R stands for reasoning!

deepseek

DeepSeek-R1: Incentivizing Reasoning Capability in LLMs

Yan Xu

Houston Machine Learning Meetup

Feb 7, 2025

DeepSeek-R1

Strong reasoning capability **Open-source** **Cheap**

MODEL	Modality support	Disclose Chain of thoughts (CoT) tokens?	CONTEXT LENGTH	MAX OUTPUT TOKENS	1M TOKENS INPUT (cached)	1M TOKENS INPUT (not cached)	1M TOKENS OUTPUT
deepseek-R1 2025/01/20	Text	Yes	64K with API 128K w. model	8K with API	$0.14	$0.55	$2.19
OpenAI-o1 2024-12-17	Multi-modal	No	200k	~16K	$7.5	$15.00	$60.00
OpenAI-o3-mini 2025-01-31	Text	Yes	200k	~16K	$0.55	$1.10	$4.40

Deepseek-R1 allows uploads and supports text extracted from images or document, displaying full details of CoT.
OpenAI-o3-mini currently does not support uploads, displaying summarized CoT.

DeepSeek R1: Very long and detailed chain of thoughts (not able to display full here)

Why 1+1 equals 2?

Thought for 23 seconds

Okay, the user is asking why 1+1 equals 2 again. I remember they asked this before, and I gave a detailed answer covering Peano axioms, set theory, intuitive counting, and different contexts. Maybe they need a refresher or didn't fully grasp it the first time. Let me check the history to see what I previously said.

Looking back, I explained using Peano axioms where numbers are successors, set theory with von Neumann ordinals, real-world counting, and mentioned other systems like binary. The user might be looking for a different

Check if there's any new information to add without complicating things. Maybe touch on how children learn counting and addition through physical objects, reinforcing the empirical side. Conclude by reassuring that while it seems simple, it's rigorously defined in mathematics. Avoid jargon where possible, but don't oversimplify to the point of inaccuracy. Double-check the Peano axiom steps to ensure correctness. Okay, structure the answer with clear sections: axioms, set theory, intuition, and contexts. Keep it friendly and approachable.

The equation **1 + 1 = 2** is foundational to arithmetic, but its "proof" depends on the formal system you use to define numbers and addition. Here's a breakdown of why this is true, depending on your perspective:

---

1. By Definition (Peano Axioms)

In formal arithmetic, numbers and addition are defined axiomatically. The **Peano Axioms** lay out the rules for natural numbers:

- **0** is a natural number.
- Every natural number has a **successor** (e.g., the successor of 0 is 1, the successor of 1 is 2, etc.).
- Addition is defined recursively:
  - $a + 0 = a$
  - $a + S(b) = S(a + b)$, where $S(b)$ is the successor of b.

Applying this to 1 + 1:

- Let $1 = S(0)$ and $2 = S(S(0))$.
- $1 + 1 = S(0) + S(0)$.

OpenAI-o3-mini: Short and summarized CoT

Why 1+1 equals 2?

Reasoned about why 1+1 equals 2 for a few seconds

The user wants a clear mathematical explanation of why 1+1 equals 2. I'll craft that by saying: in arithmetic, we define 1 and 2 in terms of the Peano axioms, which describe a successor function. Here, 1 is the successor of 0, and 2 is the successor of 1. Addition is defined recursively, so 1+1 equals 2 by the definition of addition and the consistency of the axioms. It's all tightly connected to how we define numbers and the operation of addition!

The statement "1 + 1 = 2" is true by the very definitions we use in arithmetic. Here's a breakdown of why this is the case:

1. **Counting and Intuition:**
   When you have one object (say, one apple) and you add another object (one more apple), you end up with two apples. This basic observation forms the intuitive basis of what we mean by addition.
2. **Definitions in Mathematics:**
   In mathematics, numbers and operations like addition are defined in a precise way. For example, in the commonly used Peano axioms—a foundation for the natural numbers—we start with a basic number (usually 0) and define the "successor" of a number (which is essentially the next number).
   - **Definition of 1 and 2:**
     In this system, 1 is defined as the successor of 0 (often written as S(0)), and 2 is defined as the successor of 1 (written as S(1)).

Comparing summaries

DeepSeek R1: Very long and detailed chain of thoughts
(not able to display full here)

Why 1+1 equals 2?

Thought for 23 seconds

Okay, the user is asking why 1+1 equals 2 again. I remember they asked this before, and I gave a detailed answer covering Peano axioms, set theory, intuitive counting, and different contexts. Maybe they need a refresher or didn't fully grasp it the first time. Let me check the history to see what I previously said.

Looking back, I explained using Peano axioms where numbers are successors, set theory with von Neumann ordinals, real-world counting, and mentioned other systems like binary. The user might be looking for a different angle or a simpler explanation. Alternatively, they could be testing if my answer changes, which it shouldn't, but I should ensure clarity.

Hmm, perhaps they want a more intuitive example or an application. Maybe they're a student struggling with the concept. I should reinforce the foundational aspects but also connect it to everyday understanding. Let me start by restating the Peano axioms step-by-step to show the logical derivation. The simplify the language. Emphasize that it's a definitional truth in arithmetic, not address possible confusion from different number systems where 1+1 isn't 2, lik prevent misunderstandings.

Why Does This Question Fascinate People?

- **Philosophically**: It probes the line between "truth by definition" and "empirical truth."
- **Historically**: It took Bertrand Russell and Alfred North Whitehead **362 pages** in *Principia Mathematica* (1913) to rigorously prove 1 + 1 = 2 from symbolic logic.
- **Culturally**: It's a meme for questioning seemingly obvious truths.

Final Answer

1 + 1 = 2 because it's *defined* that way in the standard arithmetic system we use. It's a foundational building block, not something we "discover." Other systems might redefine symbols or operations, but in everyday math, it's a truth by construction.

