



南方科技大学
SOUTHERN UNIVERSITY OF SCIENCE AND TECHNOLOGY

Advanced Natural Language Processing

Lecture 21: Reasoning LLMs



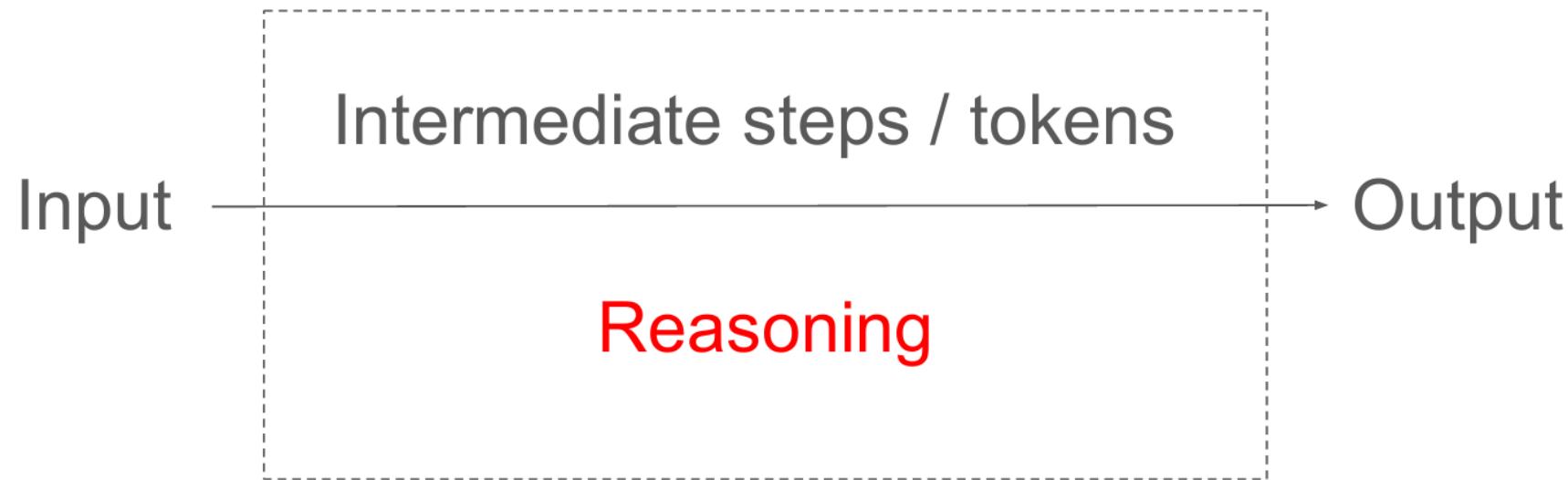
陈冠华 CHEN Guanhua

Department of Statistics and Data Science

What is LLM Reasoning?



南方科技大学
SOUTHERN UNIVERSITY OF SCIENCE AND TECHNOLOGY



Ling et al. Program Induction by Rationale Generation: Learning to Solve and Explain Algebraic Word Problems. ACL 2017

Using intermediate steps or program traces was widely adopted in neural-symbolic literature, e.g., neural-symbolic stack machines (NeurIPS 2020) by Compositional generalization via Chen et al.

Stanford, CS25: Transformers United V5, April 29, 2025

What is LLM Reasoning?



南方科技大学
SOUTHERN UNIVERSITY OF SCIENCE AND TECHNOLOGY

What is the output when concatenating the last letter of each word in “artificial intelligence”?

No reasoning

The answer is “le”.

Reasoning

The last letter of “artificial” is “l”. The last letter of “intelligence” is “e”. Concatenating “l” and “e” leads to “le”. So the answer is “le”.



What is LLM Reasoning?



南方科技大学
SOUTHERN UNIVERSITY OF SCIENCE AND TECHNOLOGY

Problem 1:

Question: Two trains running in opposite directions cross a man standing on the platform in 27 seconds and 17 seconds respectively and they cross each other in 23 seconds. The ratio of their speeds is:

Options: A) 3/7 B) 3/2 C) 3/88 D) 3/8 E) 2/2

Rationale: Let the speeds of the two trains be x m/sec and y m/sec respectively. Then, length of the first train = $27x$ meters, and length of the second train = $17y$ meters. $(27x + 17y) / (x + y) = 23 \rightarrow 27x + 17y = 23x + 23y \rightarrow 4x = 6y \rightarrow x/y = 3/2$.

Correct Option: B

What is LLM Reasoning?



南方科技大学
SOUTHERN UNIVERSITY OF SCIENCE AND TECHNOLOGY

GSM8K: <Problem, Intermediate Steps, Answer>

Following the work by Ling et al 2017, Cobbe et al 2021 in OpenAI built a much larger math word problem dataset (GSM8K) with natural language rationales, and used it to finetune GPT3

Problem: Ali is a dean of a private school where he teaches one class. John is also a dean of a public school. John has two classes in his school. Each class has $1/8$ the capacity of Ali's class which has the capacity of 120 students. What is the combined capacity of both schools?

Solution: Ali's class has a capacity of 120 students. Each of John's classes has a capacity of $120/8 = 15$ students. The total capacity of John's two classes is $15 \text{ students} * 2 \text{ classes} = 30 \text{ students}$. The combined capacity of the two schools is $120 \text{ students} + 30 \text{ students} = 150 \text{ students}$.

Final answer: 150



Cobbe et al. Training Verifiers to Solve Math Word Problems. [arXiv:2110.14168](https://arxiv.org/abs/2110.14168) [cs.LG]. 2021

Reasoning LLMs



南方科技大学
SOUTHERN UNIVERSITY OF SCIENCE AND TECHNOLOGY

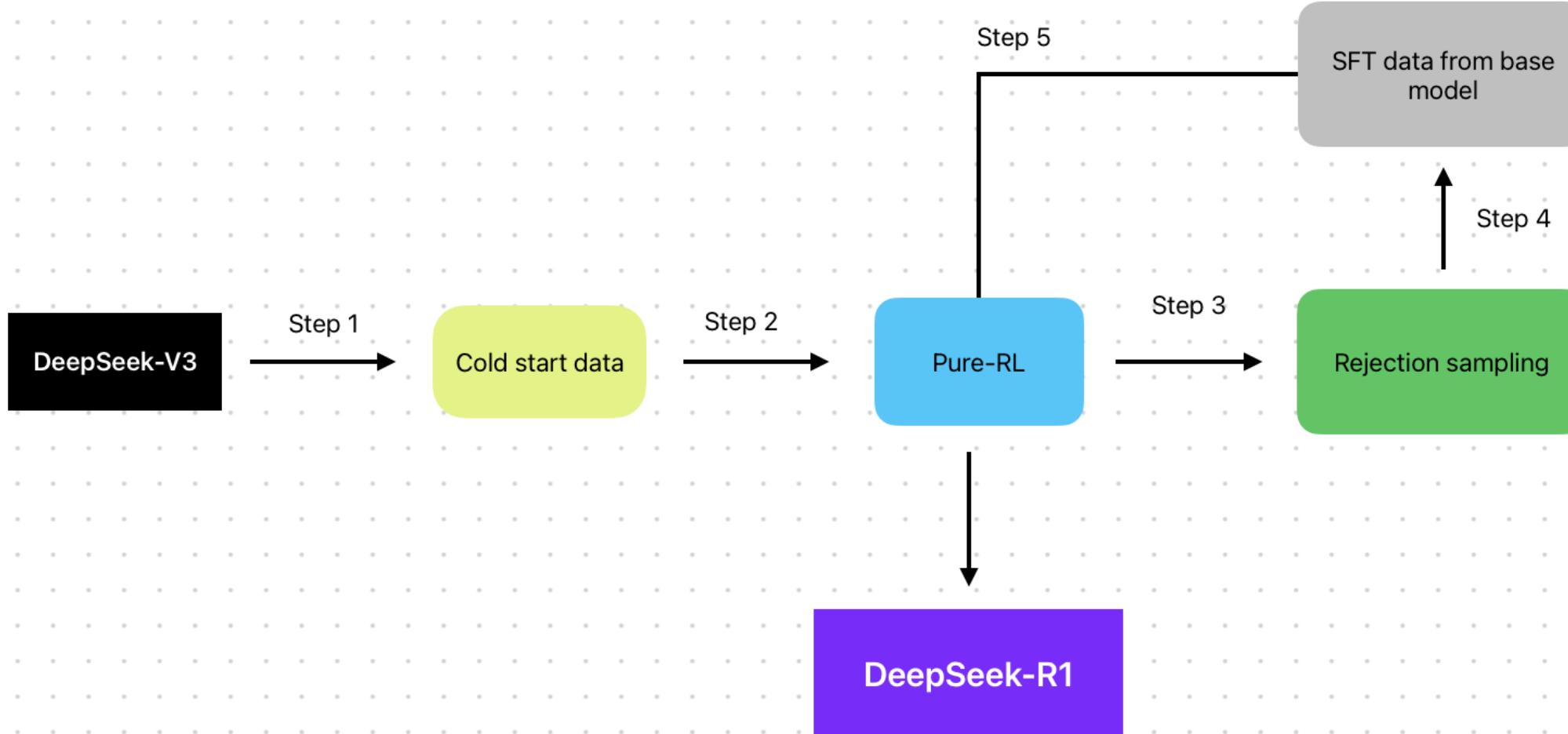


DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning

DeepSeek-AI

research@deepseek.com

Training DeepSeek-R1



Group Relative Policy Optimization



- GRPO algorithm is based on the Proximal Policy Optimization (PPO) algorithm.
- Reward model is a trained model that assigns rewards.
- Value model estimates the long-term expected future reward from a given state.

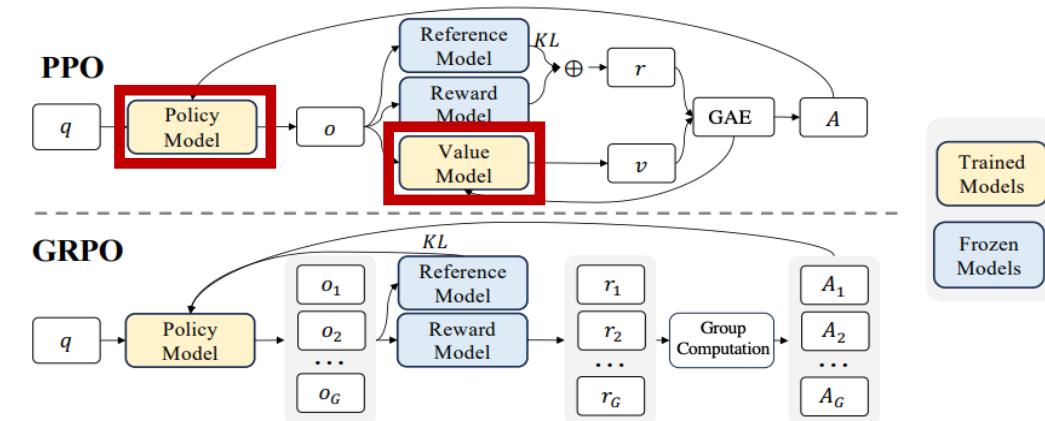


Figure 4 | Demonstration of PPO and our GRPO. GRPO foregoes the value model, instead estimating the baseline from group scores, significantly reducing training resources.

Group Relative Policy Optimization



- GRPO generates multiple responses at once and generates multiple rewards.
- Group computation decides the advantage of the response based on other responses in the group.
- The value model is removed since it's expensive to evaluate multiple actions every state.

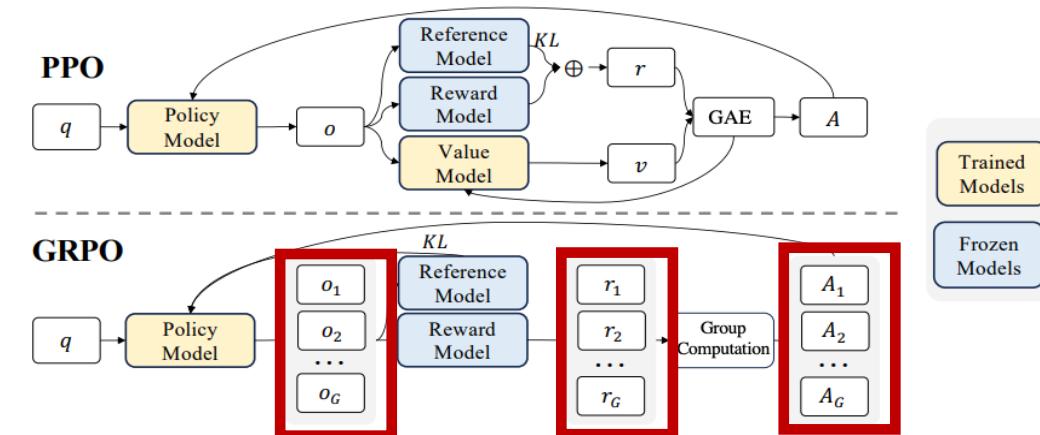


Figure 4 | Demonstration of PPO and our GRPO. GRPO foregoes the value model, instead estimating the baseline from group scores, significantly reducing training resources.

Group Relative Policy Optimization



- Optimizes the policy to maximize the objective JGRPO.
- For every question (q), the model generates multiple outputs (o_i).
- Scale the reward by the ratio or clipped ratio (conservative updates).
- Maintain divergence from the ref model (base model).

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

$$\begin{aligned} \mathcal{J}_{GRPO}(\theta) &= \mathbb{E}_{[q \sim P(Q), \{o_i\}_{i=1}^G \sim \pi_{\theta_{old}}(O|q)]} \\ &\quad \frac{1}{G} \sum_{i=1}^G \left(\min \left(\frac{\pi_\theta(o_i|q)}{\pi_{\theta_{old}}(o_i|q)} A_i, \text{clip} \left(\frac{\pi_\theta(o_i|q)}{\pi_{\theta_{old}}(o_i|q)}, 1 - \varepsilon, 1 + \varepsilon \right) A_i \right) - \beta \mathbb{D}_{KL} (\pi_\theta || \pi_{ref}) \right), \end{aligned} \quad (1)$$

Reward Modeling in GRPO



南方科技大学
SOUTHERN UNIVERSITY OF SCIENCE AND TECHNOLOGY

- Uses a rule-based reward system with two types:
 - Accuracy rewards: Ensures correct responses (e.g., math, LeetCode).
 - Format rewards: Employ a format reward model that enforces the model to put its thinking process between '<think>' and '</think>' tags.

