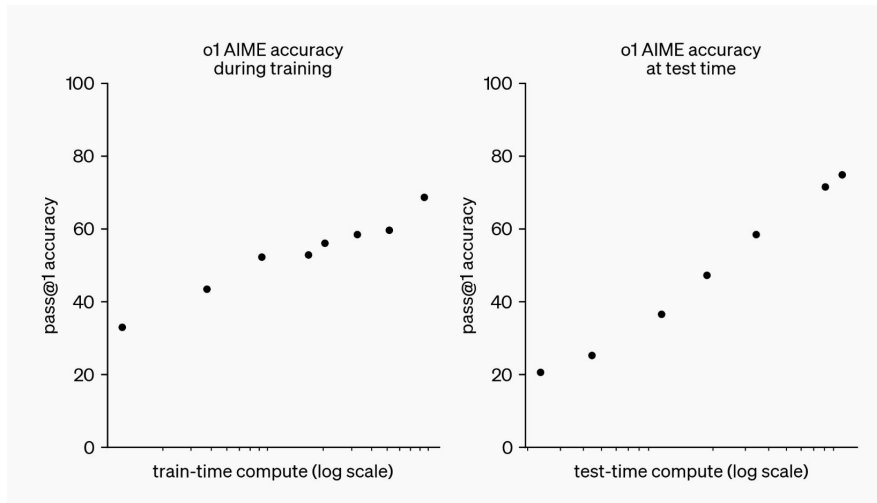


Reinforcement Learning inside DeepSeek-R1

Tianbing Xu
Jan. 2015

Learning to reason with LLMs (OpenAI 2024)

Our large-scale **reinforcement learning** algorithm teaches the model how to think productively using its **chain of thought** in a highly *data-efficient training* process.



Deepseek-R1

Benchmark Results (DeepSeek-AI 2025)

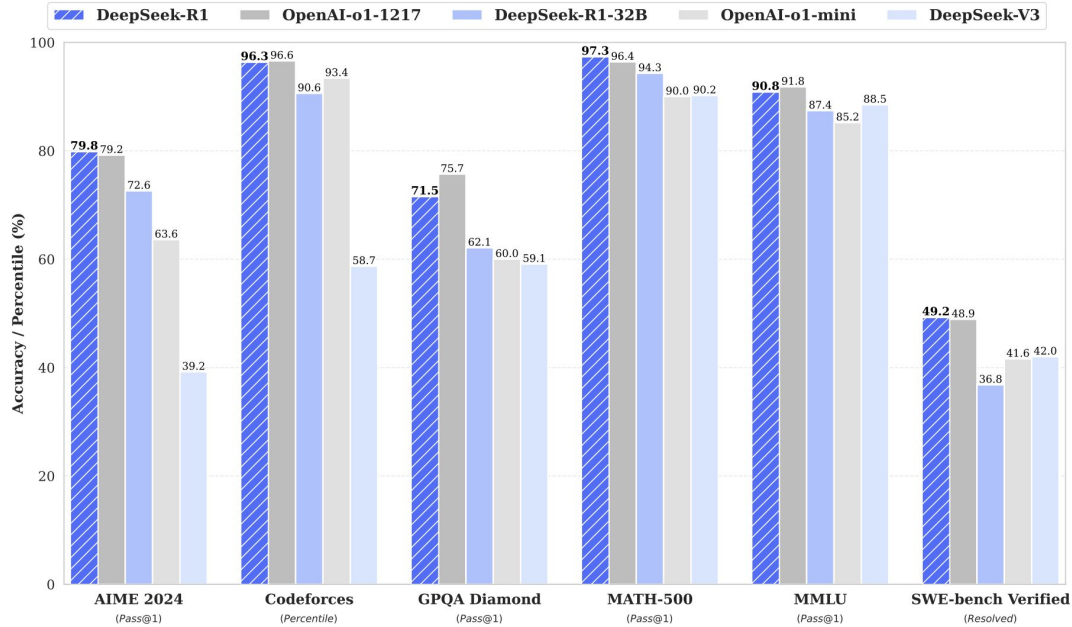


Figure 1 | Benchmark performance of DeepSeek-R1.

