Training Language Models to Reason - II

Advances in Large Language Models

ELL8299 · AIL861

Gaurav Pandey
Senior Research Scientist, IBM Research

The Flow

Chain-of-Thought (CoT) prompting - inference time only

STaR (Self-Taught Reasoner) – SFT on self-generated thoughts

TPO (Thought Preference Optimization) – DPO on self-generated thoughts

DeepSeek-R1-zero – GRPO with verifiable reward

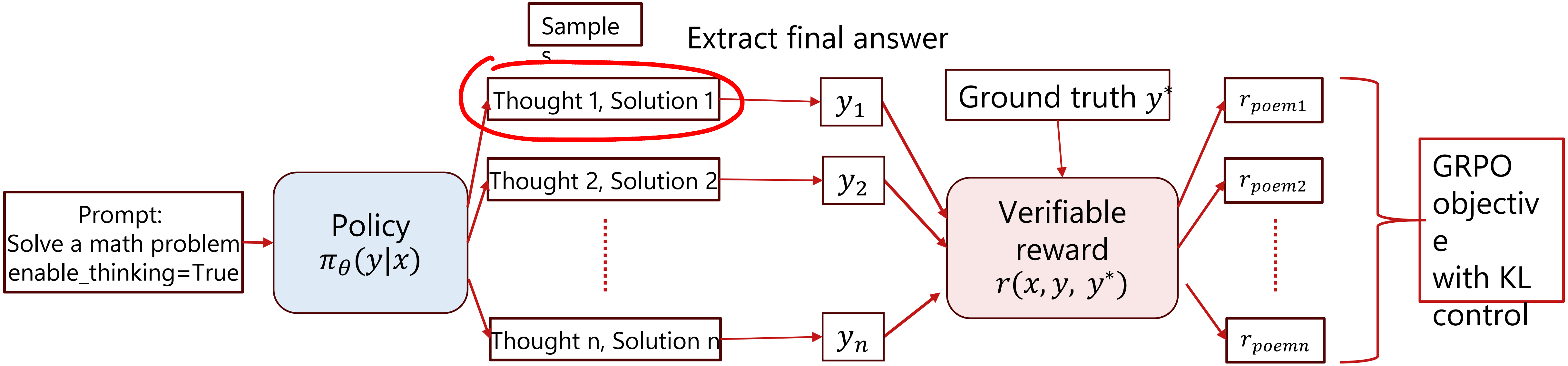
Reasoning in weaker models - distillation, MCTS + self-training

Verifiable rewards

- Rewards that can be computed objectively and reproducibly from a ground truth.
- Examples of Verifiable Reward Functions
  - **Math:** Exact numerical answer match
  - **Code:** Passes all test cases
  - **QA:** String match or F1-score over entities
  - **Formal logic tasks:** Correct proof sequence
  - **Chemistry:** Exact Match in Reaction Prediction
  - **Biology:** RMSD for Protein structure prediction
- Does not depend on noisy human or AI preferences.
- Responsible for the latest revolution in reasoning in AI

  (Deepseek-R1, OpenAI o1, Kimi K1.5 and Kimi K2, Qwen3)

RL with verifiable rewards



$r_1 \log \pi_\theta(y_1|x) + \;\ldots\; r_n \log \pi_\theta(y_n|x)$

The GRPO objective – a quick recap

$$L^{GRPO}(\theta) = E\left[\min\left(r_t(\theta)\widehat{A}(y_i),\ clip(r_t(\theta),\ 1-\epsilon,\ 1+\epsilon)\widehat{A}(y_i)\right)\right]$$

GRPO

Sampling – $\pi_{\theta_{old}}$.

$$r_t(\theta) = \frac{\pi_\theta(y|x)}{\pi_{\theta_{old}}(y|x)}$$

The GRPO objective – a quick recap

$$L^{GRPO}(\theta) = E\left[\min\left(r_t(\theta)\widehat{A}(y_i),\ clip(r_t(\theta),\ 1-\epsilon,\ 1+\epsilon)\widehat{A}(y_i)\right)\right]$$