DeepSeek-R1-Zero Response Length

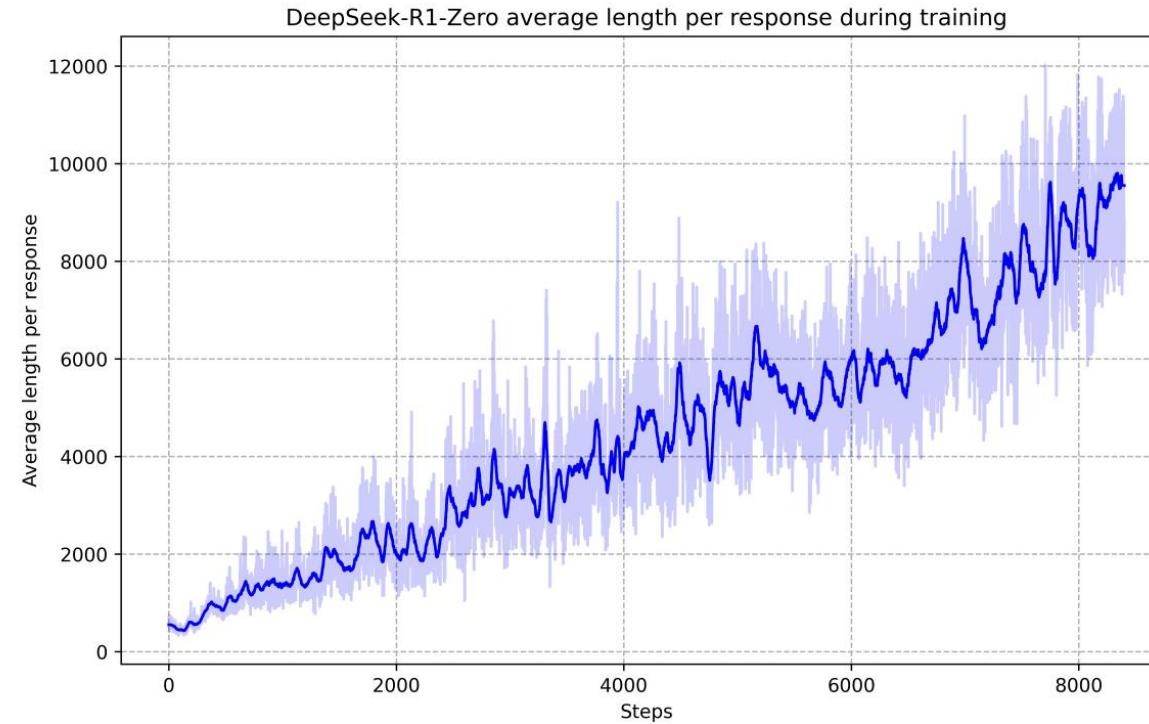


Figure 3 | The average response length of DeepSeek-R1-Zero on the training set during the RL process. DeepSeek-R1-Zero naturally learns to solve reasoning tasks with more thinking time.

DeepSeek-R1-Zero



Model	AIME 2024		MATH-500	GPQA Diamond	LiveCode Bench	CodeForces
	pass@1	cons@64	pass@1	pass@1	pass@1	rating
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

Table 2 | Comparison of DeepSeek-R1-Zero and OpenAI o1 models on reasoning-related benchmarks.

Aha Moment of DeepSeek-R1-Zero



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 \dots

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

Rearrange to isolate the inner square root term:

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

\dots

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 \dots

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: \dots

\dots

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.

Drawback of DeepSeek-R1-Zero



- DeepSeek-R1-Zero struggles with challenges like
 - poor readability, and
 - language mixing.
- “To make reasoning processes more readable, we explore DeepSeek-R1, a method that utilizes RL with human-friendly cold-start data.”

DeepSeek-R1 Achieves SOTA Performance



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

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

DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning
<https://github.com/alaaj27/DeepSeek-R1-slides>

Distilled Model Evaluation



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	1820
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	72.6	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	86.7	94.5	65.2	57.5	1633

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

Takeaways



- DeepSeek-R1 matches OpenAI o1 in complex reasoning tasks.
- DeepSeek-R1-zero achieves strong logical reasoning without supervised fine-tuning.
- Distillation enables smaller models to inherit advanced reasoning capabilities from DeepSeek-R1.

Chain-of-Thought Decoding

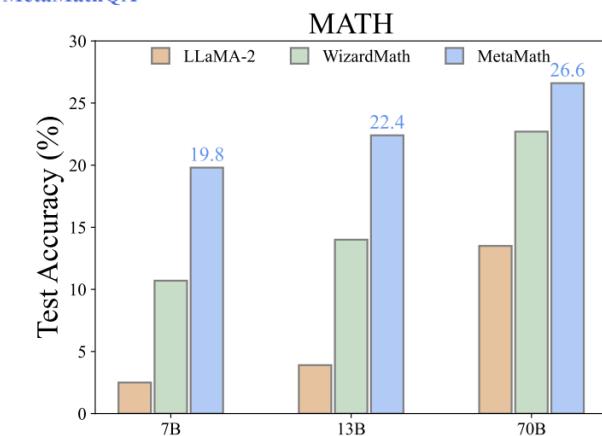
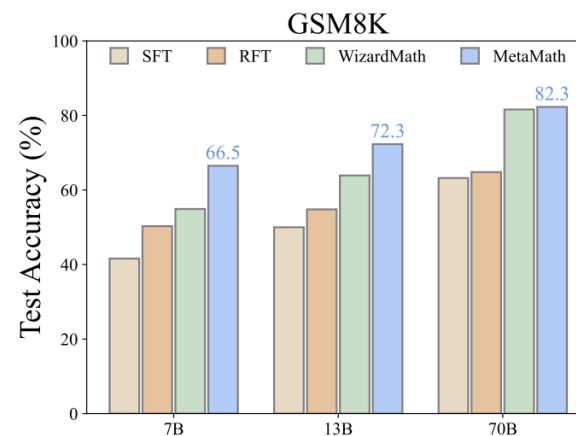
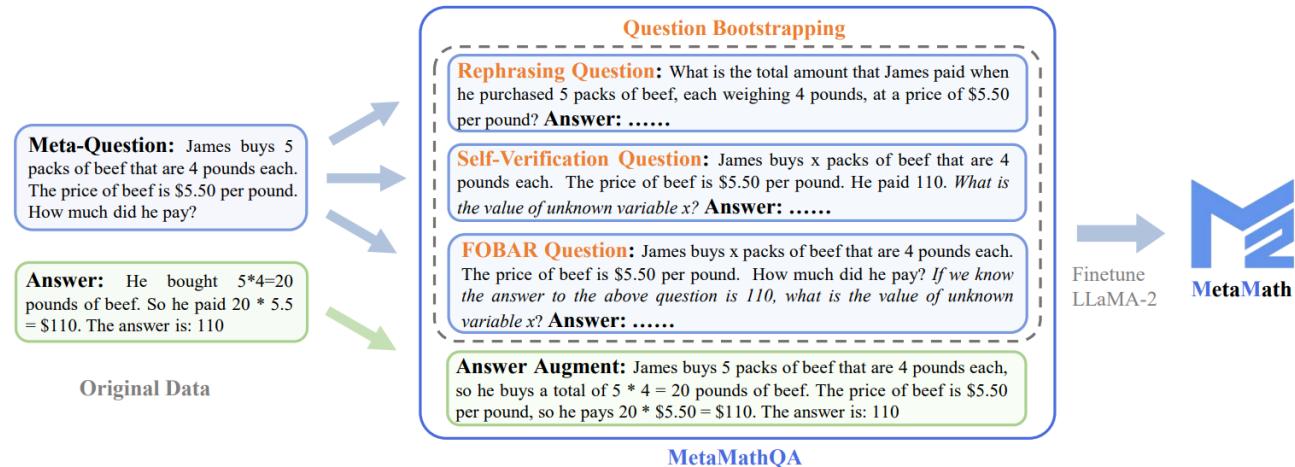
-
1. Go beyond greedy decoding by checking more generation candidates
 2. Choose candidates which have the highest confidence on the final answer
-

- How can we train a model to learn cot reason?
 - 1. High-quality reasoning data
 - 2. Novelty Training Methods

High-quality reasoning data



- MetaMath: Bootstrap Your Own Mathematical Questions for Large Language Models (ICLR 2024 Spotlight)
- MetaMathQA expands beyond traditional answer augmentation to generate diverse question formulations that capture the same mathematical concepts from multiple perspectives.

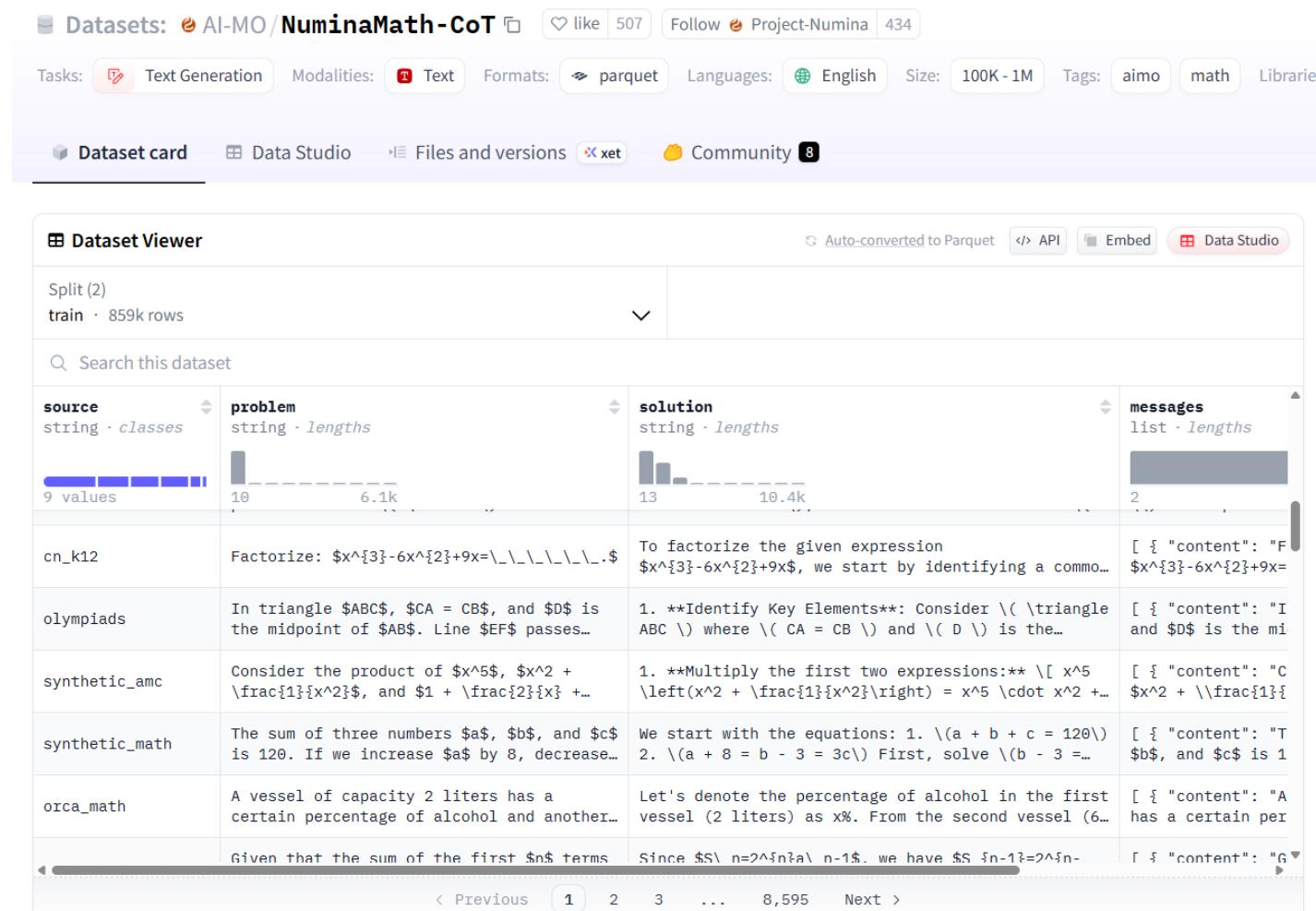


High-quality reasoning data



AI-MO/NuminaMath-CoT

- Approximately 860k math problems, where each solution is formatted in a Chain of Thought (CoT) manner.
- The processing steps include
 - (a) OCR from the original PDFs,
 - (b) segmentation into problem-solution pairs,
 - (c) Translation into English,
 - (d) realignment to produce a CoT reasoning format, and
 - (e) final answer formatting.



R1-Distill Dataset



zwhe99/DeepMath-103K

- DeepMath-103K is meticulously curated to push the boundaries of mathematical reasoning in language models. Key features include:
 - 1. Challenging Problems
 - 2. Data Diversity and Novelty

Datasets: zwhe99/DeepMath-103K like 274

Tasks: Text Generation Modalities: Text Formats: parquet Languages: English Size: 100K - 1M ArXiv: arxiv:2504.11456 Tag

Libraries: Datasets Dask Croissant +1 License: mit

Dataset card Data Studio Files and versions xet Community 4

Dataset Viewer Auto-converted to Parquet API Embed Data Studio

Split (1)
train · 103k rows

Search this dataset

question	final_answer	difficulty	topic	r1_solution_1	r1_solution_2
$\lim_{x \rightarrow \infty} \dots$	\infty	5	Mathematics -> ...	"Okay, so I need to find the limit as x approaches infinity of (x!) raised to the power of 1/x."	"Okay, so I need to find the limit as x approaches infinity of (x!) raised to the power of 1/x."
"Find the length of the polar curve..."	$\frac{\pi}{2}$	5	Mathematics -> Calculus...	"Okay, so I need to find the length of the polar curve given by r = \sqrt{x^2 + y^2} from theta = 0 to pi/2."	"Okay, so I need to find the length of the polar curve given by r = \sqrt{x^2 + y^2} from theta = 0 to pi/2."
"Let \$A\$ be a proper infinite..."	A	3	Mathematics -> Algebra ...	"Okay, let's try to figure out what A is..."	"Okay, so I have this problem here: Let A be a proper infinite subset of a set X..."
"Consider a regular pentagon..."	ϕ^4	5.5	Mathematics -> Geometry...	"Okay, so I have this problem about a regular pentagon circumscribed in a circle..."	"Okay, so I have this problem about a regular pentagon circumscribed in a circle..."
"Find the longest geometric..."	6	4	Mathematics -> Algebra ...	"Okay, so I need to find the longest geometric progression (GP) with a common ratio of 2..."	"Okay, so I need to find the longest geometric progression (GP) with a common ratio of 2..."
"Find the expectation of X_t..."	0	8	Mathematics -> Applied...	"Okay, so I need to find the expectation of X_t, where X_t = sin(B_t)..."	"Okay, so I need to find the expectation of X_t, where X_t = sin(B_t)..."