Large-Scale Reinforcement Learning (GRPO)

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

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

RL: Learning from Scratch

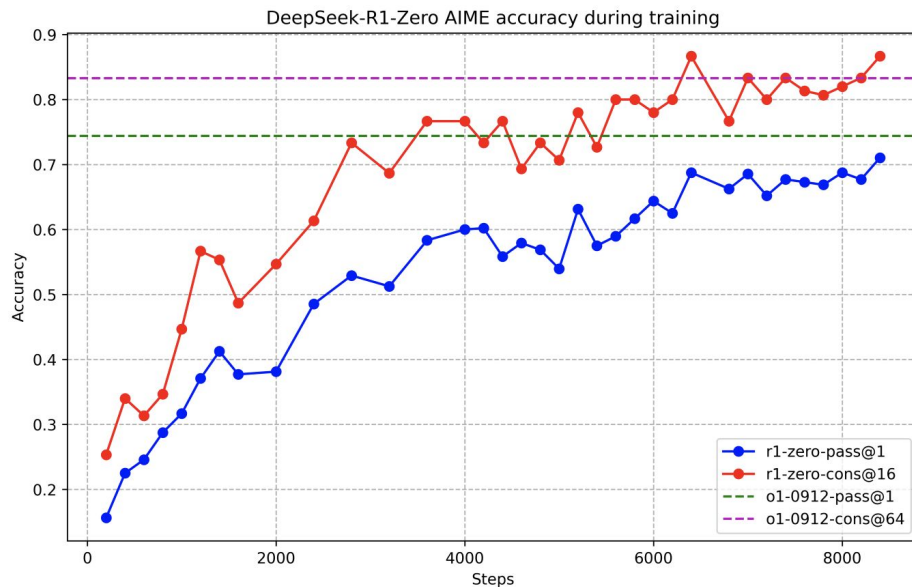


Figure 2 | AIME accuracy of DeepSeek-R1-Zero during training. For each question, we sample 16 responses and calculate the overall average accuracy to ensure a stable evaluation.

Learning to improve reasoning with more thinking

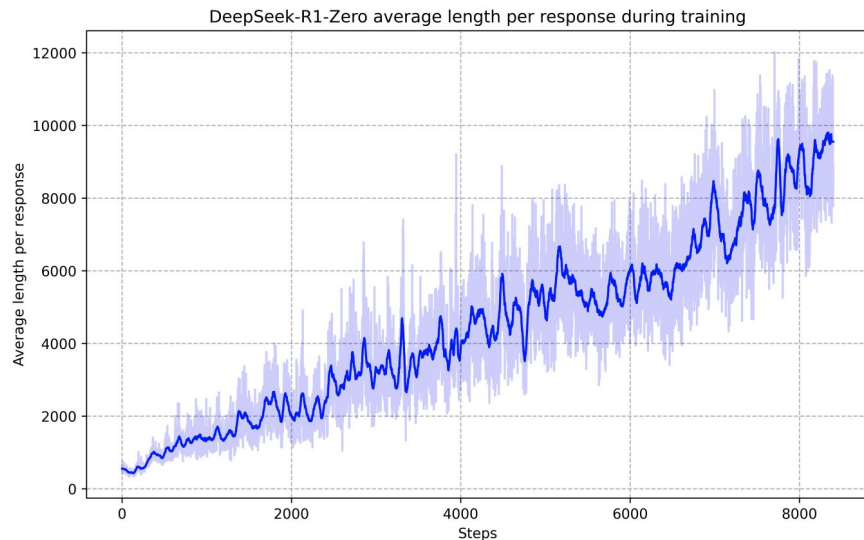


Figure 3 | The average response length of DeepSeek-R1-Zero on the training set during the RL process. DeepSeek-R1-Zero naturally learns to solve reasoning tasks with more thinking time.