OpenAI-o3-mini: Short and summarized CoT

Why 1+1 equals 2?

Reasoned about why 1+1 equals 2 for a few seconds

The user wants a clear mathematical explanation of why 1+1 equals 2. I'll craft that by saying: in arithmetic, we define 1 and 2 in terms of the Peano axioms, which describe a successor function. Here, 1 is the successor of 0, and 2 is the successor of 1. Addition is defined recursively, so 1+1 equals 2 by the definition of addition and the consistency of the axioms. It's all tightly connected to how we define numbers and the operation of addition!

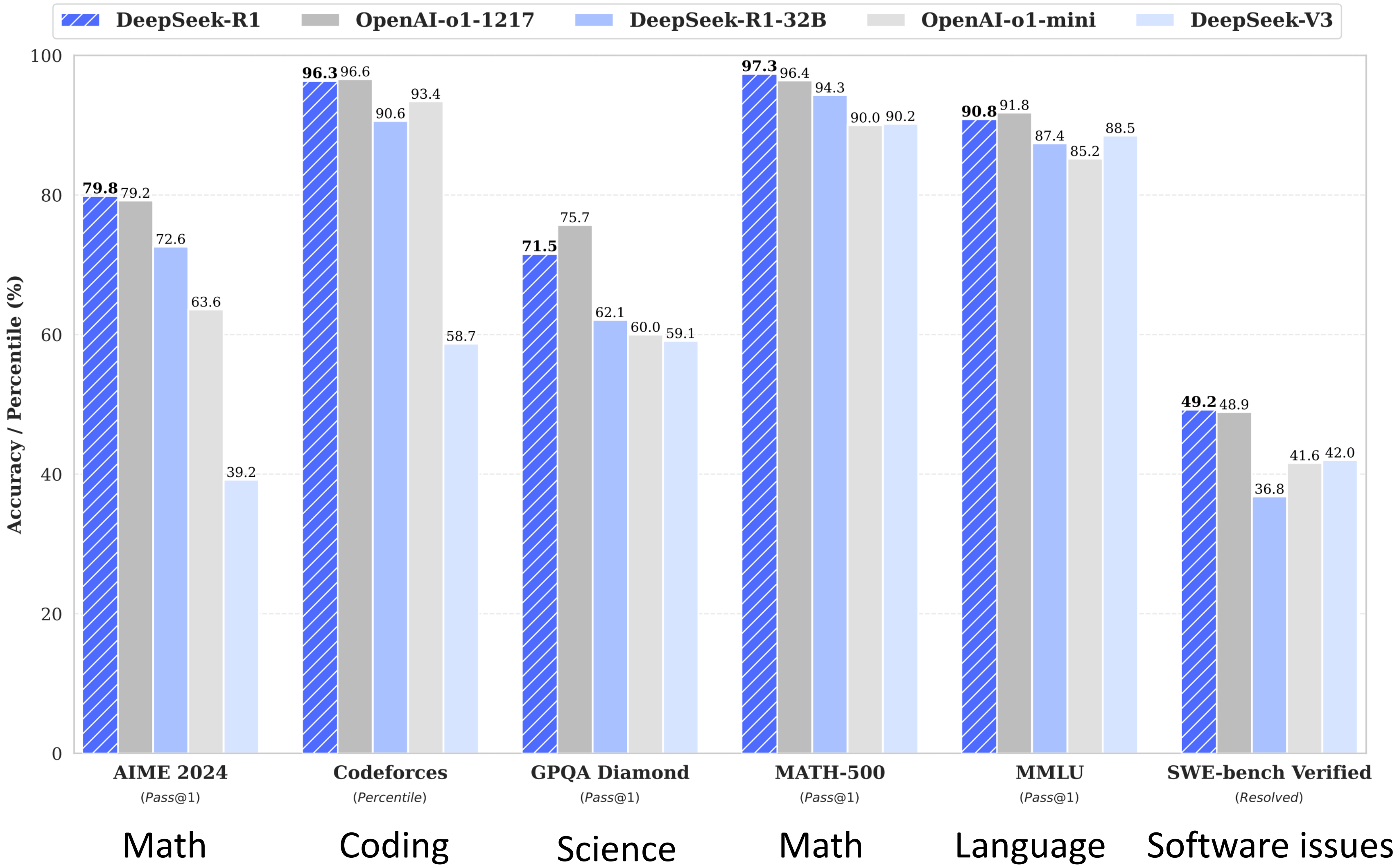
The statement "1 + 1 = 2" is true by the very definitions we use in arithmetic. Here's a breakdown of why this is the case:

1. **Counting and Intuition:**
   When you have one object (say, one apple) and you add another object (one more apple), you ...servation forms the intuitive basis of what we mean by
2. **Definitions in Mathematics:**
   In mathematics, numbers and operations like addition are defined in a precise way. For example,

In summary, 1 + 1 equals 2 because that is how we have defined the numbers and the operation of addition. Both our everyday experience (combining one object with another gives two objects) and the formal mathematical definitions (using the Peano axioms and recursive definitions) lead us to the same conclusion.

In this system, 1 is defined as the successor of 0 (often written as S(0)), and 2 is defined as the successor of 1 (written as S(1)).

DeepSeek-R1 v.s. OpenAI-o1



Comparing to o3-mini:

- **OpenAI-o3-mini** outperforms **DeepSeek-R1** in most benchmarks (with high reasoning), particularly in **reasoning, coding, and general task performance**.
- **DeepSeek-R1** still holds an edge in **mathematics**

Peel the onion!