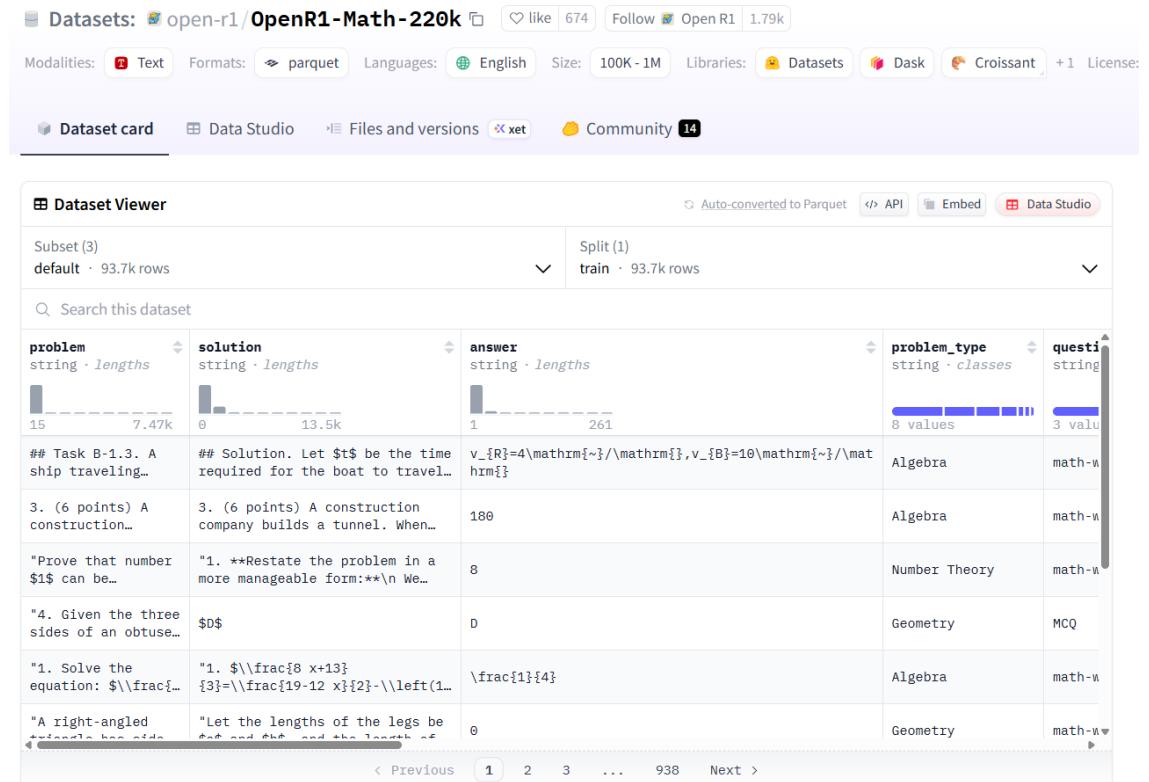
End of preview. Expand in Data Studio

< Previous 1 2 3 ... 1,031 Next >

R1-Distill Dataset

open-r1/OpenR1-Math-220k

- OpenR1-Math-220k is a large-scale dataset for mathematical reasoning.
- It consists of 220k math problems with two to four reasoning traces generated by [DeepSeek R1](#) for problems from NuminMath 1.5.
- The traces were verified using [Math Verify](#) for most samples and [Llama-3.3-70B-Instruct](#) as a judge for 12% of the samples, and each problem contains at least one reasoning trace with a correct answer.



Training Methods



南方科技大学
SOUTHERN UNIVERSITY OF SCIENCE AND TECHNOLOGY

- What problems exist with the current SFT?

$$\min_{\theta} \mathcal{L}_{\text{CE}}(\theta) = -\mathbb{E}_{x \sim \rho} \mathbb{E}_{y \sim p(\cdot|x)} [\log f_{\theta}(y|x)].$$

- 1. SFT treats all tokens equally.
- 2. SFT reduces output diversity.

- Preserving Diversity in Supervised Fine-Tuning of Large Language Models. ICLR 2025.

Question: Marissa is hiking a 12-mile trail. She took 1 hour to walk the first 4 miles, then another hour to walk the next two miles. If she wants her average speed to be 4 miles per hour, what speed (in miles per hour) does she need to walk the remaining distance?

Answer: ## 6



Figure 1: Illustration of diversity preservation in SFT. While pre-trained LLMs produce diverse outputs, these often lack proper formatting. Standard SFT using CE improves readability but reduces diversity. We aim to maintain output diversity while enhancing the readability of LLMs' responses.

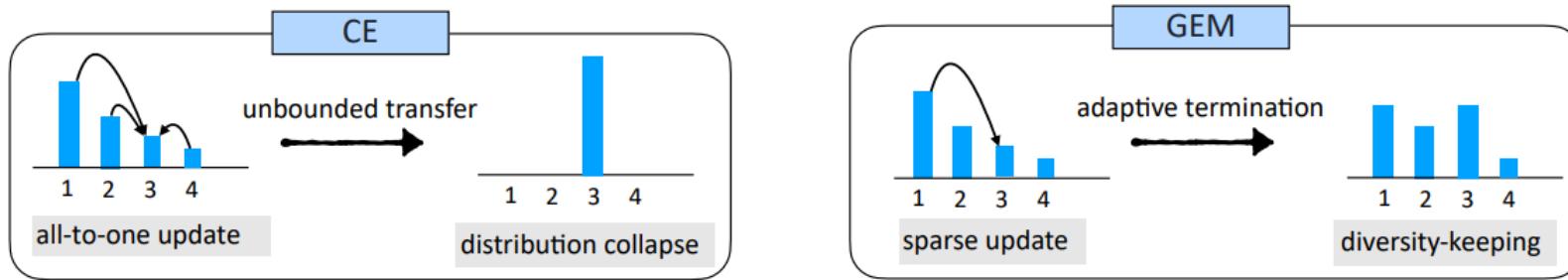


Figure 2: Comparison of learning schemes: CE v.s. GEM ($\beta = 0$). The arrows illustrate the probability movement directions during the learning process, with Token 3 as the target token.

(GEM)

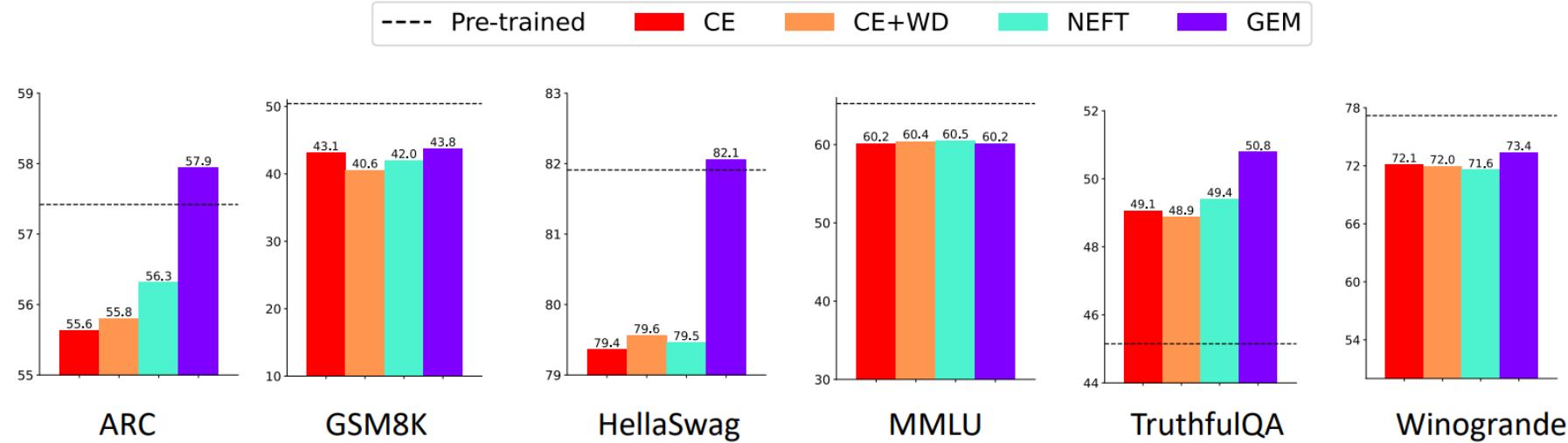
While the target token $i \notin \text{argmax } f_{\theta_k}(\cdot|x)$, continue the following steps.

- Calculate the model's best prediction $j = \text{argmax } f(\cdot|x)$
- Decrease the logit for source token j by learning rate η and weight $w_{i \leftarrow j}$:

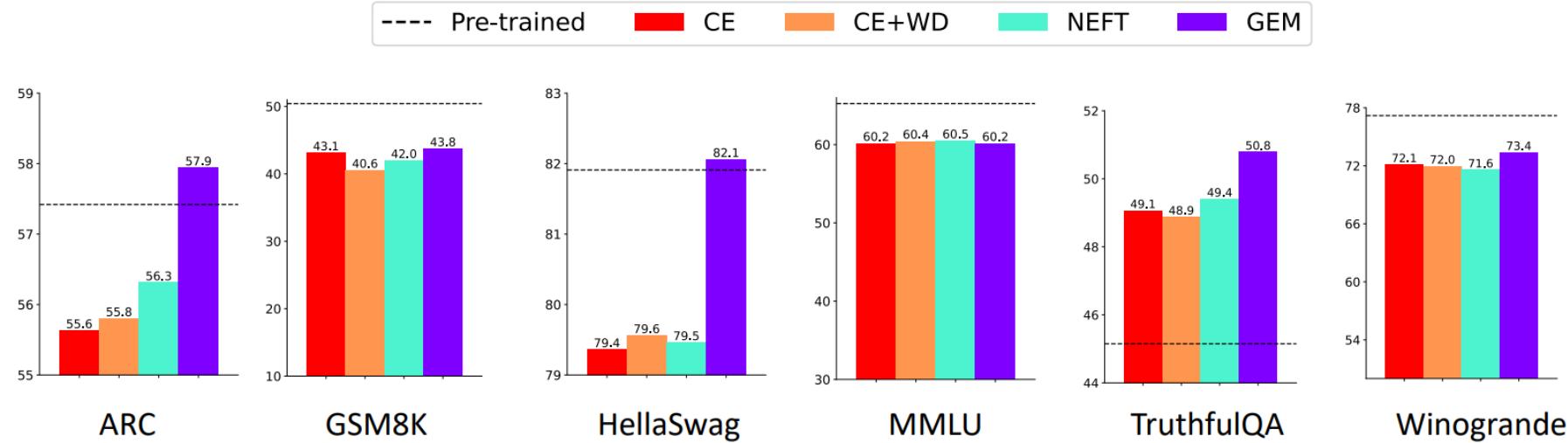
$$\theta_{k+1}[j] = \theta_k[j] - \eta * w_{i \leftarrow j}$$

- Increase the logit for the target token i in a similar manner:

$$\theta_{k+1}[i] = \theta_k[i] + \eta * w_{i \leftarrow j}$$



Performance on tasks from the OpenLLM leaderboard. The results indicate that GEM outperforms CE, demonstrating a lower alignment tax.



Performance on tasks from the OpenLLM leaderboard. The results indicate that GEM outperforms CE, demonstrating a lower alignment tax.

- While SFT is widely adopted for its simplicity and efficiency in adapting models to expert demonstrations, it suffers from poor generalization to out-of-distribution examples and tends to overfit training data.

$$\mathcal{L}_{\text{SFT}}(\theta) = \mathbb{E}_{(x, y^*) \sim \mathcal{D}} [-\log \pi_\theta(y^* | x)].$$

Its gradient is:

$$\nabla_\theta \mathcal{L}_{\text{SFT}}(\theta) = \mathbb{E}_{(x, y^*) \sim \mathcal{D}} [-\nabla_\theta \log \pi_\theta(y^* | x)].$$

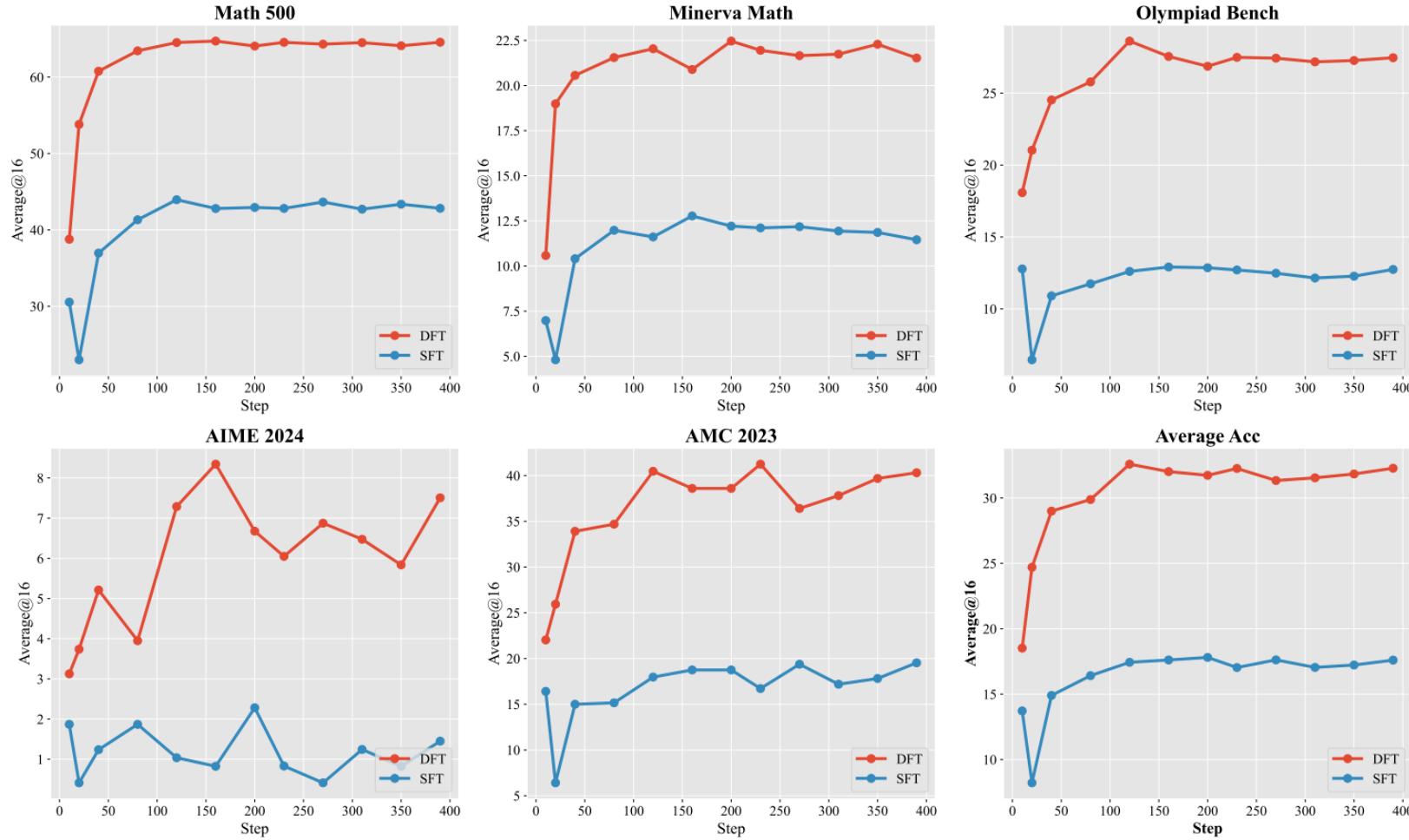
On the Generalization of SFT: A Reinforcement Learning Perspective with Reward Rectification



