1	pass@1
QwQ-32B-Preview	50.0	60.0	90.6	54.5	41.9
DeepSeek-R1-Zero-Qwen-32B	47.0	60.0	91.6	55.0	40.2
DeepSeek-R1-Distill-Qwen-32B	72.6	83.3	94.3	62.1	57.2

Table 6 | Comparison of distilled and RL Models on Reasoning-Related Benchmarks.

# GDPO: Group reward-Decoupled Normalization Policy Optimization for Multi-reward RL Optimization

GRPO

$$r_{\text{sum}}^{(i,j)} = r_1^{(i,j)} + \dots + r_n^{(i,j)} \quad A_{\text{sum}}^{(i,j)} = \frac{r_{\text{sum}}^{(i,j)} - \text{mean}\{r_{\text{sum}}^{(i,1)}, \dots, r_{\text{sum}}^{(i,G)}\}}{\text{std}\{r_{\text{sum}}^{(i,1)}, \dots, r_{\text{sum}}^{(i,G)}\}}$$

$$r_{\text{sum}} = w_1 r_1 + \dots + w_n r_n$$

GDPO

$$A_1^{(i,j)} = \frac{r_1^{(i,j)} - \text{mean}\{r_1^{(i,1)}, \dots, r_1^{(i,G)}\}}{\text{std}\{r_1^{(i,1)}, \dots, r_1^{(i,G)}\}}, \quad \dots, \quad A_n^{(i,j)} = \frac{r_n^{(i,j)} - \text{mean}\{r_n^{(i,1)}, \dots, r_n^{(i,G)}\}}{\text{std}\{r_n^{(i,1)}, \dots, r_n^{(i,G)}\}}$$

$$A_{\text{sum}}^{(i,j)} = A_1^{(i,j)} + \dots + A_n^{(i,j)} \quad \longrightarrow \quad A_{\text{sum}}^{(i,j)} = w_1 A_1^{(i,j)} + \dots + w_n A_n^{(i,j)}$$

$$\hat{A}_{\text{sum}}^{(i,j)} = \frac{A_{\text{sum}}^{(i,j)} - \text{mean}\{A_{\text{sum}}^{(i',j')} \mid i' \in D_{\text{Batch}}, j' = 1, \dots, G\}}{\text{std}\{A_{\text{sum}}^{(i',j')} \mid i' \in D_{\text{Batch}}, j' = 1, \dots, G\} + \epsilon}$$

# GDPO: Group reward-Decoupled Normalization Policy Optimization for Multi-reward RL Optimization



Figure 9 | Training behavior of GRPO and GDPO when optimizing DeepSeek-R1-7B across correctness reward, length reward, and maximum batch response length on math reasoning data. We can see that GDPO maintains improving correctness and better adherence to length constraints over GRPO.



Figure 10 | Training behavior of GRPO and GDPO when optimizing Qwen3-4B-Instruct across correctness reward, length reward, and maximum batch response length on math reasoning data. We can see that GDPO maintains improving correctness and better adherence to length constraints over GRPO.

RQ: Effective Reward Model for Open-Ended Data

RQ: Effective Reward Automatic Weighting Strategy