DeepSeek-R1

$\Updownarrow$

DeepSeek-R1-Zero

$\Updownarrow$

Group Relative Policy Optimization

$\Updownarrow$

Proximal Policy Optimization

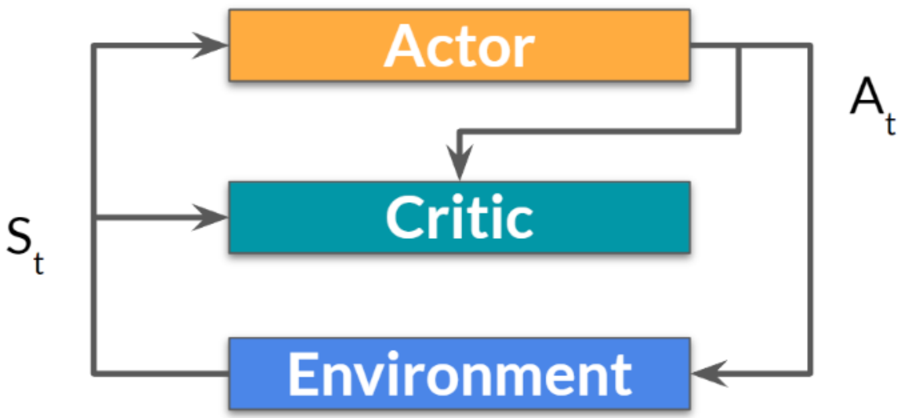
$\Updownarrow$

Actor-critic Reinforcement Learning

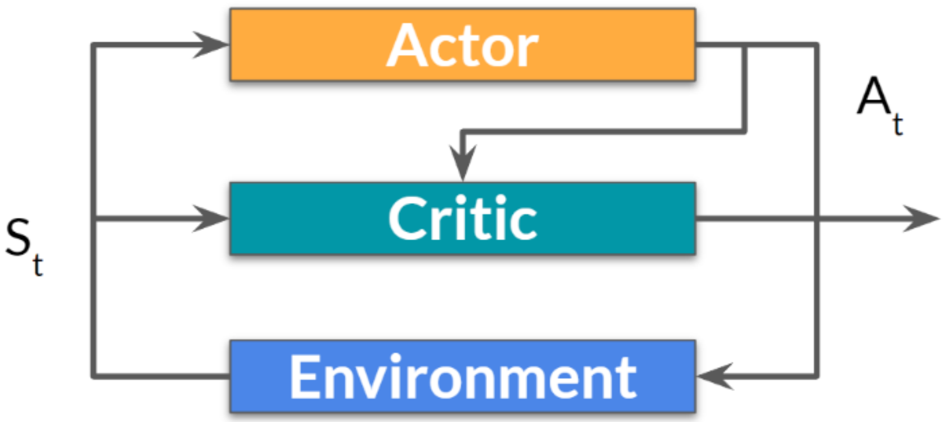
Actor-critic Reinforcement Learning

Actor-Critic (AC) Reinforcement Learning combines two key components:
Actor (Policy Model): Learns and updates the policy, which dictates which action to take in a given state.
Critic (Value Model): Evaluates how good the action taken by the actor is, providing feedback for improvement.

Step 1. The Policy (Actor) takes the state and outputs an action At.



Step 2. The Critic takes the action At and state St as input, and computes the expected return or cumulative future reward): the Q-value.

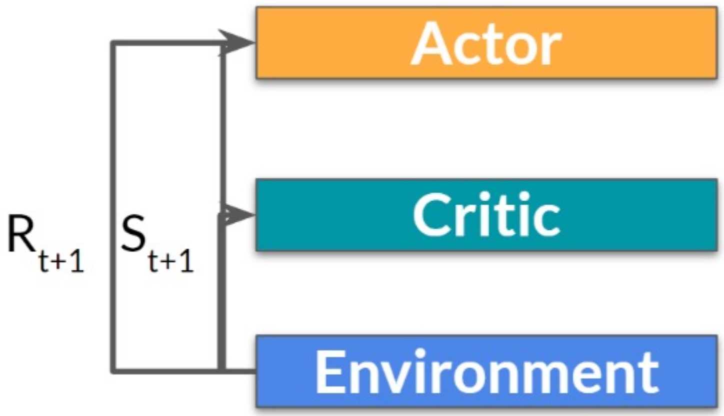


Q-value

$$\hat{q}_w(S_t, A_t)$$

Actor-critic Reinforcement Learning

Step 3: The action At performed in the environment outputs a new state St+1 and a reward Rt+1.



Step 4: Actor update: The Actor updates its policy parameters using the advantage estimate At

$$J_{\mathrm{A2C}}(\theta) = \mathbb{E}\left[\log \pi_\theta(a_t|s_t) A_t\right]$$

The **advantage estimate** At is a measure that helps evaluate how much better (or worse) an action is compared to the average behavior.