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 ...

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

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:

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

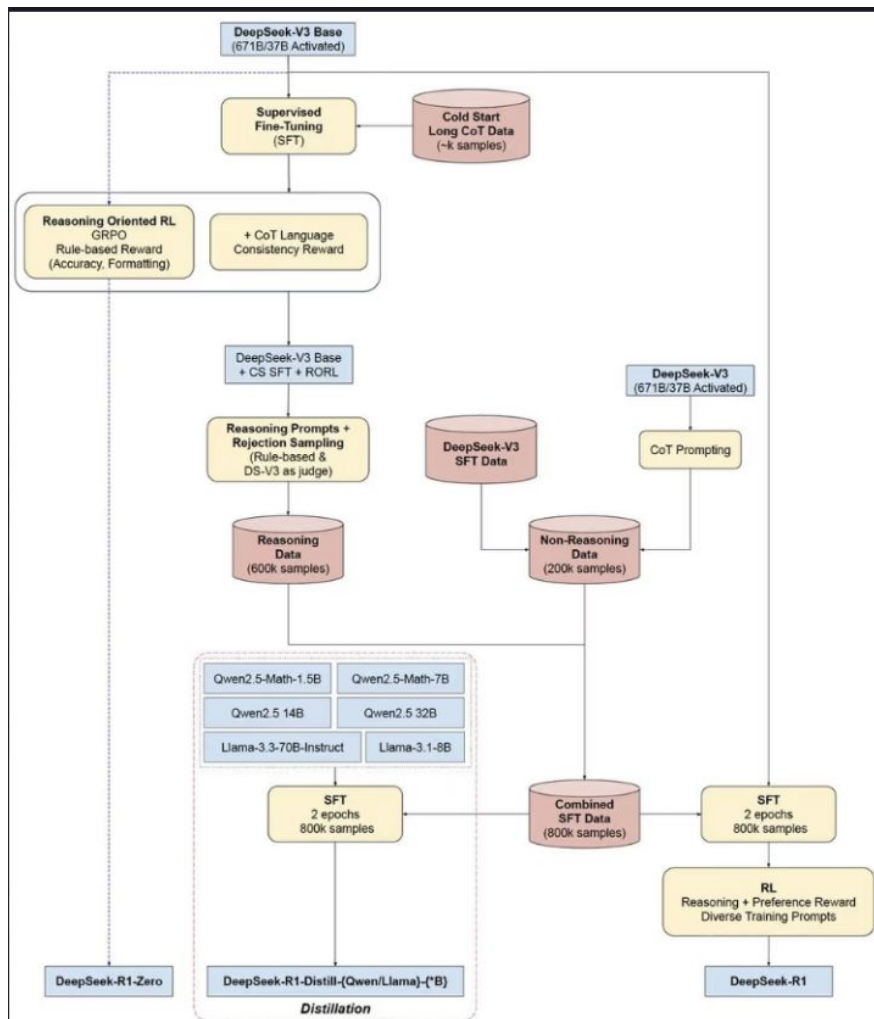
First, let's square both sides:

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

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.