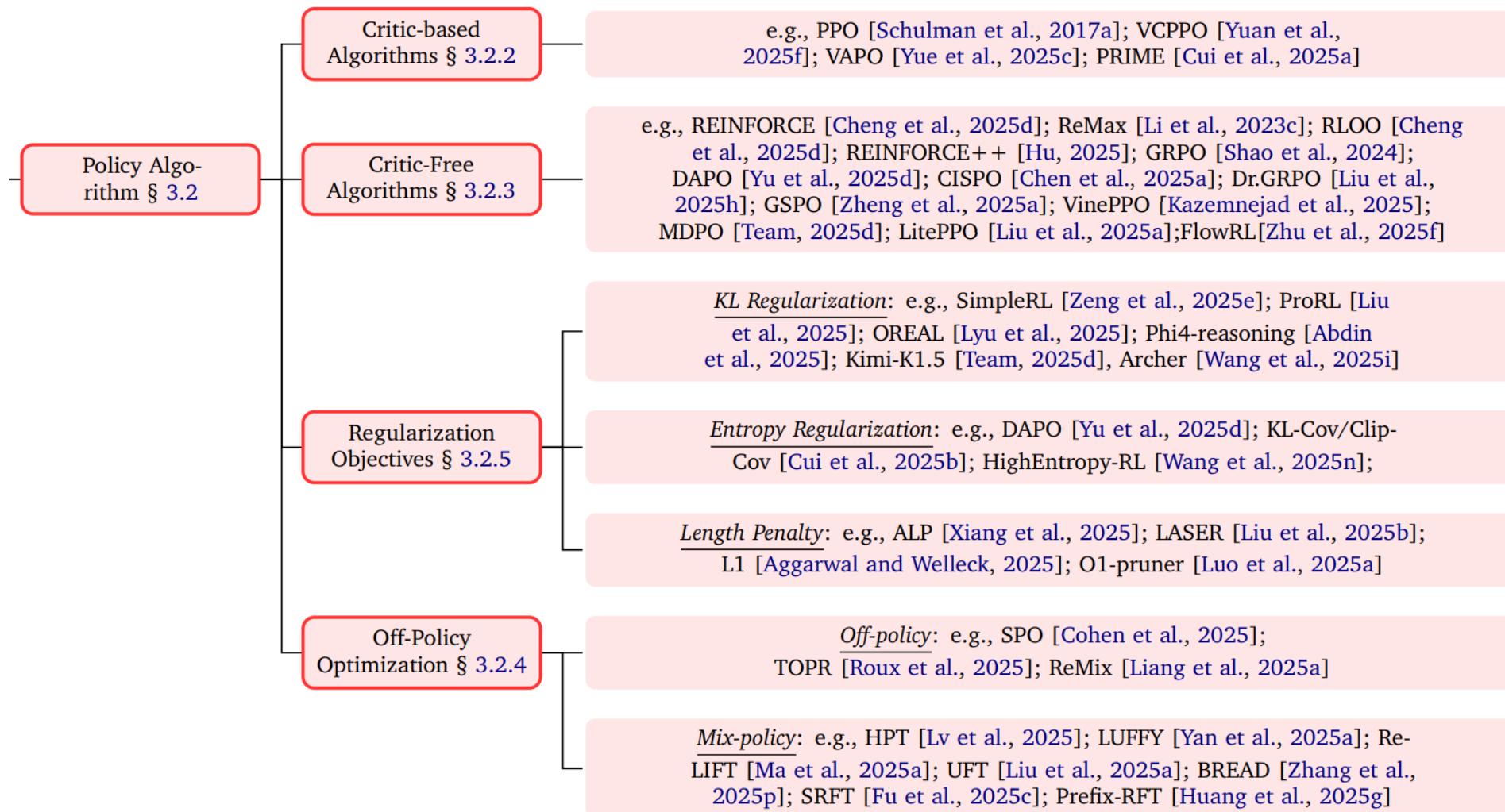
- The DFT proposes dynamically reweighting rewards by multiplying by a corrective inverse ratio (i.e., the policy probability itself). Specifically, the DFT modifies the gradient update of the SFT as follows:

$$\mathcal{L}_{\text{DFT}}(\theta) = \mathbb{E}_{(x, y^*) \sim \mathcal{D}} \left[- \sum_{t=1}^{|y^*|} \text{sg}(\pi_\theta(y_t^* | y_{<t}^*, x)) \log \pi_\theta(y_t^* | y_{<t}^*, x) \right].$$

- This formula can be understood as follows: for each token, its loss contribution is weighted by its current predicted probability.



Accuracy progression for Qwen2.5-Math-1.5B across mathematical benchmarks, illustrating faster convergence and better performance achieved by DFT relative to SFT.



Critic-based Algorithms

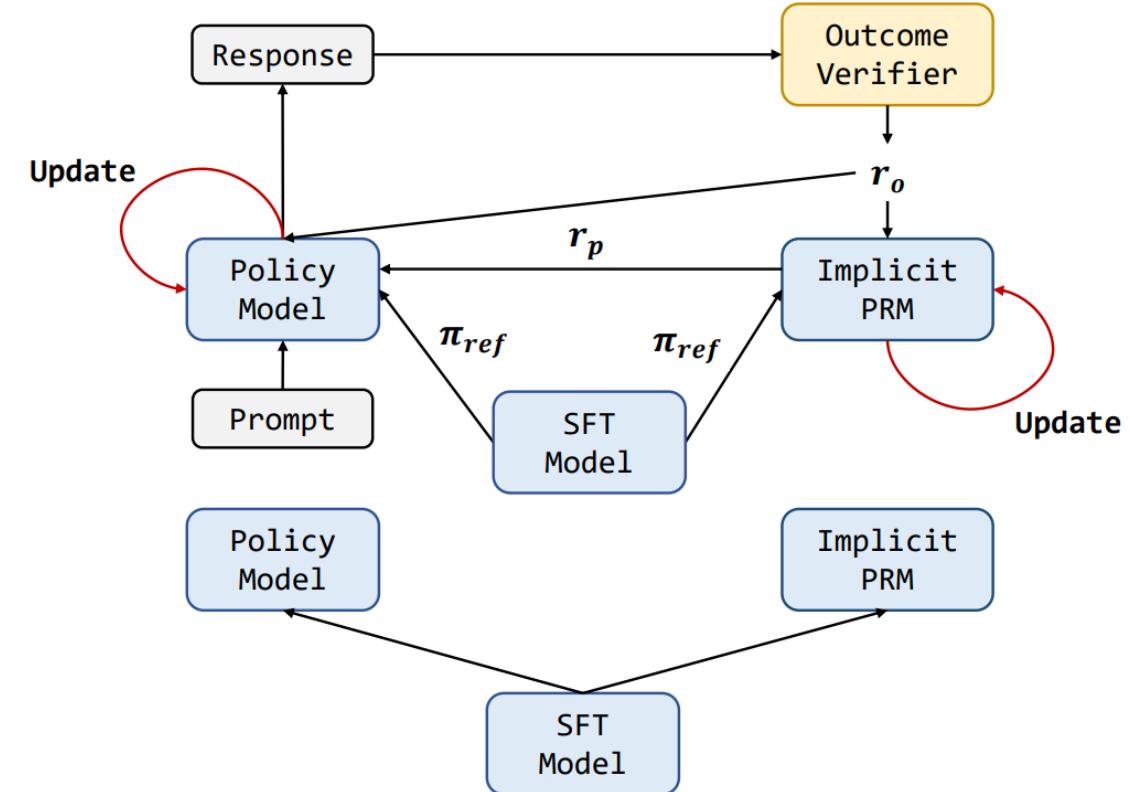


- Process Reinforcement through Implicit Rewards

- Motivation

- Current RL methods rely on inefficient sparse outcome rewards, as implementing effective dense process feedback is prohibitively expensive and difficult to scale online.

- PRIME enables scalable online learning by deriving dense, token-level rewards implicitly from outcome labels, eliminating the need for costly step-by-step annotations.



Critic-based Algorithms

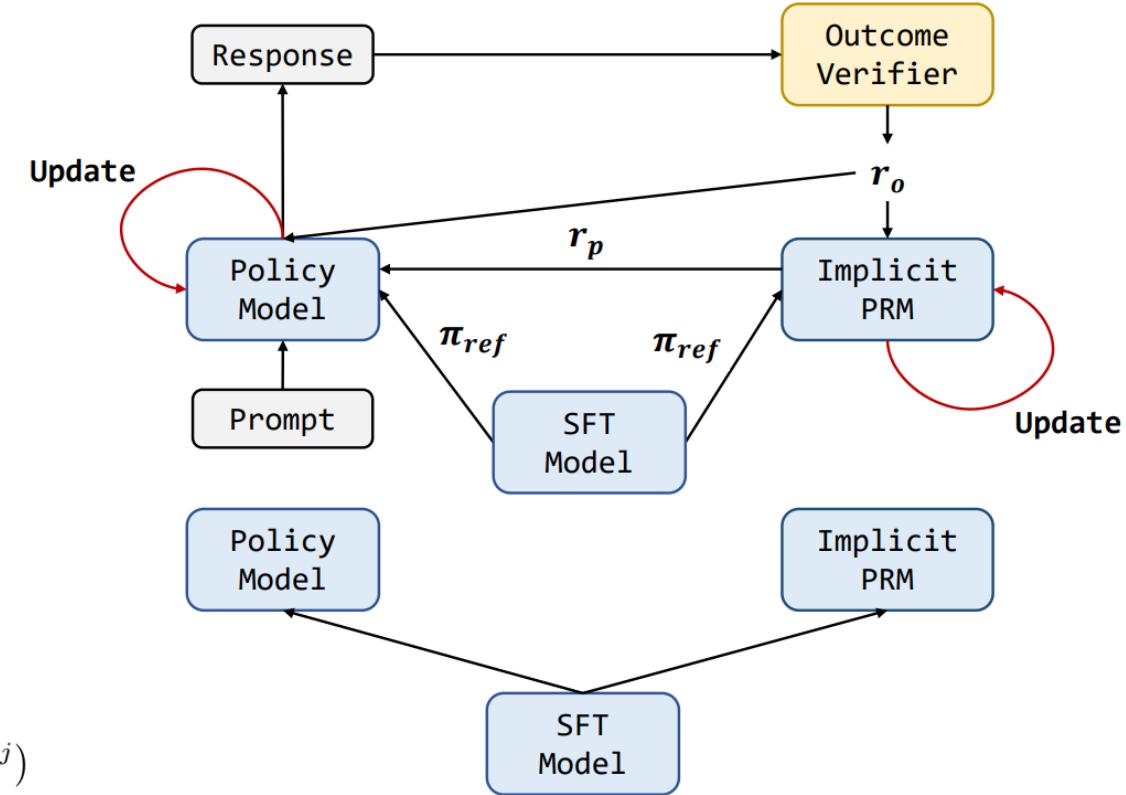


- PRIME operates through a four-step process that elegantly sidesteps the traditional challenges of dense reward learning. The key innovation lies in the implicit process reward formulation:

$$r_\phi(y_t) := \beta \log \frac{\pi_\phi(y_t | \mathbf{y}_{<t})}{\pi_{ref}(y_t | \mathbf{y}_{<t})}$$

- The advantage estimation combines returns from both implicit process rewards and sparse outcome rewards:

$$A_t^i = \underbrace{\sum_{s=t}^{|\mathbf{y}^i|} \gamma^{s-t} \cdot \left[r_\phi(y_s^i) - \frac{1}{K-1} \sum_{j \neq i} r_\phi(\mathbf{y}^j) \right]}_{\text{RLOO with implicit process rewards}} + \underbrace{r_o(\mathbf{y}^i) - \frac{1}{K-1} \sum_{j \neq i} r_o(\mathbf{y}^j)}_{\text{RLOO with outcome rewards}}$$



Critic-based Algorithms



Algorithm 1 Process Reinforcement through Implicit Rewards (PRIME)

Input Language model $\pi_{\theta_{\text{init}}}$; outcome verifier r_o ; dataset \mathcal{D} ; sample number K ; total iteration N .

- 1: Initialize policy model $\pi_\theta, \pi_{\theta_{\text{old}}} \leftarrow \pi_{\theta_{\text{init}}}$, implicit PRM and reference model $\pi_\phi, \pi_{\text{ref}} \leftarrow \pi_{\theta_{\text{init}}}$
- 2: **for** iteration = 1, ..., N **do**
- 3: Sample batch of prompts $\mathcal{B} \sim \mathcal{D}$
- 4: Generate K responses: $\{\mathbf{y}^1, \dots, \mathbf{y}^K\} \sim \pi_\theta(\cdot | \mathbf{x})$ for $\mathbf{x} \in \mathcal{B}$
- 5: Compute outcome rewards: $r_o(\mathbf{y}^{1:K})$
- 6: Apply accuracy filter (§3.3) on all prompts: $\mathcal{T} \leftarrow \text{Filter}(\mathbf{x}, \mathbf{y}^{1:K}, r_o(\mathbf{y}^{1:K}))$ for $\mathbf{x} \in \mathcal{B}$
- 7: Forward pass $\pi_\phi, \pi_{\text{ref}}$ on each $(\mathbf{x}, \mathbf{y}) \in \mathcal{T}$ to obtain implicit process reward $r_\phi(y_t)$ with Eq. 3
- 8: Update Implicit PRM π_ϕ by CE loss on $(\mathbf{x}, \mathbf{y}, r_o(\mathbf{y})) \in \mathcal{T}$:

$$\mathcal{L}_{\text{CE}}(\phi) = -\mathbb{E}_{(\mathbf{x}, \mathbf{y}, r_o(\mathbf{y})) \sim \mathcal{T}} [r_o(\mathbf{y}) \cdot \log \sigma(r_\phi(\mathbf{y})) + (1 - r_o(\mathbf{y})) \cdot \log(1 - \sigma(r_\phi(\mathbf{y})))]$$

- 9: Compute advantages A with Eq. 5
- 10: Update policy π_θ by PPO loss in Eq. 6
- 11: Update old parameters: $\theta_{\text{old}} \leftarrow \theta$
- 12: **end for**