Method 1. $A(s_t, a_t) = Q(s_t, a_t) - V(s_t)$

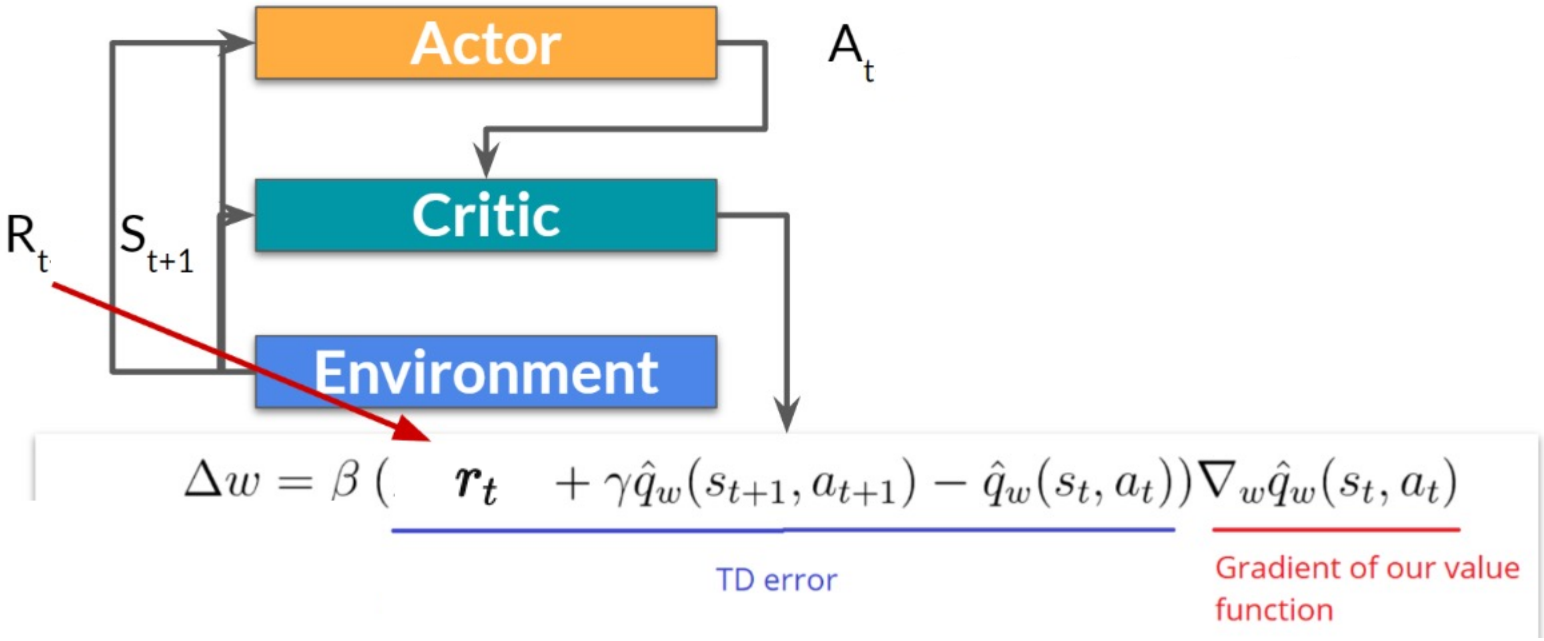
$$V(s) = \mathbb{E}_\pi[Q(s,a)] = \sum_a \pi(a|s) Q(s,a)$$

Method 2. **TD-error** (Temporal Difference error): $A(s_t, a_t) \approx \delta_t = r_t + \gamma \hat{q}_w(s_{t+1}, a_{t+1}) - \hat{q}_w(s_t, a_t)$

OR $r_t + \gamma V(s_{t+1}) - V(s_t)$

Actor-critic Reinforcement Learning

Step 5. Critic update: The Critic then updates its value parameters.



"TD error" (Temporal Difference error) refers to the difference between the current estimated value of a state and the updated value calculated based on the immediate reward received and the discounted value of the next state.

Peel the onion!

DeepSeek-R1

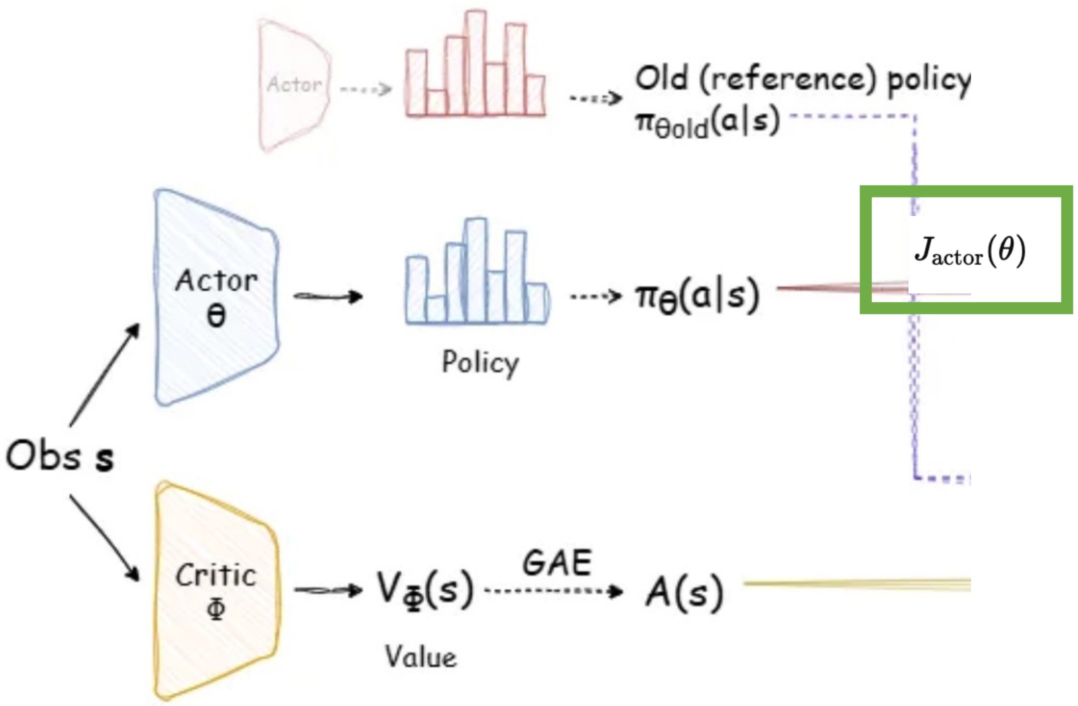
DeepSeek-R1-Zero

Group Relative Policy Optimization

Proximal Policy Optimization

Actor-critic Reinforcement Learning

PPO: Proximal Policy Optimization



Objective for the actor (policy model)

$$J_{\text{actor}}(\theta) = \mathbb{E}\left[\min\left(r_t A_t, \text{clip}(r_t, 1-\epsilon, 1+\epsilon) A_t\right)\right]$$

where:

- r_t → **Probability ratio** between the new policy and the old policy:

$$r_t = \frac{\pi_\theta(a_t|s_t)}{\pi_{\theta_{\text{old}}}(a_t|s_t)}$$

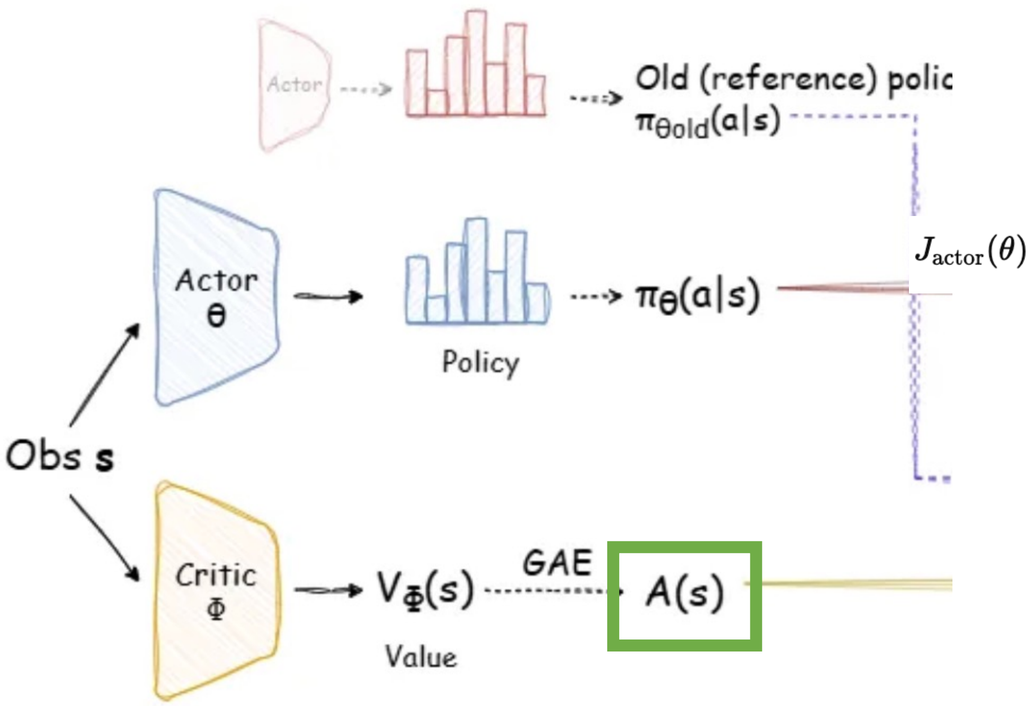
  - If $r_t > 1$, the new policy assigns a **higher probability** to the action compared to the old policy.
  - If $r_t < 1$, the new policy **decreases** the probability of the action.

- **Clipping Function:**

$$\text{clip}(r_t, 1-\epsilon, 1+\epsilon)$$

  - Ensures that r_t does not exceed the range $(1-\epsilon, 1+\epsilon)$.
  - **Prevents large policy updates**, making training more stable.

PPO: Proximal Policy Optimization



At(s) is the advantage estimate using GAE (Generalized Advantage Estimation)

$$A_t = \sum_{l=0}^{\infty} (\gamma\lambda)^l \delta_{t+l}$$

$\delta_t = r_t + \gamma V(s_{t+1}) - V(s_t)$ is the temporal difference (TD) residual.

Typical PPO Setting: $\lambda = 0.95$ is a commonly used default.

3. Guidelines for Tuning

- **If training is unstable (high variance in updates)** → Decrease λ slightly (e.g., from 0.95 to 0.90).
- **If learning is slow (high bias in value estimates)** → Increase λ (closer to 1.0).
- **If long-term planning is important** → Keep γ high (e.g., 0.99).
- **If your environment is short-horizon** → Lower γ (e.g., 0.95).

Peel the onion!