- $r_t(\theta)$ is the probability ratio $\frac{\pi_\theta(a_t|s_t)}{\pi_{\theta_{old}}(a_t|s_t)}$
    - a_t is the tth token of the generated output
    - s_t is the tokens before a_t
    - $\pi_{\theta_{old}}$ is the distribution from which y_i was sampled.
- Key trick of GRPO - Advantage is relative

$$A(y_i) = \frac{r(y_i\ ) - mean\left(r(y_1), \ldots, r(y_K)\right)}{stddev\left(r(y_1), \ldots, r(y_K)\right)}$$

Deepseek-R1-zero

there is no SFT

- Uses a strong base model – Deepseek-V3-base
  - Extensive Math and Code data – sources not stated
- **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> </answ~~er> tags, respectively, i.e., <think> reasoning process~~ here ~~</think> <answer> answer here~~ </answer>. User: prompt. Assistant:

- Uses GRPO with RLVR for Maths & Coding problems

Reward Modelling

- Accuracy reward for Math
  - Instruct the model to put the final output in a specific format
  - Extract the output from the generated response
  - Verify against the ground truth answer.
- Accuracy reward for Code
  - Run unit tests on the generated code
- Format reward
  - The format should look like

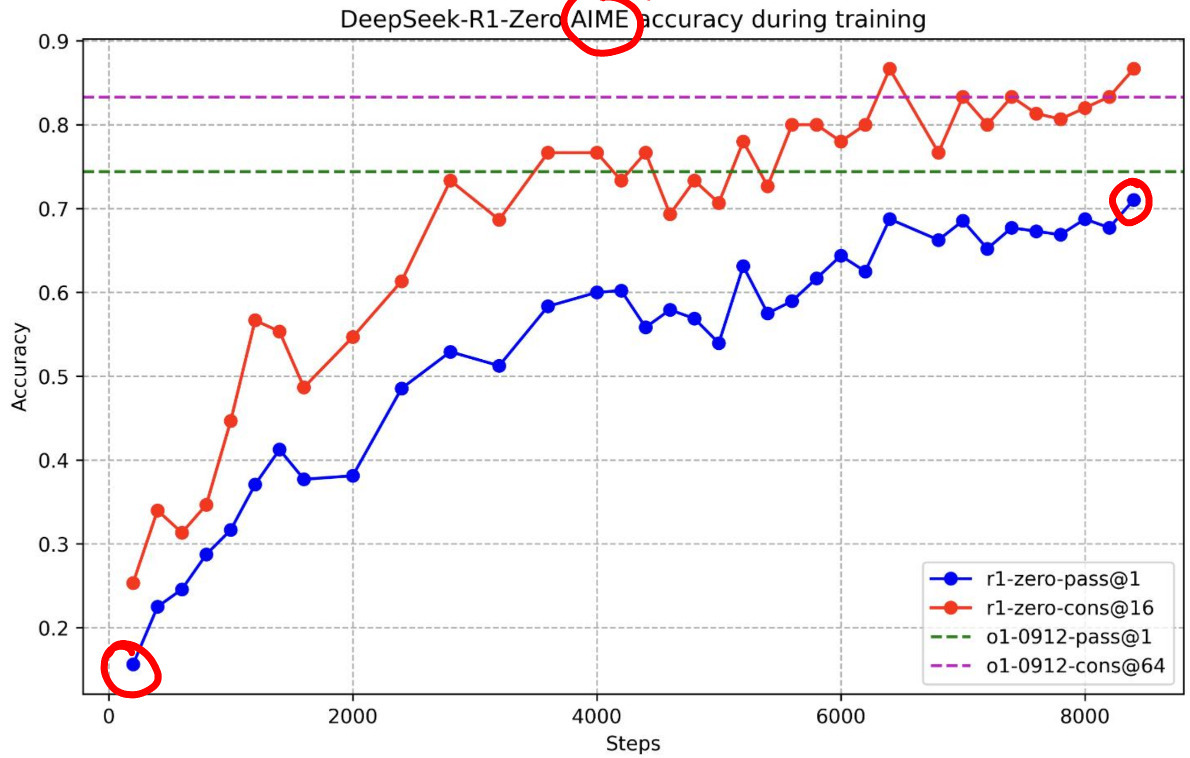
$$< think > Thought\ process </think >< response > \ Response \ </response >$$