Output Optimized policy model π_θ

Critic-based Algorithms

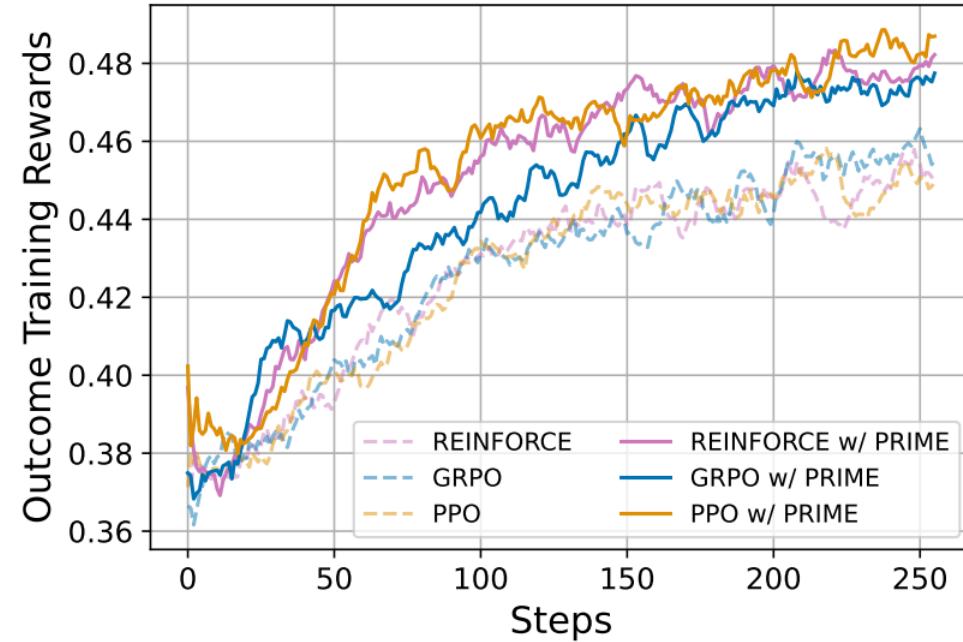


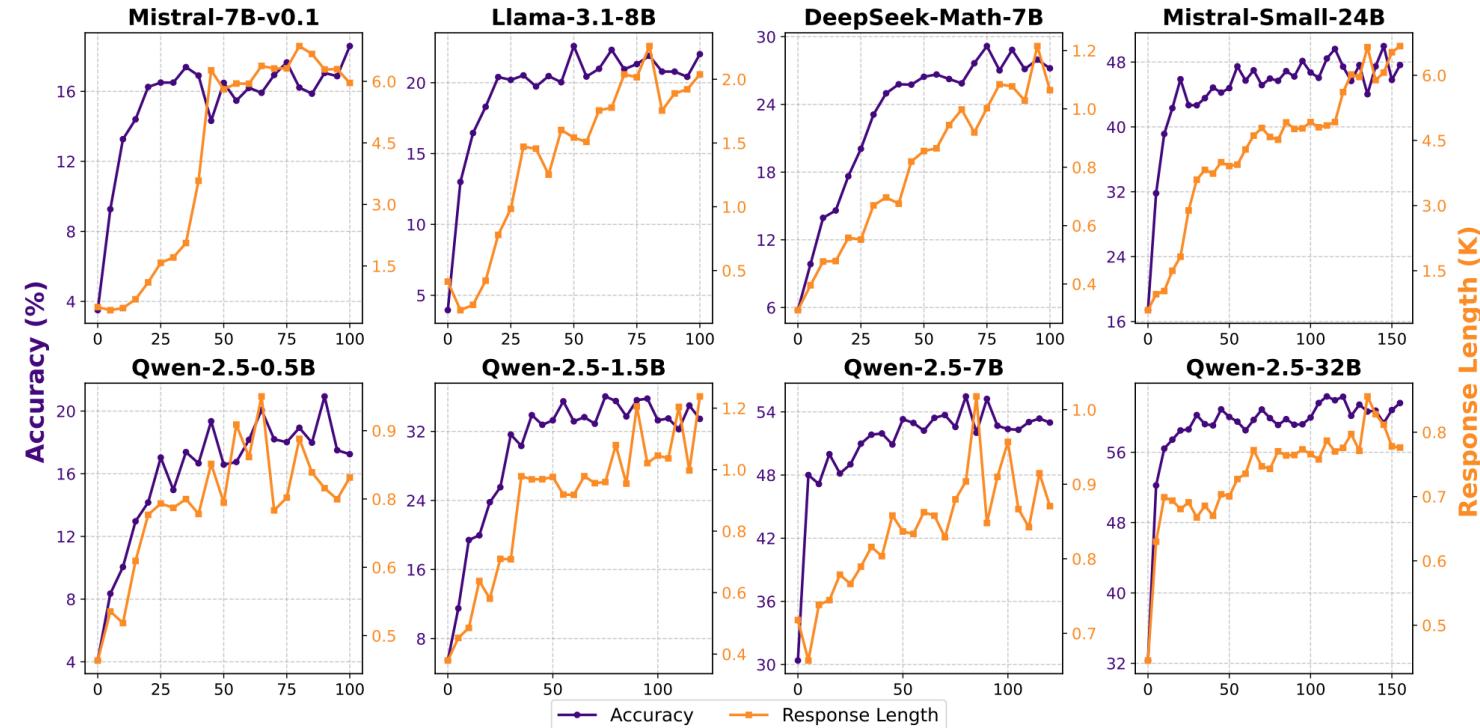
Figure 7: PRIME also generally benefits REINFORCE, GRPO, and PPO.

- PRIME contributes consistently regardless of the policy update method, making it a generic algorithm.
- It indicates that PRIME is a general plug-in for almost any RL algorithm for LLM., which largely extends the use cases of PRIME.

Regularization Objectives



- Zero reinforcement learning (zero RL) training represents a paradigm shift in large language model development, where complex reasoning capabilities emerge through reinforcement learning applied directly to base models without prior supervised fine-tuning.



Regularization Objectives



- Zero RL Training also Demonstrates Strong Generalization Performance.

(ID)

Model	GSM8K	MATH 500	Minerva Math	Olympiad Bench	AIME24 (Pass@1)	AIME24 (Avg@32)	AMC23	Avg.
<i>Llama, DeepSeek and Mistral Models</i>								
Mistral-v0.1-7B	21.2	4.2	4.0	2.4	0.0	0.0	0.0	5.3
↪ + SimpleRL-Zoo	75.0	15.8	6.6	4.1	0.0	0.2	10.0	18.6
Llama-3.1-8B	39.7	13.6	4.8	3.1	0.0	0.2	2.5	10.6
↪ + SimpleRL-Zoo	79.2	23.0	9.6	5.3	0.0	0.2	15.0	22.0
DeepSeek-Math-7B	28.4	19.4	5.5	4.7	0.0	0.0	10.0	11.3
↪ + SimpleRL-Zoo	78.5	39.6	21.0	12.6	3.3	0.6	20.0	29.2
Mistral-Small-24B	78.6	43.6	10.7	11.6	3.3	0.5	17.5	27.6
↪ + SimpleRL-Zoo	92.0	70.6	36.8	36.6	16.7	13.1	45.0	49.6
<i>Qwen Series Models</i>								
Qwen-2.5-0.5B	36.7	15.8	4.8	2.8	0.0	0.3	12.5	12.1
↪ + SimpleRL-Zoo	49.5	34.4	10.3	8.9	0.0	0.7	22.5	20.9
Qwen-2.5-1.5B	55.7	29.6	6.6	6.5	0.0	0.1	12.5	18.5
↪ + SimpleRL-Zoo	74.4	59.0	20.2	21.0	6.7	4.2	35.0	36.1
Qwen-2.5-7B	88.2	64.6	25.7	30.1	3.3	0.3	30.0	40.3
↪ + SimpleRL-Zoo	91.7	78.2	38.6	40.4	20.0	15.6	62.5	55.2
Qwen-2.5-Math-7B	65.5	63.6	12.5	25.8	13.3	8.6	42.5	37.2
↪ + SimpleRL-Zoo	90.2	80.2	37.5	39.0	40.0	24.0	70.0	59.5
Qwen-2.5-14B	91.6	65.4	24.3	33.5	6.7	3.4	37.5	43.2
↪ + SimpleRL-Zoo	94.4	80.2	40.4	44.9	23.3	14.2	57.6	56.8
Qwen-2.5-32B	92.9	68.6	27.9	31.1	10.0	4.5	45.0	45.9
↪ + SimpleRL-Zoo	95.9	82.4	42.6	46.4	36.7	27.2	67.5	61.9

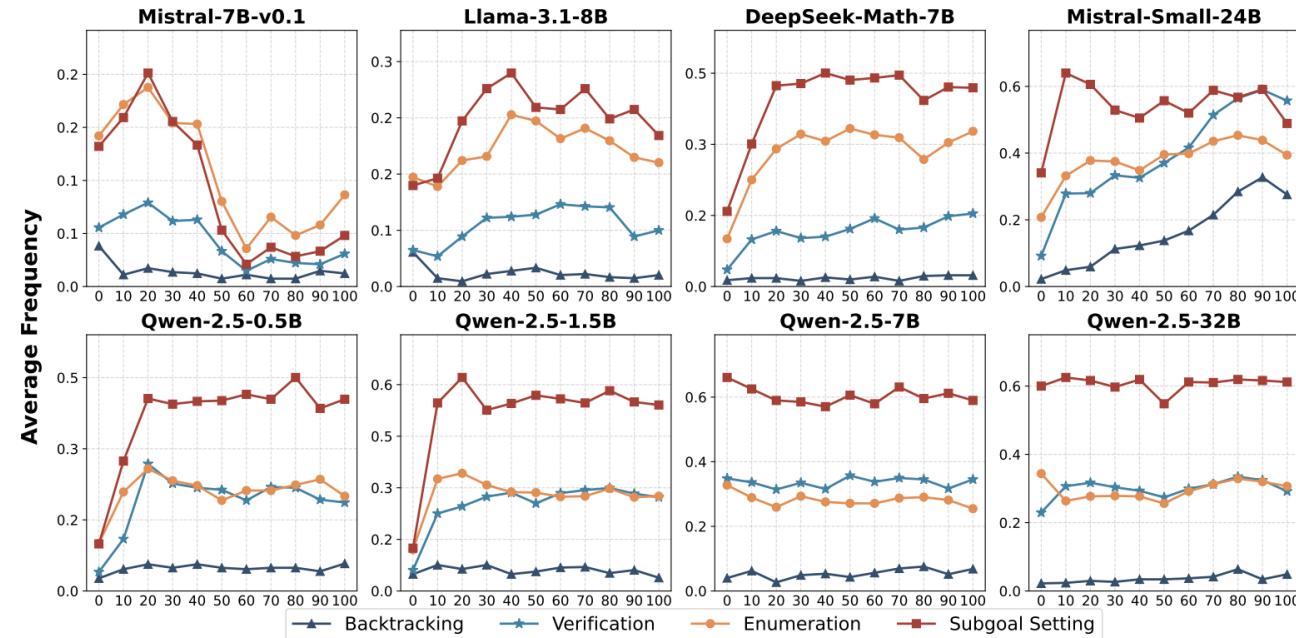
(OOD)

Model	IFEVAL strict-prompt	MMLU Stem	MMLU	GPQA	Avg.
<i>Llama, DeepSeek and Mistral Models</i>					
Mistral-v0.1-7B	13.5	26.1	28.0	23.2	22.7
↪ + SimpleRL-Zoo	21.8	28.1	34.6	30.3	28.7
Llama-3.1-8B	16.1	27.1	28.7	22.7	23.6
↪ + SimpleRL-Zoo	25.1	40.7	44.5	20.2	32.6
DeepSeek-Math-7B	11.5	21.6	22.7	19.2	18.7
↪ + SimpleRL-Zoo	16.3	47.4	45.5	27.3	34.1
Mistral-Small-24B	17.4	30.9	31.7	20.2	25.0
↪ + SimpleRL-Zoo	23.5	73.9	78.8	45.0	55.3
<i>Qwen Series Models</i>					
Qwen-2.5-0.5B	9.6	23.2	24.9	24.8	20.6
↪ + SimpleRL-Zoo	14.4	32.1	34.6	26.3	26.8
Qwen-2.5-1.5B	15.2	33.1	35.4	24.8	27.1
↪ + SimpleRL-Zoo	20.3	42.1	45.2	28.8	34.1
Qwen-2.5-7B	21.3	39.8	38.6	23.7	30.8
↪ + SimpleRL-Zoo	25.9	49.6	47.0	29.8	38.1
Qwen-2.5-Math-7B	14.1	40.6	38.0	27.8	30.1
↪ + SimpleRL-Zoo	17.0	55.6	56.6	35.4	41.1
Qwen-2.5-14B	22.9	59.8	63.5	24.8	42.7
↪ + SimpleRL-Zoo	29.4	76.3	79.1	50.0	58.7
Qwen-2.5-32B	24.6	60.7	62.7	38.9	46.7
↪ + SimpleRL-Zoo	31.2	79.0	82.5	49.5	60.6

Regularization Objectives



- A particularly significant finding was the observation of the "aha moment" in non-Qwen small models for the first time. Models like Llama3-8B and DeepSeek-Math-7B, which initially lacked strong instruction-following and self-verification abilities, showed dramatic increases in reflection-oriented behaviors from near 0% to approximately 50%.



Regularization Objectives



- Format rewards that enforce strict output formatting (such as requiring `\boxed{}` for answers) significantly harm exploration, especially for models with weaker initial instruction-following capabilities.

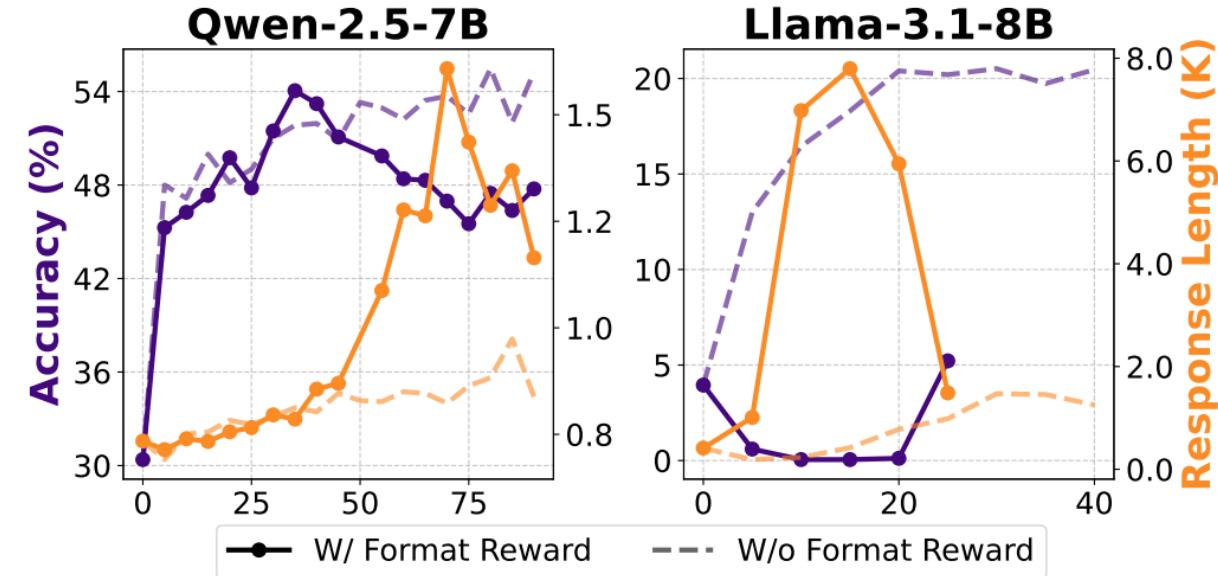


Figure 6: Accuracy and response length with and without format rewards.