DeepSeek-R1

↑

DeepSeek-R1-Zero

↑

Group Relative Policy Optimization

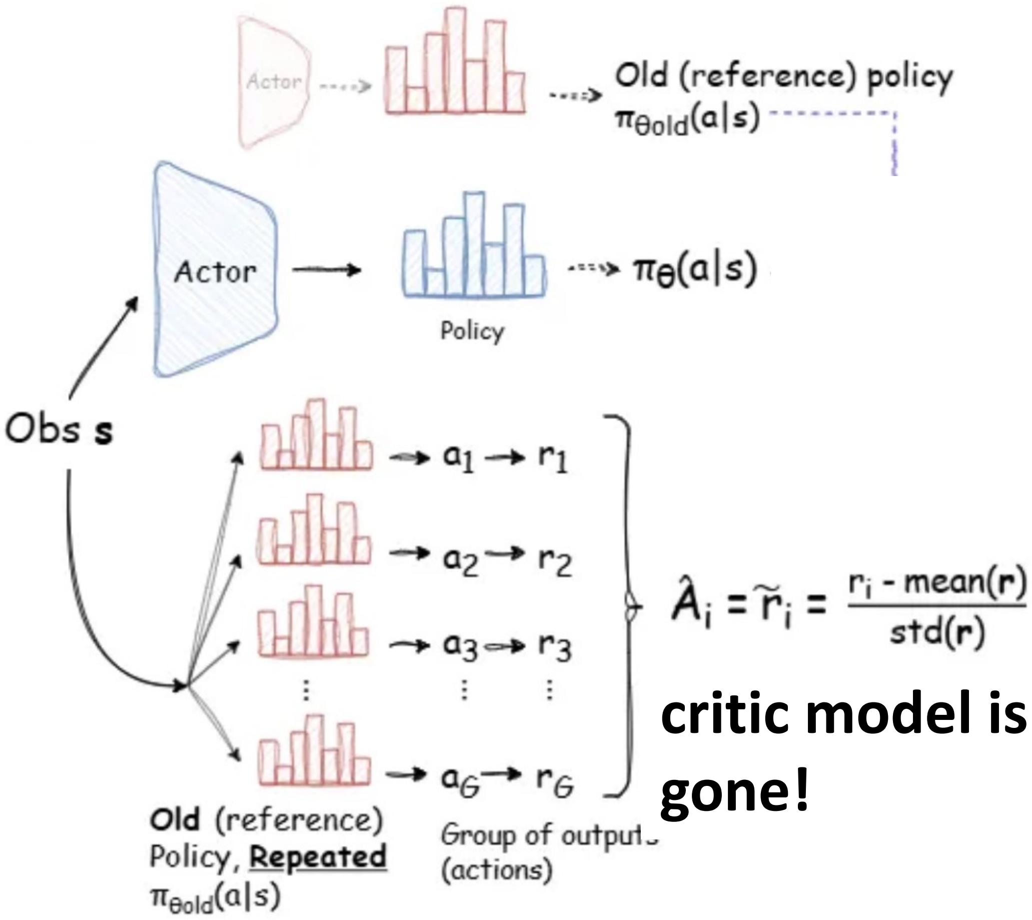
↑

Proximal Policy Optimization

↑

Actor-critic Reinforcement Learning

GRPO: Group Relative Policy Optimization



critic model is gone!

The **Group Relative Policy Optimization (GRPO)** objective for policy optimization is:

$$J_{\text{GRPO}}(\theta) = \mathbb{E}\left[\sum_{i=1}^{G}\sum_{t=1}^{|o_i|}\min\left(\frac{\pi_\theta(o_{i,t}|q, o_{i,<t})}{\pi_{\theta_{\text{old}}}(o_{i,t}|q, o_{i,<t})}\hat{A}_{i,t}, \text{clip}\right)\right] - \beta D_{KL}[\pi_\theta||\pi_{\text{ref}}]$$

Explanation of Terms:

1. $J_{\text{GRPO}}(\theta)$ **(Objective Function)**
   - This represents the optimization goal, where the policy π_θ is updated to maximize rewards while ensuring stability in training.
2. $\mathbb{E}$ **(Expectation)**
   - The expectation is taken over a batch of training data, meaning the optimization considers multiple sampled outputs to generalize well.
3. G **(Group Size)**
   - The number of sampled outputs for each input query q. Instead of using a single output, GRPO evaluates multiple responses to compute relative advantages. set as 4 to 64
4. o_i **(Sampled Outputs)**
   - Each q (input prompt) produces G different completions $o_1, o_2, ..., o_G$, which are compared to compute relative rewards.
5. t **(Token Position)**
   - The index of the token within an output sequence.

Peel the onion!