+1, -1

RL training - performance

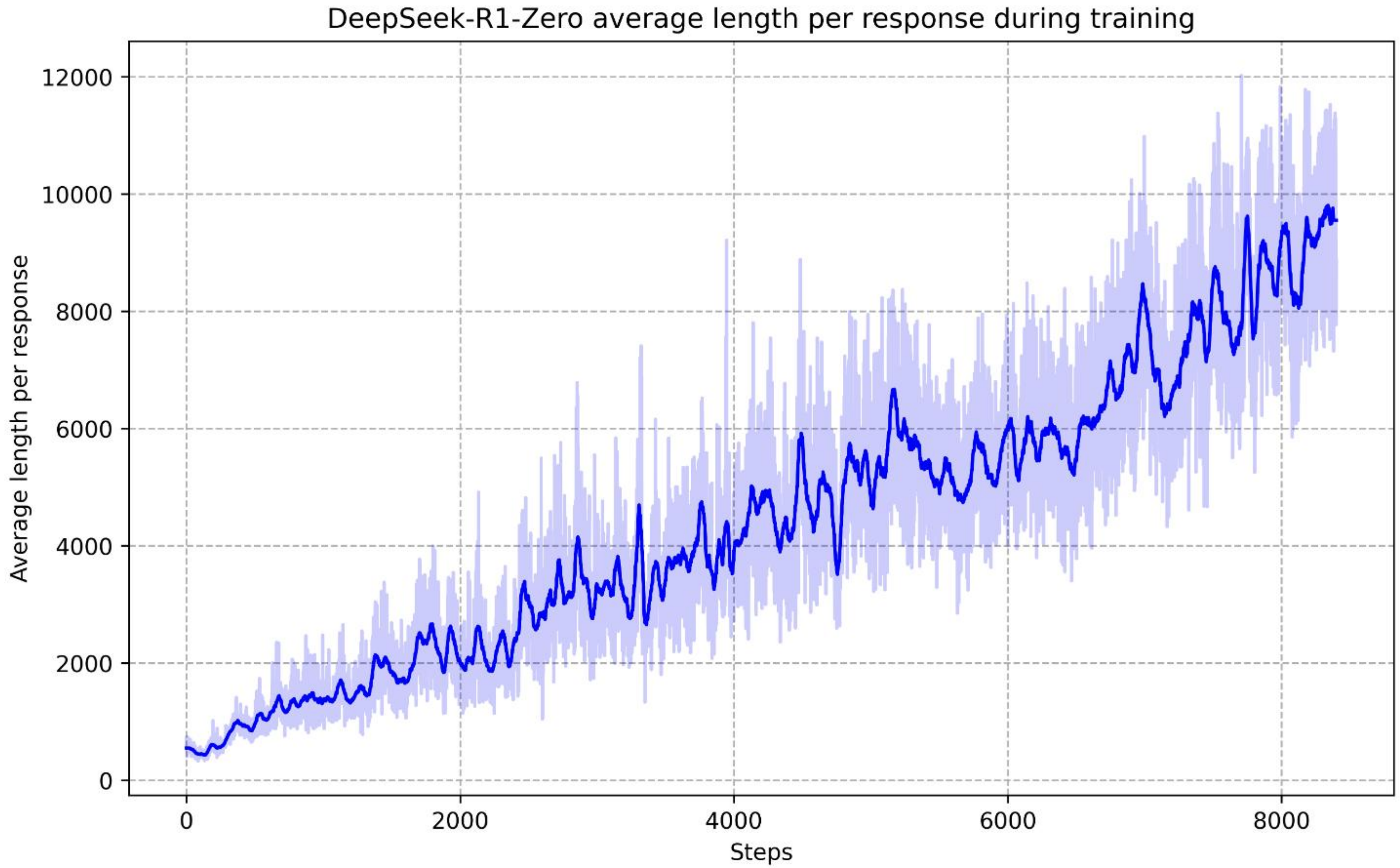
- GRPO on top of math and code data Olympiad math problems 1983 - 2023