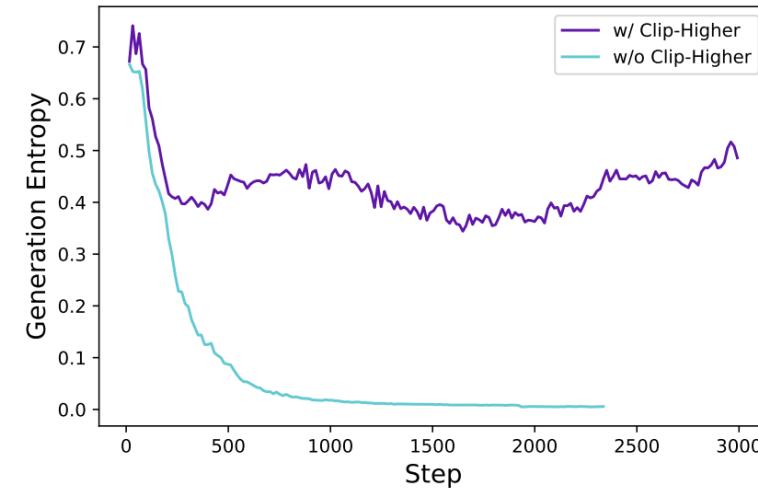
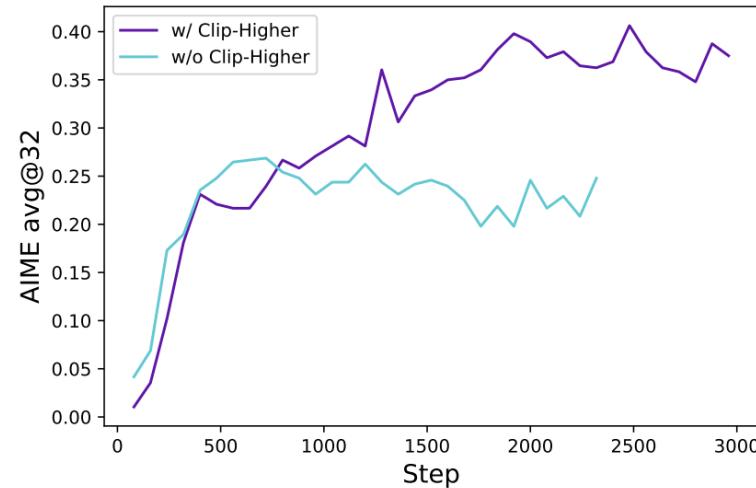


DAPO: An Open-Source LLM Reinforcement Learning System at Scale

- Challenges in Large-Scale RL for Reasoning Context
 - Entropy Collapse
 - Problem: Policy rapidly converges to high-probability tokens.
 - Impact: Becomes deterministic too early, preventing exploration of diverse reasoning paths.
 - Gradient Starvation
 - Problem: Math prompts often yield uniform batch outcomes (all correct or all incorrect).
 - Impact: Causes zero gradients in group-relative calculations, wasting compute and stalling convergence.
 - Length Bias & Reward Noise
 - Problem: Standard loss treats all response lengths equally.
 - Impact: Underweights complex long-chain reasoning and introduces noise via harsh truncation penalties.

- Clip-Higher: Decoupled Clipping for Enhanced Exploration
 - The Clip-Higher technique modifies the standard PPO clipping mechanism by decoupling the importance sampling ratio bounds. The modified policy gradient objective becomes:

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

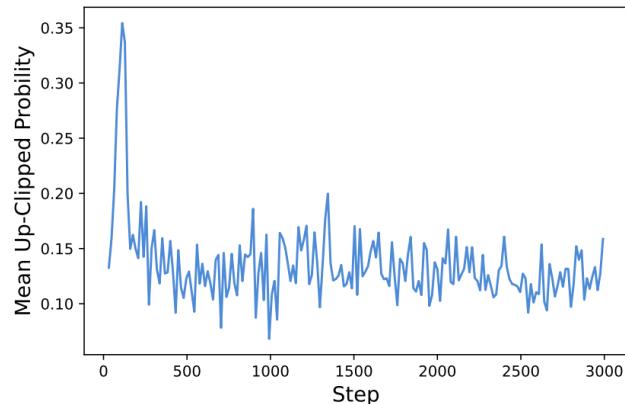


- Dynamic Sampling: Eliminating Zero-Gradient Samples

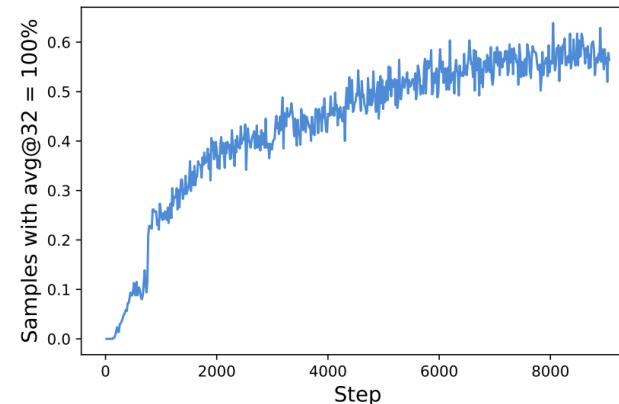
- Dynamic sampling addresses gradient starvation by ensuring every prompt in a training batch contributes meaningful gradients. The system over-samples prompts and filters out those where all generated responses yield identical rewards (all correct or all incorrect), which would result in zero group-relative acc.

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

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



(a) Mean up-clipped probability.

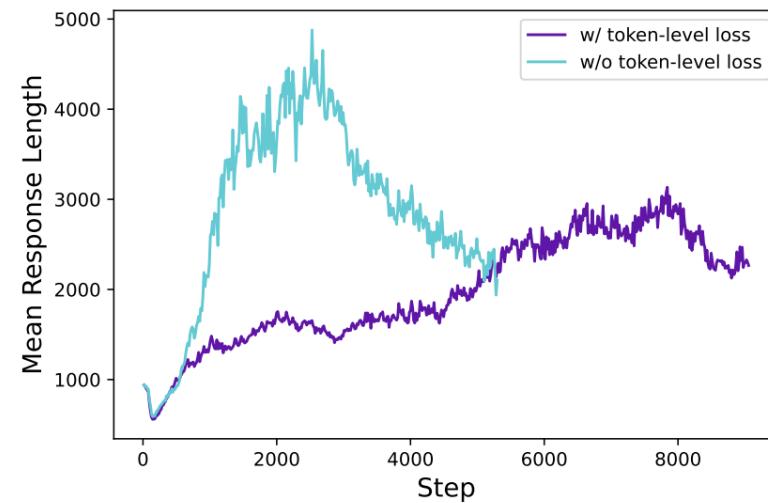
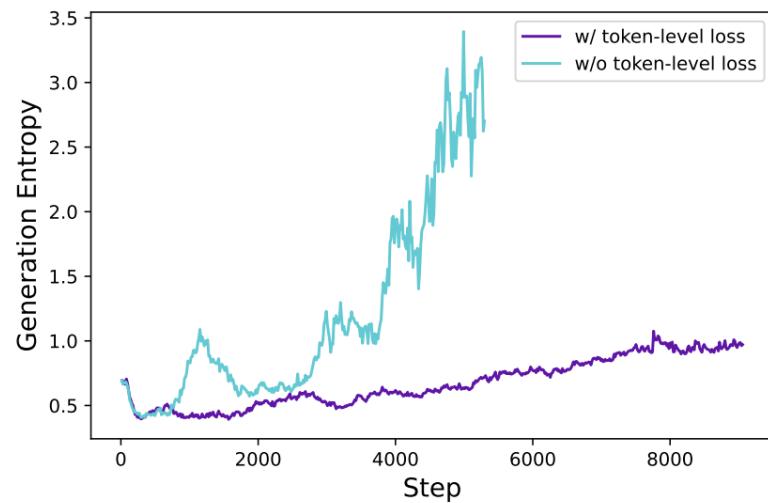


(b) The proportion of samples with an accuracy of 1.

- Token-Level Policy Gradient Loss

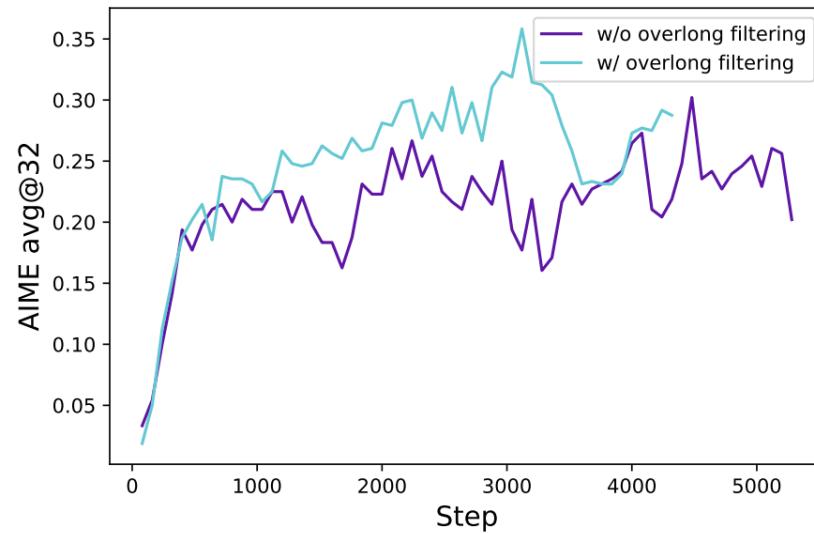
- Standard GRPO calculates loss at the sample level, averaging token losses within each response before aggregating across samples. This approach gives equal weight to all responses regardless of length, potentially diminishing the influence of longer, more detailed reasoning chains.

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

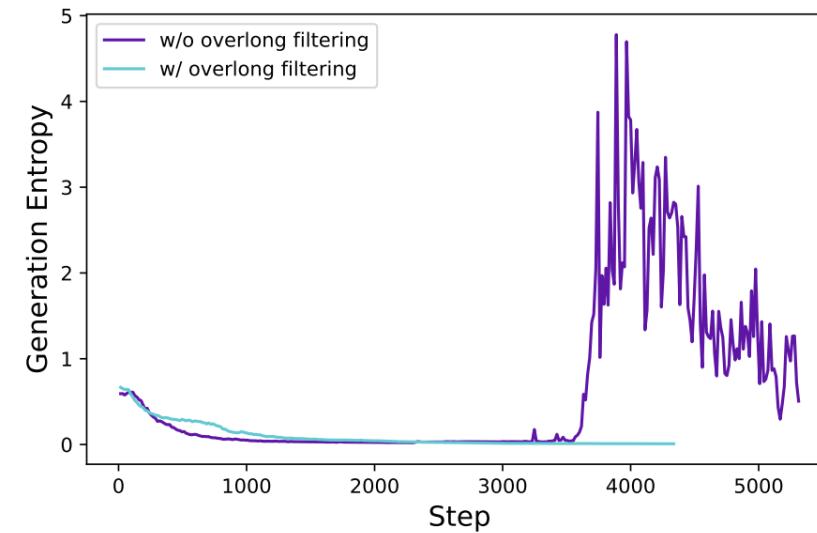


- Overlong Reward Shaping

- To address reward noise from length constraints, DAPO implements soft overlong punishment rather than harsh truncation penalties.
- Responses exceeding a threshold length receive a graduated penalty proportional to their excess length, rather than a fixed negative reward.



(a) Performance on AIME.

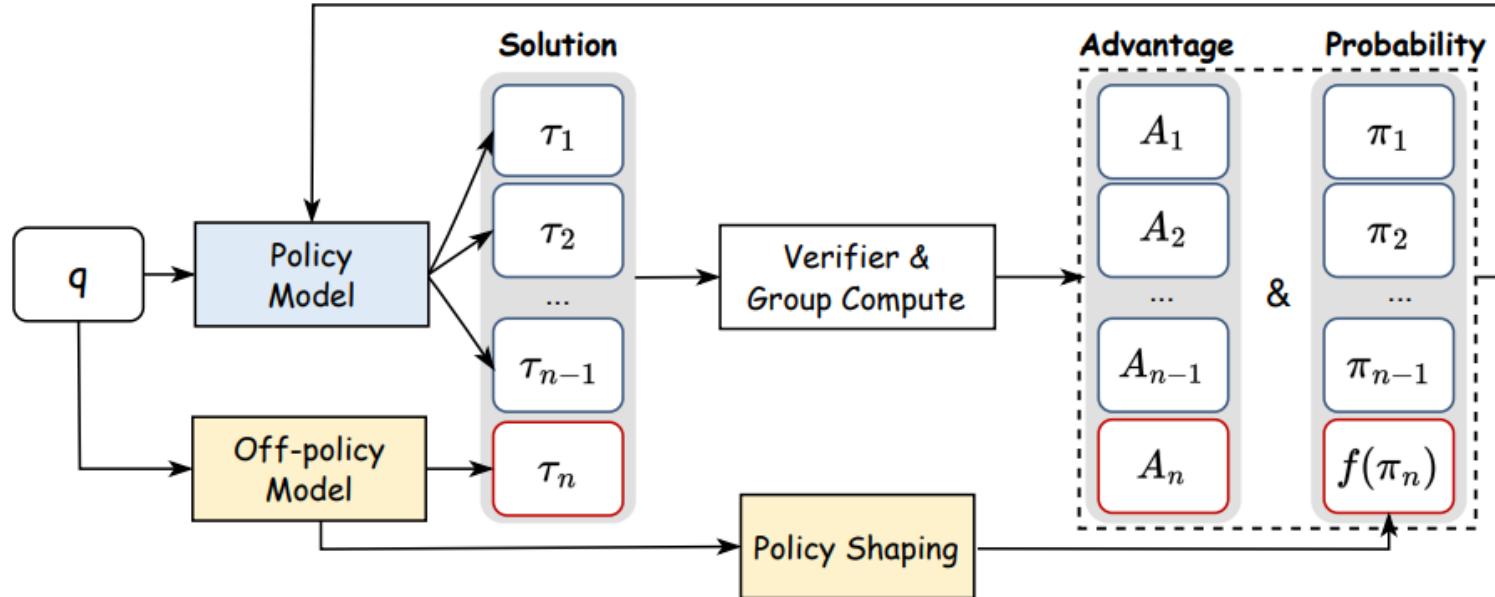


(b) Entropy of actor model.