DeepSeek-R1

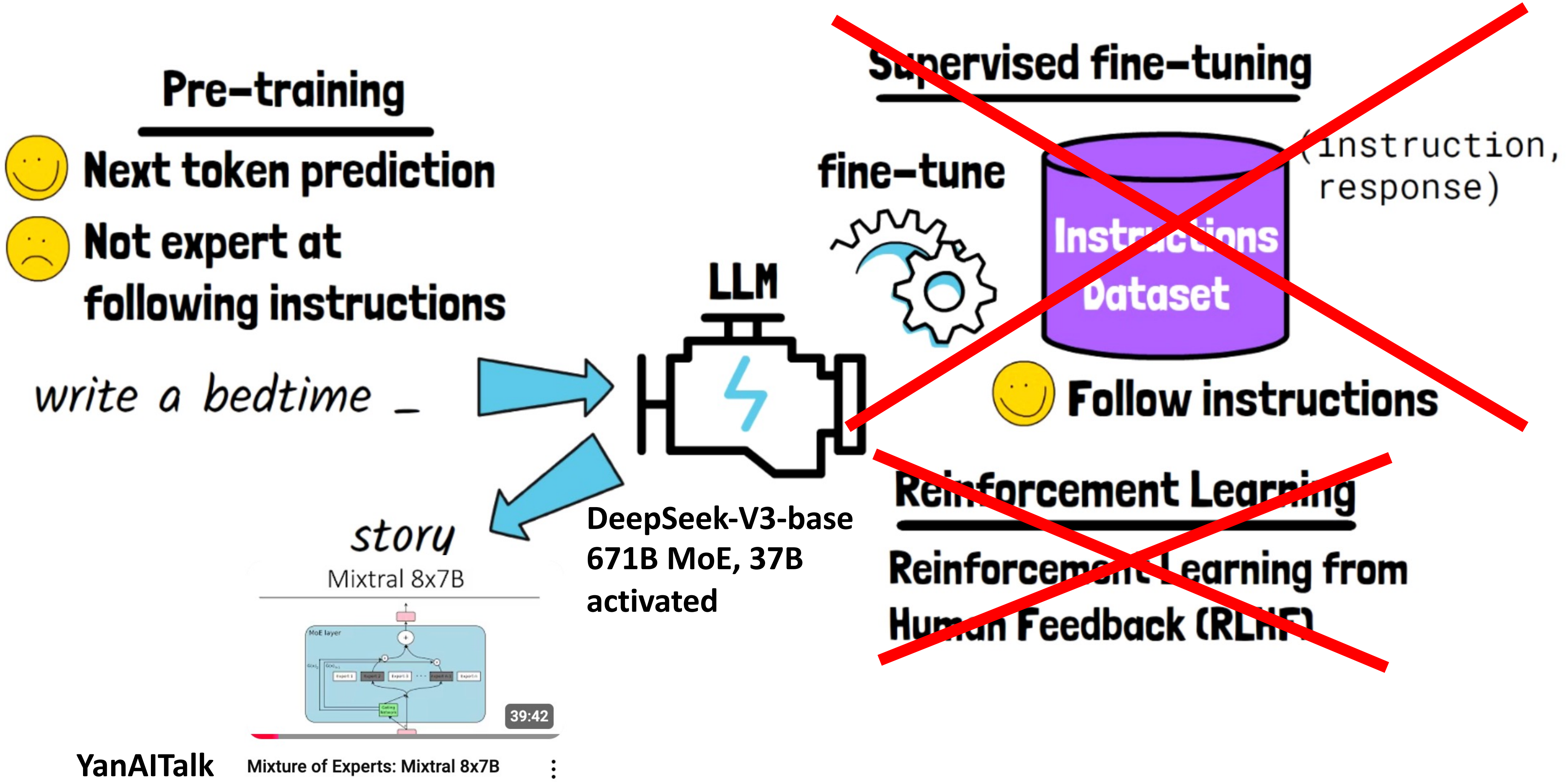
DeepSeek-R1-Zero

Group Relative Policy Optimization

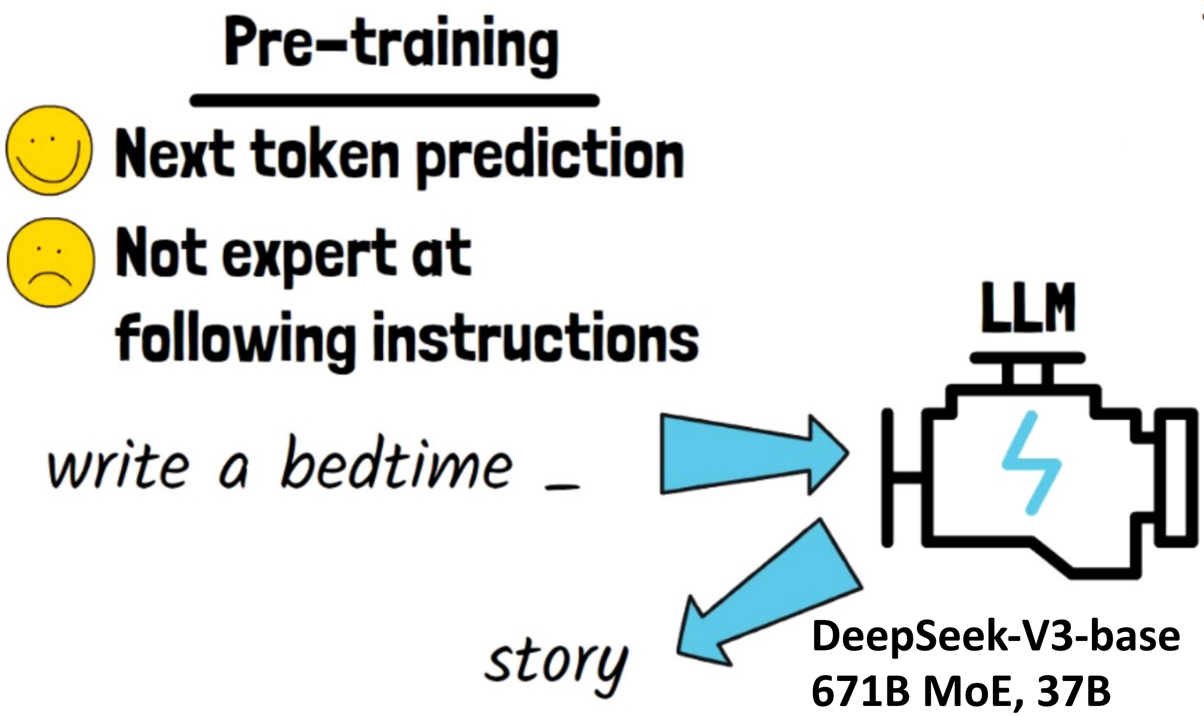
Proximal Policy Optimization

Actor-critic Reinforcement Learning

DeepSeek-R1-Zero



DeepSeek-R1-Zero



DeepSeek-V3-base
671B MoE, 37B
activated

Reinforcement Learning

- **Group Relative Policy Optimization (GRPO)** in order to save the training costs of RL
- **Simple Rule-based Reward Modeling**:

  Accuracy rewards: The accuracy reward model evaluates whether the response is correct.

  Format rewards: In addition to the accuracy reward model, we employ a format reward model that enforces the model to put its thinking process between tags, (e.g., <think> ... </think>).

DeepSeek-R1-Zero: Prompt template

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

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.

Peel the onion!