2024 & 2025 Evaluation



AIME accuracy of Deepseek-R1-Zero increases with the number of RL steps

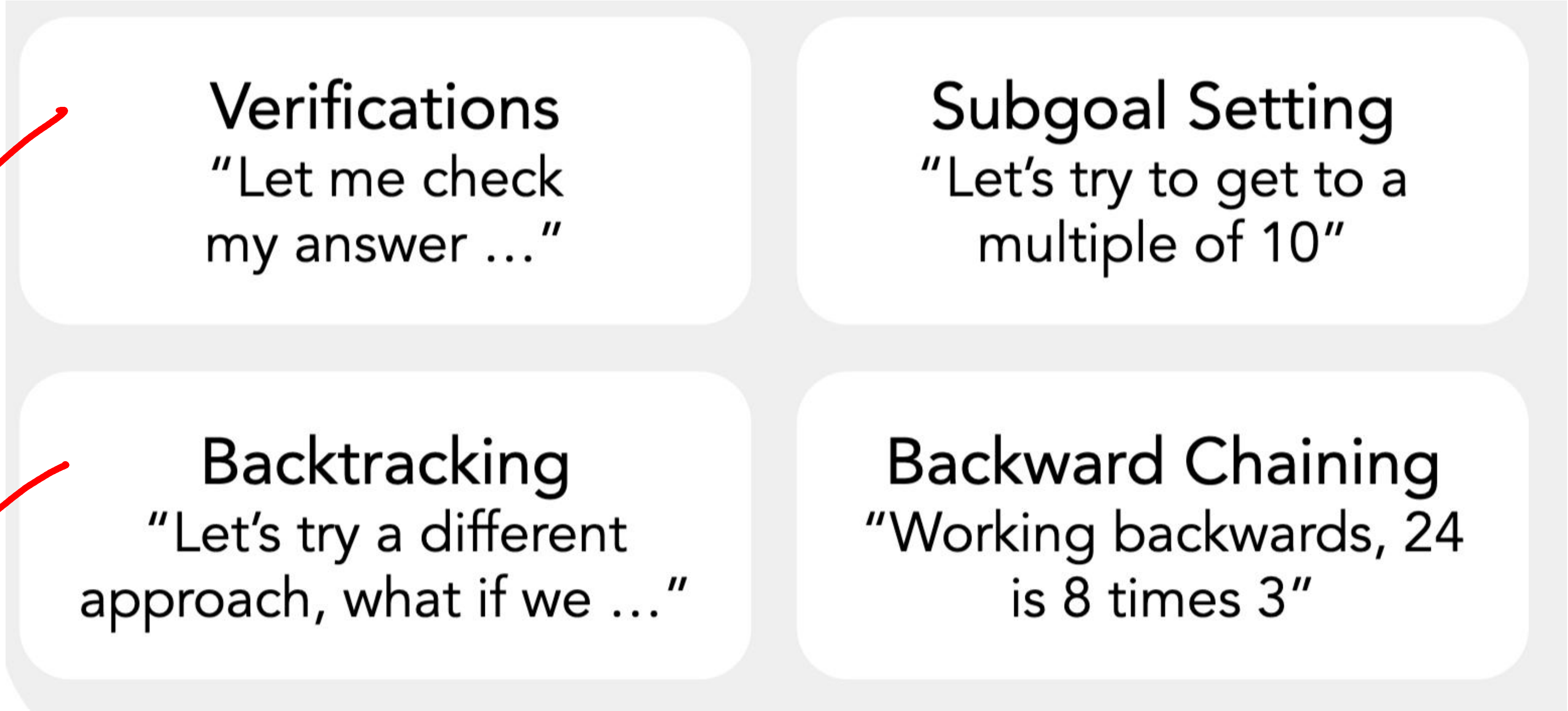
RL training – length of responses



Length of responses increases with the number of RL steps

Why do the lengths increase?