Off-Policy Optimization



- Current RLVR approaches face a fundamental limitation
 - they operate in an "on-policy" manner, meaning models can only learn from their own generated outputs. This constraint traps models in an "echo chamber" where they can only amplify existing behaviors rather than acquire genuinely novel reasoning capabilities beyond their initial cognitive boundaries.



- On-policy RLVR vs. Supervised Fine-Tuning (SFT)
 - Ø On-policy Reinforcement Learning
 - ✓ Unified Training & Inference: Unbiased rewards and strong generalization.
 - ✗ Model Constraints: Requires inherent reasoning potential; struggles with sparse rewards otherwise.
 - ✗ Limited Exploration: Difficult to discover rare but high-value reasoning paths; prone to local optima.



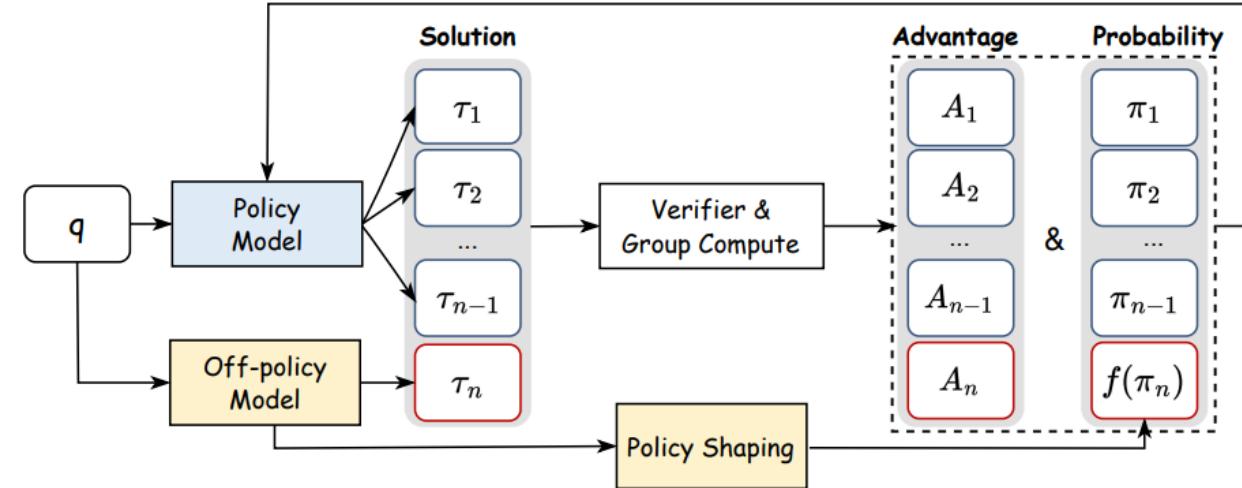
- On-policy RLVR vs. Supervised Fine-Tuning (SFT)

- Ø Supervised Fine-Tuning
 - ✓ Efficient Learning: Stable learning from demonstrations with fast convergence and easy implementation.
 - ✗ Lack of Exploration: Unable to derive improvements from new reasoning trajectories.
 - ✗ Poor Generalization: Rote imitation of external demonstrations leads to poor adaptability on unseen task structures.



$$\hat{A}_i = \frac{R(\tau_i) - \text{mean}(\mathcal{G}_{\text{on}} \cup \mathcal{G}_{\text{off}})}{\text{std}(\mathcal{G}_{\text{on}} \cup \mathcal{G}_{\text{off}})},$$

$$\mathcal{J}_{\text{Mixed}}(\theta) = \frac{1}{Z} \left(\underbrace{\sum_{j=1}^{N_{\text{off}}} \sum_{t=1}^{|\tau_j|} \text{CLIP}(\hat{r}_{j,t}(\theta, \phi), \hat{A}_j, \epsilon)}_{\text{off-policy objective}} + \underbrace{\sum_{i=1}^{N_{\text{on}}} \sum_{t=1}^{|\tau_i|} \text{CLIP}(r_{i,t}(\theta), \hat{A}_i, \epsilon)}_{\text{on-policy objective}} \right),$$



Part I: Off-policy Data Update – Leveraging external demonstrations to guide learning.
 Part II: On-policy Data Update – Enabling the model to autonomously explore high-quality strategies.



Advantage computations:

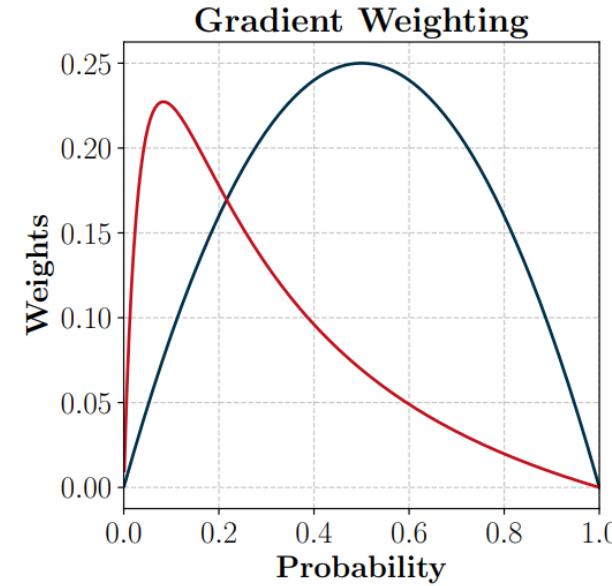
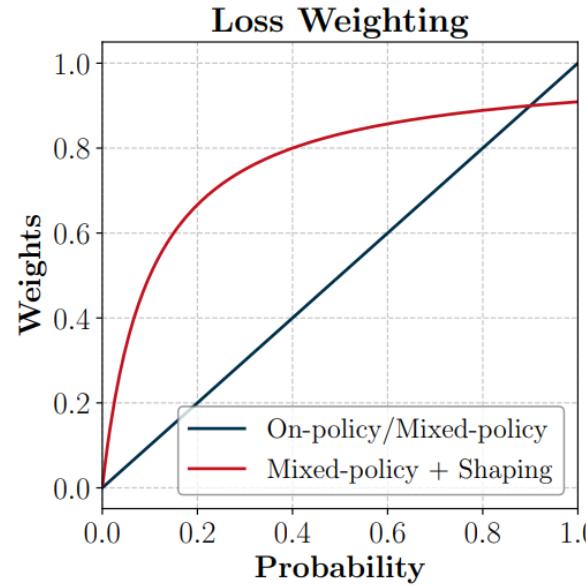
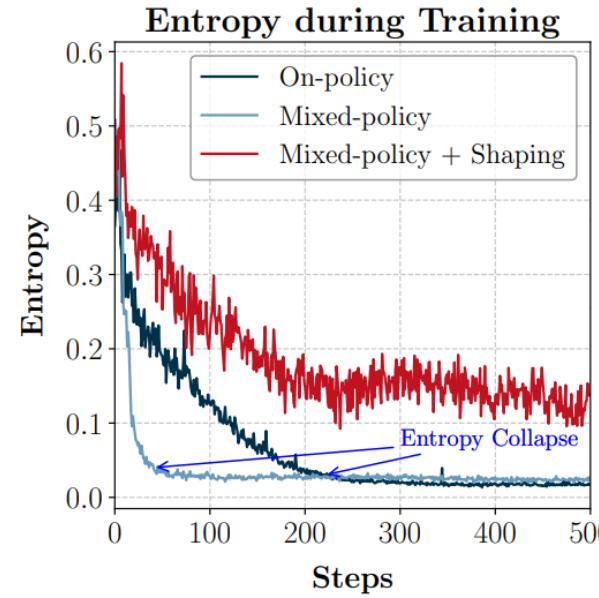
$$\hat{A}_i = \frac{R(\tau_i) - \text{mean}(\mathcal{G}_{\text{on}} \cup \mathcal{G}_{\text{off}})}{\text{std}(\mathcal{G}_{\text{on}} \cup \mathcal{G}_{\text{off}})},$$

Mixed-Policy GRPO: $\mathcal{J}_{\text{Mixed}}(\theta) = \underbrace{\frac{1}{Z} \left(\sum_{j=1}^{N_{\text{off}}} \sum_{t=1}^{|\tau_j|} \text{CLIP}(\hat{r}_{j,t}(\theta, \phi), \hat{A}_j, \epsilon) \right)}_{\text{off-policy objective}} + \underbrace{\sum_{i=1}^{N_{\text{on}}} \sum_{t=1}^{|\tau_i|} \text{CLIP}(r_{i,t}(\theta), \hat{A}_i, \epsilon)}_{\text{on-policy objective}},$

where $\hat{r}_{j,t}(\theta, \phi) = \pi_\theta(\tau_{j,t}|q, \tau_{j,< t})/\pi_\phi(\tau_{j,t}|q, \tau_{j,< t})$ and $r_{i,t}(\theta) = \pi_\theta(\tau_{i,t}|q, \tau_{i,< t})/\pi_{\theta_{\text{old}}}(\tau_{i,t}|q, \tau_{i,< t})$.
 $Z = \sum_{j=1}^{N_{\text{off}}} |\tau_j| + \sum_{i=1}^{N_{\text{on}}} |\tau_i|$ is the normalization factor.

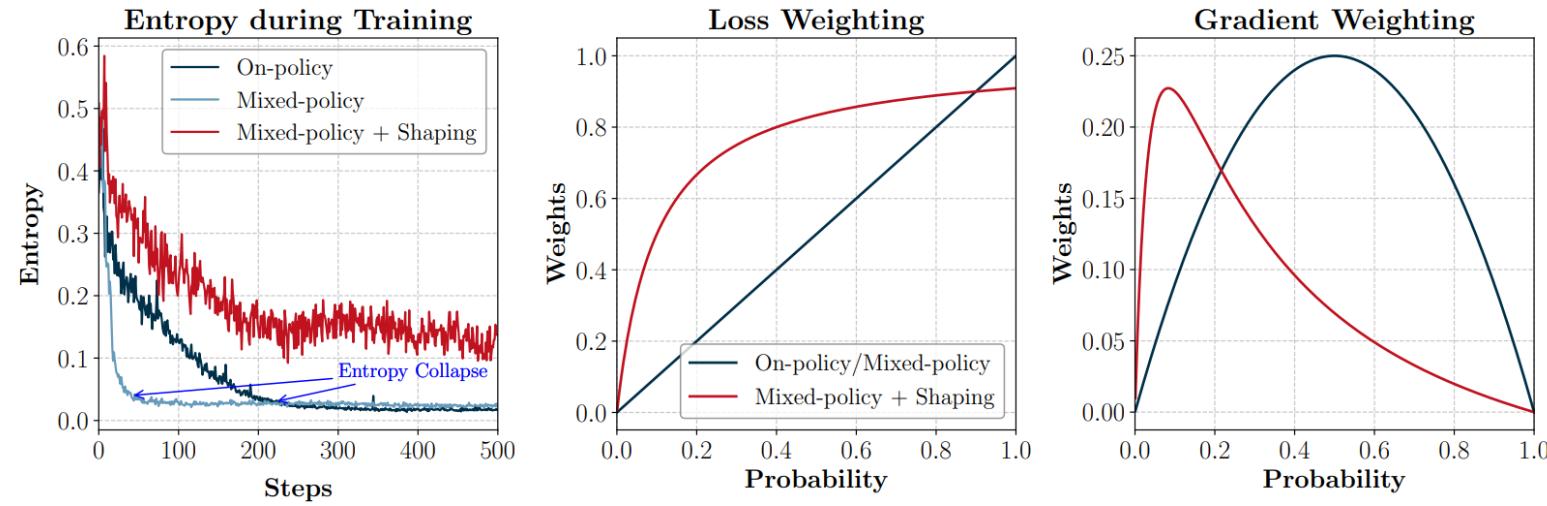
Part I: Off-policy Data Update – Leveraging external demonstrations to guide learning.

Part II: On-policy Data Update – Enabling the model to autonomously explore high-quality strategies.



Entropy Collapse in RL: Models are prone to entropy collapse; as training progresses, the policy's output distribution becomes increasingly concentrated. This reduces exploration capability, leading to performance stagnation.

Risks in Hybrid Training: Premature reliance on high-probability, deterministic outputs from off-policy data can trap the model in local optima, preventing it from learning diverse, high-quality reasoning paths.



$$\mathcal{J}_{\text{SHAPING}}(\theta) = \frac{1}{Z} \left(\sum_{j=1}^{N_{\text{off}}} \sum_{t=1}^{|\tau_j|} f(\hat{r}_{j,t}(\theta, \phi)) \cdot \hat{A}_j + \sum_{i=1}^{N_{\text{on}}} \sum_{t=1}^{|\tau_i|} \text{CLIP}(r_{i,t}(\theta), \hat{A}_i, \epsilon) \right),$$

- Off-policy Objective: Apply a non-linear function $f(x)$ to the importance sampling weights to prioritize gradients from low-probability, exploratory actions.
- On-policy Objective: Maintain standard PPO/GRPO updates.
 - Benefit: Amplifies gradient weights for low-probability actions, enabling the model to learn rare but critical reasoning behaviors from off-policy data while preserving high entropy.

Model	In-Distribution Performance						Out-of-Distribution Performance			
	AIME 24/25	AMC	MATH-500	Minerva	Olympiad	Avg.	ARC-c	GPQA*	MMLU-Pro	Avg.
Qwen-Base [30]	11.5/4.9	31.3	43.6	7.4	15.6	19.0	18.2	11.1	16.9	15.4
Qwen-Instruct [30]	12.5/10.2	48.5	80.4	32.7	41.0	37.6	70.3	24.7	34.1	43.0
Previous RLVR methods										
SimpleRL-Zero [5]	27.0/6.8	54.9	76.0	25.0	34.7	37.4	30.2	23.2	34.5	29.3
OpenReasoner-Zero [7]	16.5/15.0	52.1	82.4	33.1	47.1	41.0	66.2	29.8	58.7	51.6
PRIME-Zero [24]	17.0/12.8	54.0	81.4	39.0	40.3	40.7	73.3	18.2	32.7	41.4
Oat-Zero [6]	33.4 /11.9	61.2	78.0	34.6	43.4	43.7	70.1	23.7	41.7	45.2
Our On-policy RLVR Replication										
On-Policy RL	25.1/15.3	62.0	84.4	39.3	46.8	45.5	82.3	<u>40.4</u>	49.3	57.3
Alternative Off-policy Learning Methods										
SFT	22.2/22.3	52.8	82.6	<u>40.8</u>	43.7	44.1	75.2	24.7	42.7	47.5
RL w/ SFT Loss	19.5/16.4	49.7	80.4	34.9	39.4	40.1	71.2	23.7	43.2	46.0
SFT+RL	25.8 / 23.1	62.7	<u>87.2</u>	39.7	50.4	48.2	72.4	24.2	37.7	44.8
Our Methods										
LUFFY	29.4 / 23.1	<u>65.6</u>	87.6	37.5	57.2	<u>50.1*</u>	80.5	39.9	53.0	<u>57.8*</u>
LUFFY†	<u>30.7</u> / <u>22.5</u>	66.2	86.8	41.2	<u>55.3</u>	50.4*	<u>81.8</u>	49.0	<u>54.7</u>	61.8*

- Qwen2.5-Math-7B:

- Significant Improvement: Achieved an average score of 50.1 across 6 math benchmarks, surpassing existing RLVR methods by +6.4 points.
- Strong Generalization: Gained +8.1 points on the newly released AIME 25, validating the transfer and generalization of off-policy reasoning patterns.