DeepSeek-R1

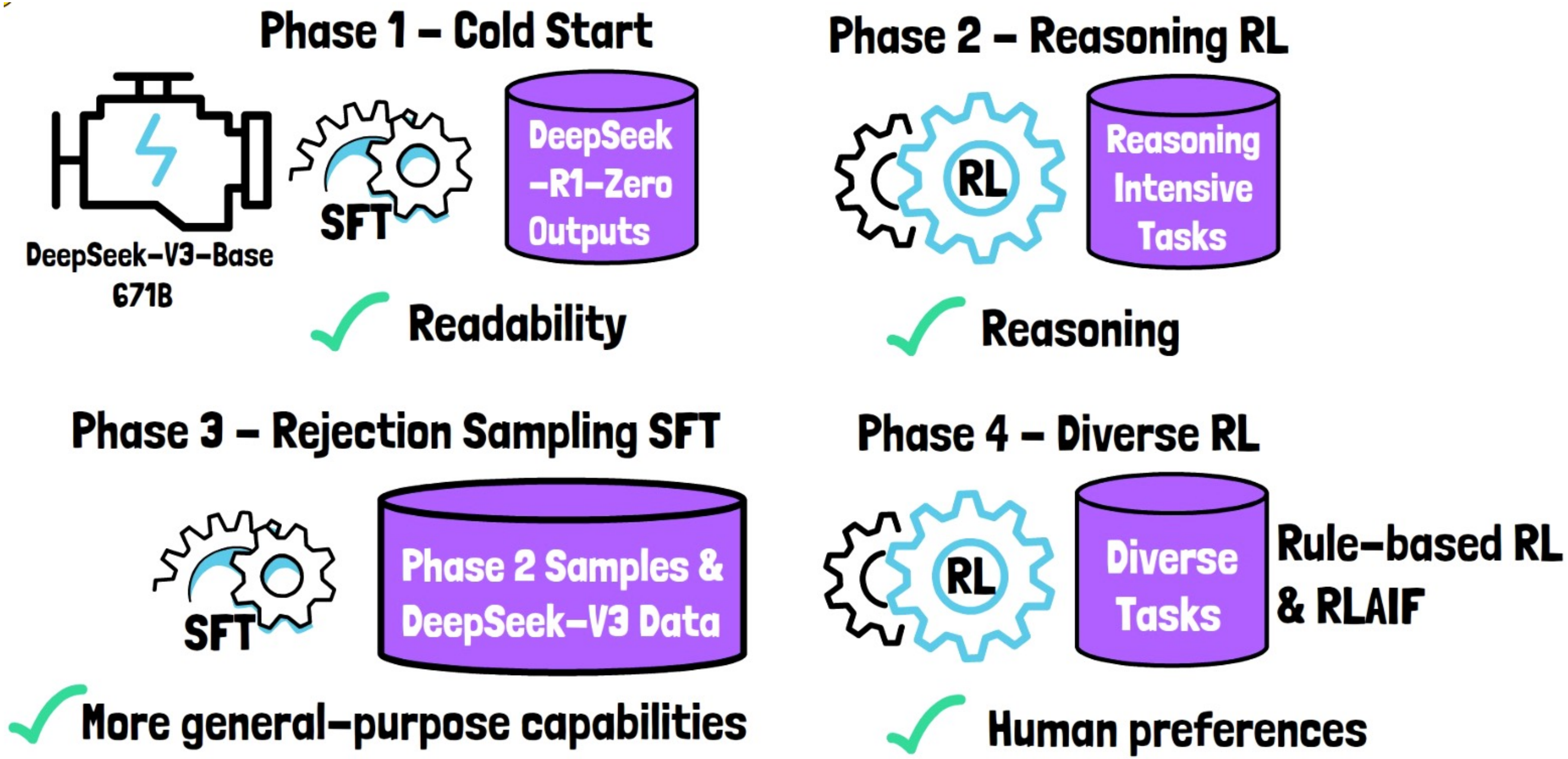
DeepSeek-R1-Zero

Group Relative Policy Optimization

Proximal Policy Optimization

Actor-critic Reinforcement Learning

DeepSeek R1 Overview



Phase 1. 1st SFT for cold start

- **Base model**: DeepSeek-V3-Base
- **1st supervised fine tuning (SFT) for cold start**: Use DeepSeek-R1-Zero to construct and collect a small amount of long CoT data (thousands) to fine-tune the model as the initial RL actor:
  - Few-shot prompting with a long CoT as an example, directly prompting models to generate detailed answers with reflection and verification
  - Gathering DeepSeek-R1-Zero outputs in a readable format, and refining the results through post-processing by human annotators.

Phase 2. 1st RL for reasoning

- **1st RL**: Same with DeepSeek-R1-Zero
  - Focus on reasoning-intensive tasks such as coding, mathematics, science, and logic reasoning, which involve well-defined problems with clear solutions
  - Reward:
    - **Accuracy rewards**: The accuracy reward model evaluates whether the response is correct.
    - **Format rewards**: In addition to the accuracy reward model, we employ a format reward model that enforces the model to put its thinking process between ‘’ and ‘’ tags.
    - **Language consistency**

Phase 3. 2nd SFT for both reasoning and non-reasoning

- When reasoning-oriented RL converges, utilize the resulting checkpoint to collect SFT (Supervised Fine-Tuning) data covering both reasoning and non-reasoning data:
  - **Reasoning data (600K):** Curate reasoning prompts and generate reasoning trajectories by performing rejection sampling from the checkpoint model.
  - Use a generative reward model for judgment by feeding the ground-truth and model predictions into DeepSeek-V3.
  - For each prompt, we sample multiple responses and retain only the correct ones. In total, we collect about 600k reasoning related training samples.
  - **Non-Reasoning data (200K):** For non-reasoning data, such as writing, factual QA, self-cognition, and translation, reuse portions of the SFT dataset of DeepSeek-V3.

Phase 4. 2nd RL for both reasoning and human preference

- **2nd RL**: Train a model for all scenarios including reasoning and human preferences
  - The same Rule-based rewards as in 1st RL to guide the learning process in math, code, and logical reasoning domains.
  - Add rewards to capture human preferences in complex and nuanced scenarios.
    - For helpfulness, focus exclusively on the final summary emphasizing the utility and relevance of the response to the user
    - For harmlessness, evaluate the entire response of the model, including both the reasoning process and the summary

	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	**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
	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**
Code	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
	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
Math	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.

Distillation into small models from R1

- Distillation leads to better reasoning than direct RL on the smaller models: Directly fine-tuned open-source models like Qwen and Llama using the 800k samples curated with DeepSeek-R1.
  - For distilled models, only apply SFT and no RL, even though incorporating RL could substantially boost model performance.

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.

Conclusion

Recording posted @

YanAITalk

@yanaitalk · 2.49K subscribers · 62 videos

Make machine learning easy to understand! ...more

Customize channel | Manage videos

deepseek

DeepSeek-R1

DeepSeek-R1-Zero

Group Relative Policy Optimization

Proximal Policy Optimization

Actor-critic Reinforcement Learning