Reinforcement Learning

Reinforcement Learning

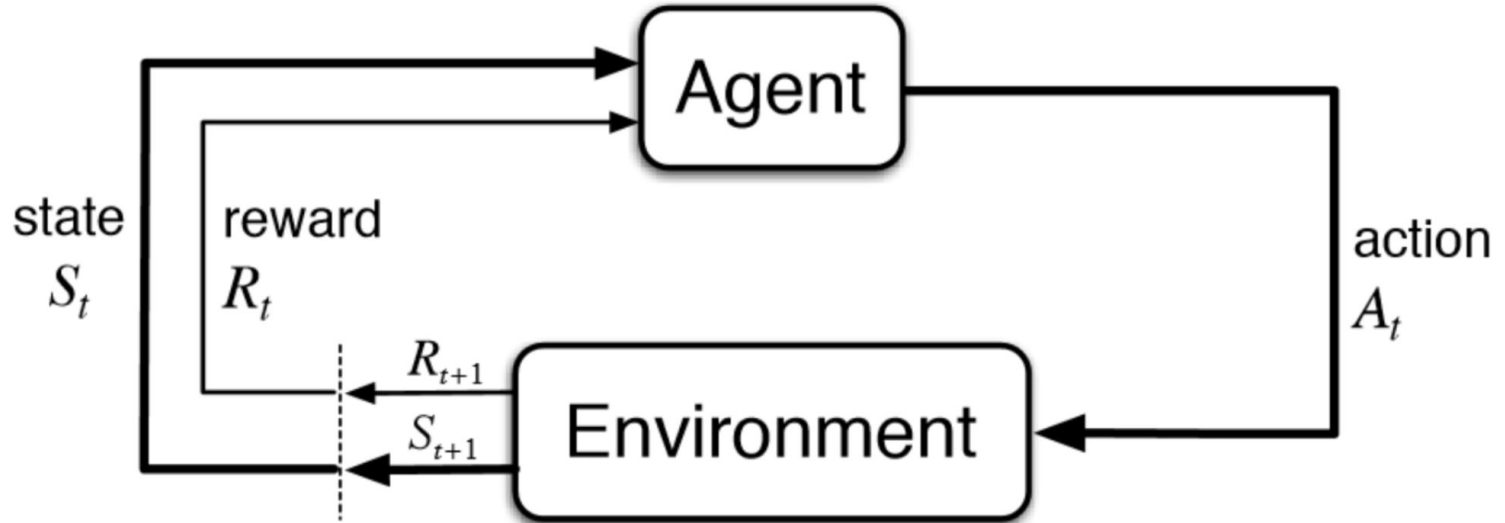


Figure: RL framework

Vanilla Policy Gradient

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

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

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

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

.

The policy gradient is then computed as:

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

Proximal Policy Optimization (PPO, Schulman, 2017)

$$L_{\text{PPO}}(\theta) = \mathbb{E}_{\pi_{\theta_{\text{old}}}} \left\{ \min \left(\frac{\pi_{\theta}}{\pi_{\theta_{\text{old}}}} A(s, a), \text{clip} \left(\frac{\pi_{\theta}}{\pi_{\theta_{\text{old}}}}, 1 - \epsilon, 1 + \epsilon \right) A(s, a) \right) \right\}, \quad (3)$$

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

Group Relatively Policy Optimization (GRPO, DeepSeek, 2025)

$$L_{\text{GRPO}}(\theta) = \mathbb{E}_{\pi_{\theta_{\text{old}}}} \left\{ \min \left(\frac{\pi_{\theta}}{\pi_{\theta_{\text{old}}}} A(s, a), \text{clip} \left(\frac{\pi_{\theta}}{\pi_{\theta_{\text{old}}}}, 1 - \epsilon, 1 + \epsilon \right) A(s, a) \right) \right\} \\ - \beta \mathbb{E}_{\pi_{\theta_{\text{old}}}} [D_{\text{KL}}(\pi_{\theta} \parallel \pi_{\text{ref}})], \quad (4)$$

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

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

Tricks in GRPO (Simplification and Optimization)

Simplified Policy Gradient (Vanilla Policy Gradient + Unbiased KL Estimation)
Variance Reduction, Computational Saving

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

Tricks in GRPO (Simplification and Optimization)

Removal of Value Model to Estimate Advantage Function (**Memory Saving, Convergence Acceleration**)

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

$$\mathbf{r} = \{r_j \mid j = 1, \dots, K\}$$

collected from trajectories induced by π_θ , helping stabilize the RL training process.