Can the model generate multiple answers during testing to improve its reasoning ability?

Self-Refine: Iterative Refinement with Self-Feedback

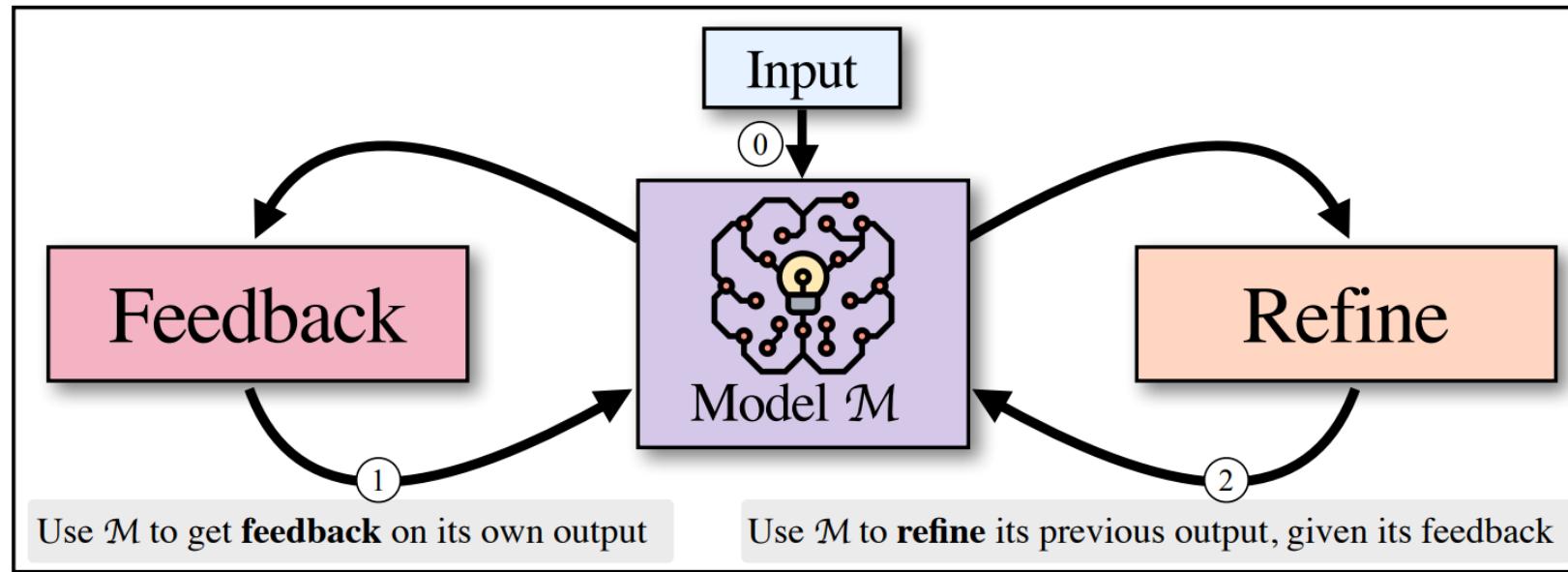


Figure 1: Given an input (①), SELF-REFINE starts by generating an output and passing it back to the same model \mathcal{M} to get feedback (②). The feedback is passed back to \mathcal{M} , which refines the previously generated output (③). Steps (②) and (③) iterate until a stopping condition is met. SELF-REFINE is instantiated with a language model such as GPT-3.5 and does not involve human assistance.

Self-Refine



南方科技大学
SOUTHERN UNIVERSITY OF SCIENCE AND TECHNOLOGY

- SELF-REFINE introduces an iterative approach that enables large language models to improve their own outputs through a simple feedback loop.
- The method operates entirely within a single LLM using three sequential steps:
 - initial generation,
 - self-feedback,
 - and refinement.
- Unlike traditional approaches that require separate training phases or additional models, SELF-REFINE leverages the few-shot prompting capabilities of modern LLMs to create a self-contained improvement system.

Self-Refine



The core of SELF-REFINE lies in its iterative self-refinement mechanism, which generates specific and actionable feedback that is then used to optimize its own output.

This approach enables the model to learn from past iterations, thus avoiding repeated mistakes and continuously improving with each refinement.

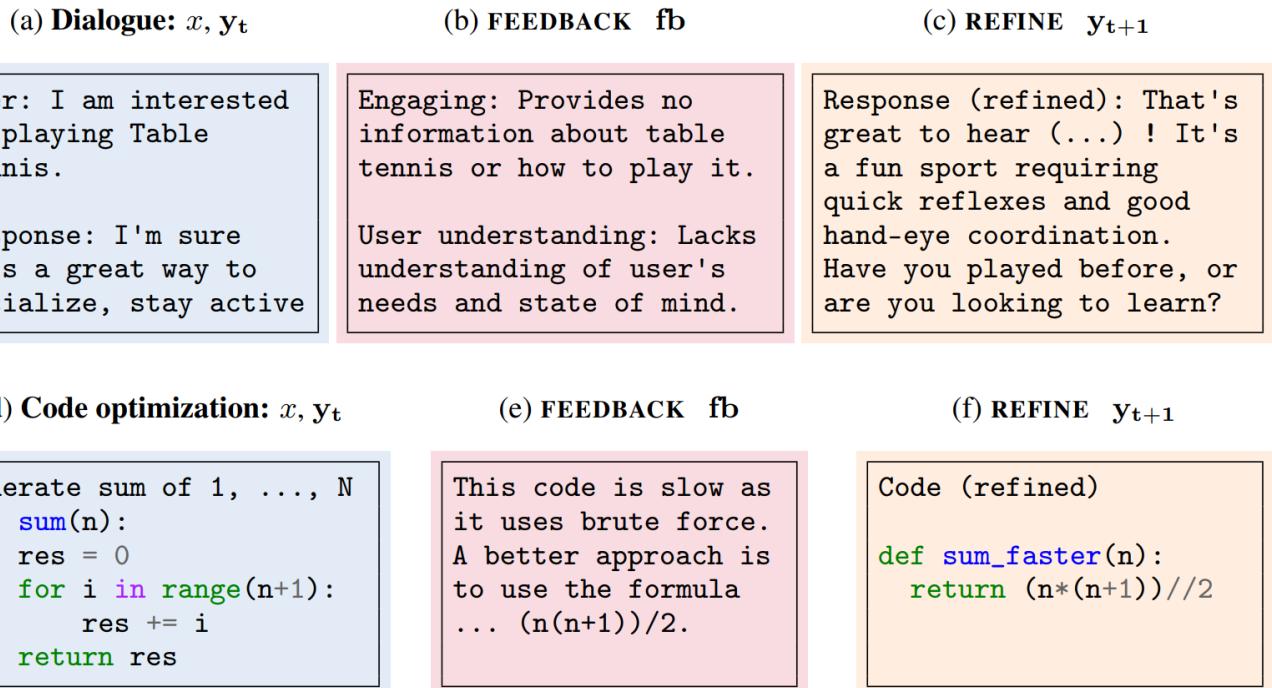
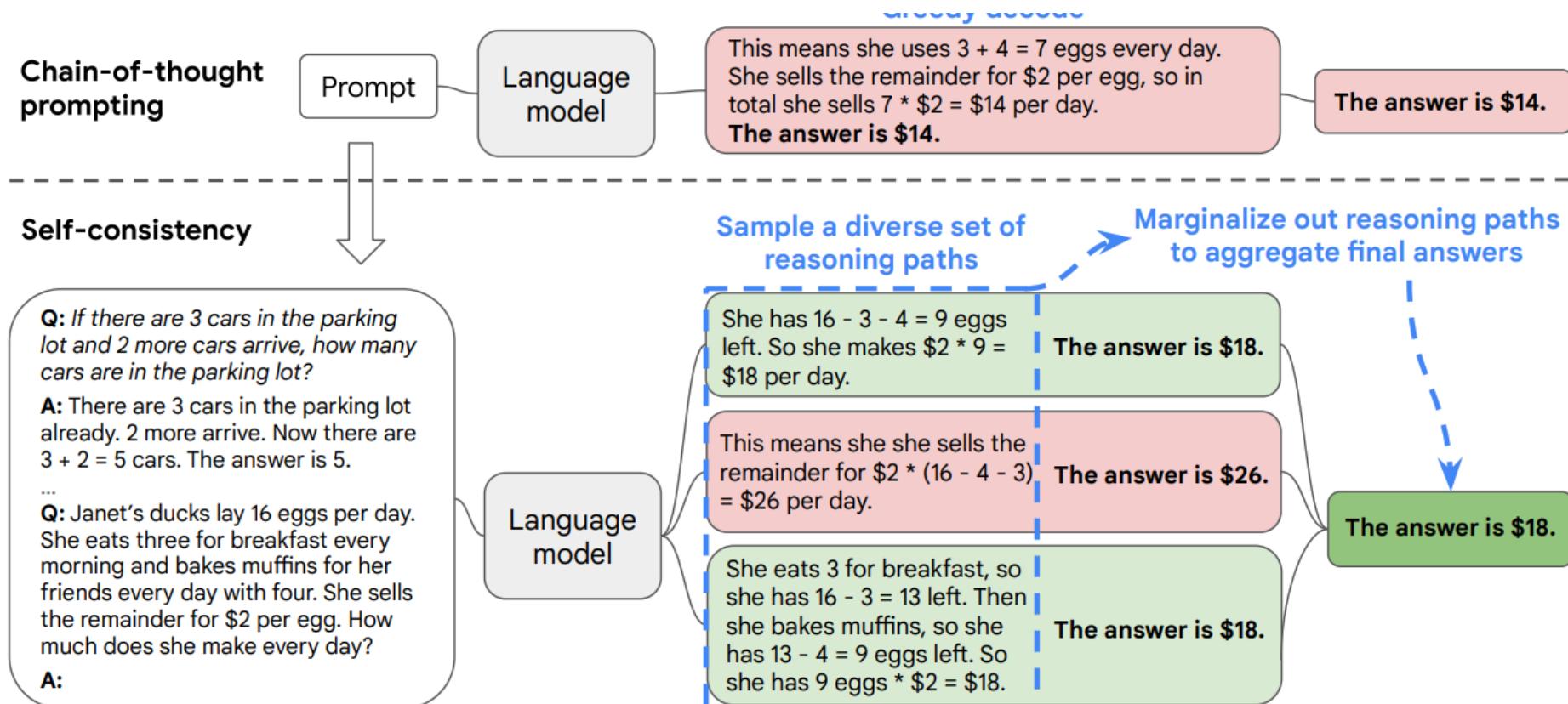


Figure 2: Examples of SELF-REFINE: an initial output █ generated by the base LLM and then passed back to the *same* LLM to receive feedback █ to the *same* LLM to refine the output █. The top row illustrates this for dialog generation where an initial dialogue response can be transformed into a more engaging one that also understands the user by applying feedback. The bottom row illustrates this for code optimization where the code is made more efficient by applying feedback.

Self-Consistency



Self-Consistency Improves Chain of Thought Reasoning in Language Models



Self-Consistency

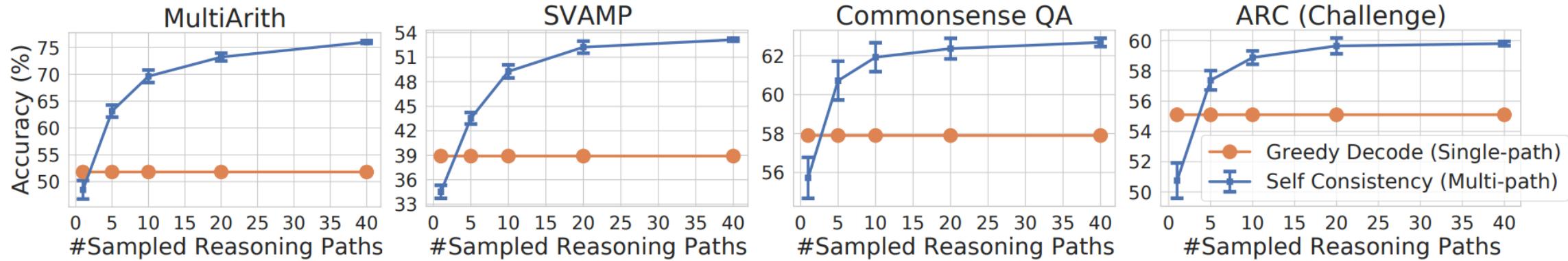


- **Self-Consistency Algorithm.** The self-consistency method operates through three main steps:
 - **Chain-of-Thought Prompting:** The model receives a prompt with few-shot examples demonstrating step-by-step reasoning, following the standard CoT format.
 - **Diverse Path Sampling:** Instead of greedy decoding, the method samples multiple reasoning paths using techniques like:
 - Temperature sampling ($T=0.7$)
 - Top-k sampling ($k=40$)
 - Nucleus sampling (top-p)
 - **Answer Aggregation:** The final answers from all sampled paths are aggregated through majority voting to select the most consistent answer.

Self-Consistency



- Self-consistency (blue) significantly improves accuracy over CoT-prompting with greedy decoding (orange) across arithmetic and commonsense reasoning tasks, over LaMDA-137B. Sampling a higher number of diverse reasoning paths consistently improves reasoning accuracy.



Further Reading



- [\[2512.01374\] Stabilizing Reinforcement Learning with LLMs: Formulation and Practices](#)



2025-12-04

Stabilizing Reinforcement Learning with LLMs: Formulation and Practices

Chujie Zheng*
Junrong Lin

Kai Dang Bowen Yu*
Yuqiong Liu Hao Lin
An Yang Jingren Zhou

Mingze Li Huiqiang Jiang
Chencan Wu Feng Hu
Junyang Lin

Qwen Team, Alibaba Inc.



南方科技大学
SOUTHERN UNIVERSITY OF SCIENCE AND TECHNOLOGY

Thank you