- Emergence of new reasoning patterns – the aha moment



Gandhi et al. Cognitive Behaviors that Enable Self-Improving Reasoners. arXiv 2025

An example

Problem:

James writes a 3-page letter to 2 different friends twice a week. How many pages does he write a year?

multiple solutions

verification

Thinking Process

<think>
Okay, let me try to figure out this problem. So, James is writing a 3-page letter to two different friends twice a week. We need to find out how many pages he writes in a year. **Hmm**, let's break this down step by step.
First, **let me understand the details**. [...]
Wait, if he writes to 2 friends twice a week, does that mean he writes 2 letters each time, and he does this twice? [...] So per week, he writes 2 friends * 3 pages * 2 times. **Let me check that.** [...]
Alternatively, maybe it's 3-page letter to 2 different friends each time he writes, and he does that twice a week. So each time he writes, he's writing two letters (each 3 pages), so that's 2 * 3 = 6 pages per writing session. Then, twice a week would be 6 * 2 = 12 pages per week. Then, multiply that by the number of weeks in a year. But how many weeks are in a year? Typically, 52 weeks. So 12 pages/week * 52 weeks = 624 pages. **Is that right?**
Wait, let me confirm. So, per week: [...] **That makes sense.**
But maybe **I need to double-check** if the problem is interpreted differently. [...] That seems consistent.
Wait, but wait a second. Is there any chance that the problem might be considering a year as 52 weeks, **but some people might account for leap years or something else?** But generally, in problems like this, unless specified, we take 52 weeks per year. So that's standard.
So, conclusion: [...] So the answer is 624. So I think that's correct.
</think>

Answer

James writes 3-page letters to 2 friends twice a week. Here's the breakdown:
[...]
Answer: James writes $\boxed{\text{624 pages in a year.}}$.

Figure from: Marjanovi´c et al. DeepSeek-R1 Thoughtolgy. arXiv 2025.

The Flow

Chain-of-Thought (CoT) prompting - inference time only

STaR (Self-Taught Reasoner) – SFT on self-generated thoughts

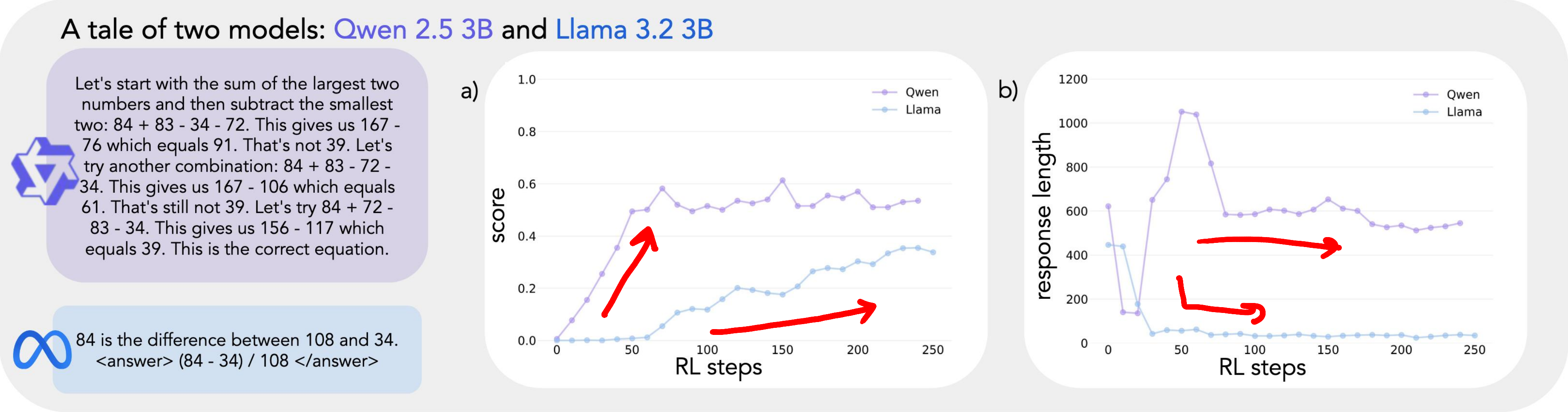
TPO (Thought Preference Optimization) – DPO on self-generated thoughts