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

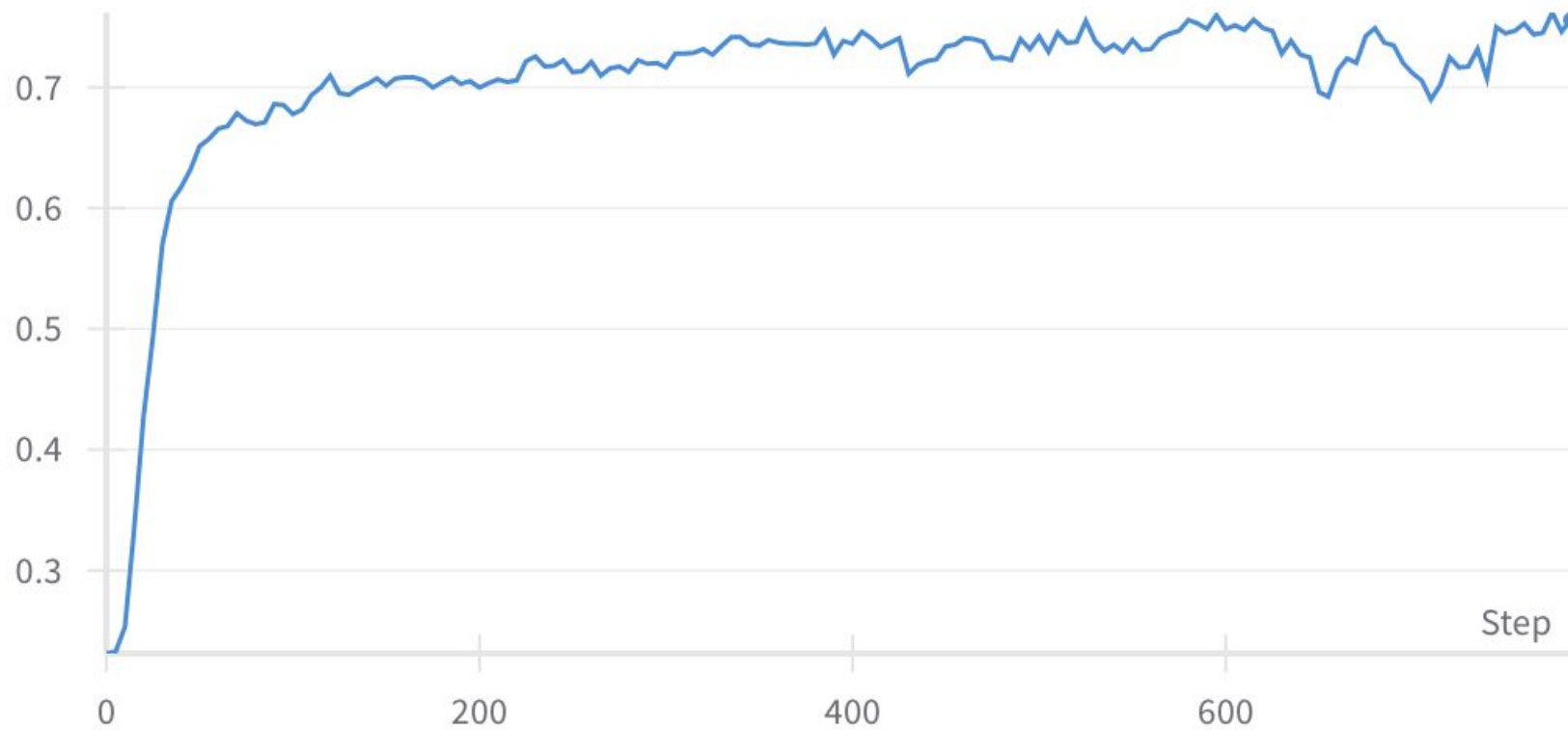
My Experiment: Running GRPO on Small Model

BASE_MODEL:Qwen/Qwen2.5-1.5B

DATA: gsm8k

4 * A10 GPU

val/test_score/openai/gsm8k



```

^[[36m(main_task pid=59766)^[[0m validation generation end
^[[36m(main_task pid=59766)^[[0m John brings his dog to the vet. His dog needs 2 vaccines, which are $20 each, and a heartworm check. The heartworm check is 60% of his total bill. If he brought $125 with him, how much does he leave with? Let's think step by step and output the final answer after "####". Given:
^[[36m(main_task pid=59766)^[[0m John brings his dog to the vet and his dog needs 2 vaccines, which are $20 each, and a heartworm check.
^[[36m(main_task pid=59766)^[[0m
^[[36m(main_task pid=59766)^[[0m Step 1: Calculate the cost of the vaccines.
^[[36m(main_task pid=59766)^[[0m John needs 2 vaccines, and each vaccine costs $20.
^[[36m(main_task pid=59766)^[[0m So, the total cost of the vaccines =  $2 * \$20 = \$40$ 
^[[36m(main_task pid=59766)^[[0m
^[[36m(main_task pid=59766)^[[0m Step 2: Calculate the cost of the heartworm check.
^[[36m(main_task pid=59766)^[[0m The heartworm check is 60% of the total bill.
^[[36m(main_task pid=59766)^[[0m So, the cost of the heartworm check = 60% of ( $\$40 + \text{cost of heartworm check}$ )
^[[36m(main_task pid=59766)^[[0m Let's denote the cost of the heartworm check as "H".
^[[36m(main_task pid=59766)^[[0m So,  $H = 60\% \text{ of } (\$40 + H)$ 
^[[36m(main_task pid=59766)^[[0m  $H = 0.6 * (\$40 + H)$ 
^[[36m(main_task pid=59766)^[[0m  $H = 0.6 * \$40 + 0.6 * H$ 
^[[36m(main_task pid=59766)^[[0m  $H = \$24 + 0.6 * H$ 
^[[36m(main_task pid=59766)^[[0m  $0.4 * H = \$24$ 
^[[36m(main_task pid=59766)^[[0m  $H = \$24 / 0.4$ 
^[[36m(main_task pid=59766)^[[0m  $H = \$60$ 
^[[36m(main_task pid=59766)^[[0m
^[[36m(main_task pid=59766)^[[0m Step 3: Calculate the total bill.
^[[36m(main_task pid=59766)^[[0m The total bill = cost of vaccines + cost of heartworm check
^[[36m(main_task pid=59766)^[[0m Total bill =  $\$40 + \$60 = \$100$ 
^[[36m(main_task pid=59766)^[[0m
^[[36m(main_task pid=59766)^[[0m Step 4: Calculate the amount John leaves with.
^[[36m(main_task pid=59766)^[[0m John brought $125 with him.
^[[36m(main_task pid=59766)^[[0m Amount left =  $\$125 - \text{Total bill}$ 
^[[36m(main_task pid=59766)^[[0m Amount left =  $\$125 - \$100$ 
^[[36m(main_task pid=59766)^[[0m Amount left = $25
^[[36m(main_task pid=59766)^[[0m
^[[36m(main_task pid=59766)^[[0m So, John leaves with $25.
^[[36m(main_task pid=59766)^[[0m #### 25
^[[36m(main_task pid=59766)^[[0m The final answer is: 25
^[[36m(main_task pid=59766)^[[0m The final answer is: 25</endoftext|>

```

Why RL is the Gateway to AGI ?

- **Goal-Oriented Learning, Long-term Planning**
- **Learning from Data Auto-Generated via Interaction with Complex Environments**
- **Search, Exploration and Discovery Novel Strategies**
- **Trial and Error, Backtracking and Learning from Mistakes**
- **Autonomy and Continuous Improvement**

Conclusion and Discussion

*One thing that should be learned from the bitter lesson is the great power of **general purpose methods**, of methods that continue to **scale with increased computation** even as the available computation becomes very great. The two methods that seem to scale arbitrarily in this way are **search** and **learning**.*

The Bitter Lesson: Rich Sutton

Reference

1. [Learning to reason with LLMs](#)
2. [DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning](#)
3. [*Proximal Policy Optimization Algorithms*](#)
4. [From PPO to GRPO: A Policy Gradient Approach within DeepSeek R1](#)
5. [GRPO Experiment on Small Model](#)