DeepSeek-R1-zero – GRPO with verifiable reward

Reasoning in weaker models - distillation, MCTS + self-training

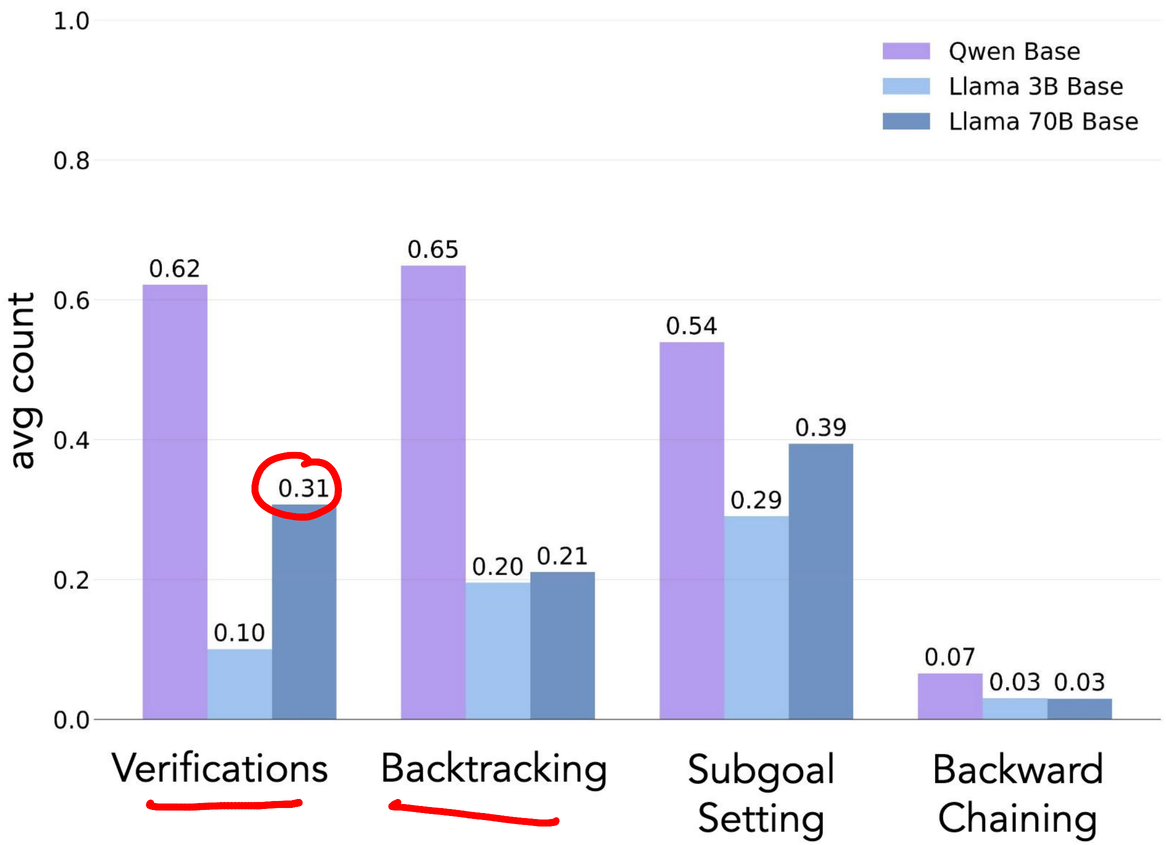
Will this work on all models?

- Short answer, no
- The model must seen have such patterns during its pretraining.
- Countdown game — a mathematical puzzle where players must combine a set of input numbers using the four basic arithmetic operations (+, −, ×, ÷) to reach a target number



Gandhi et al. Cognitive Behaviors that Enable Self-Improving Reasoners. arXiv 2025

The role of initial behavior

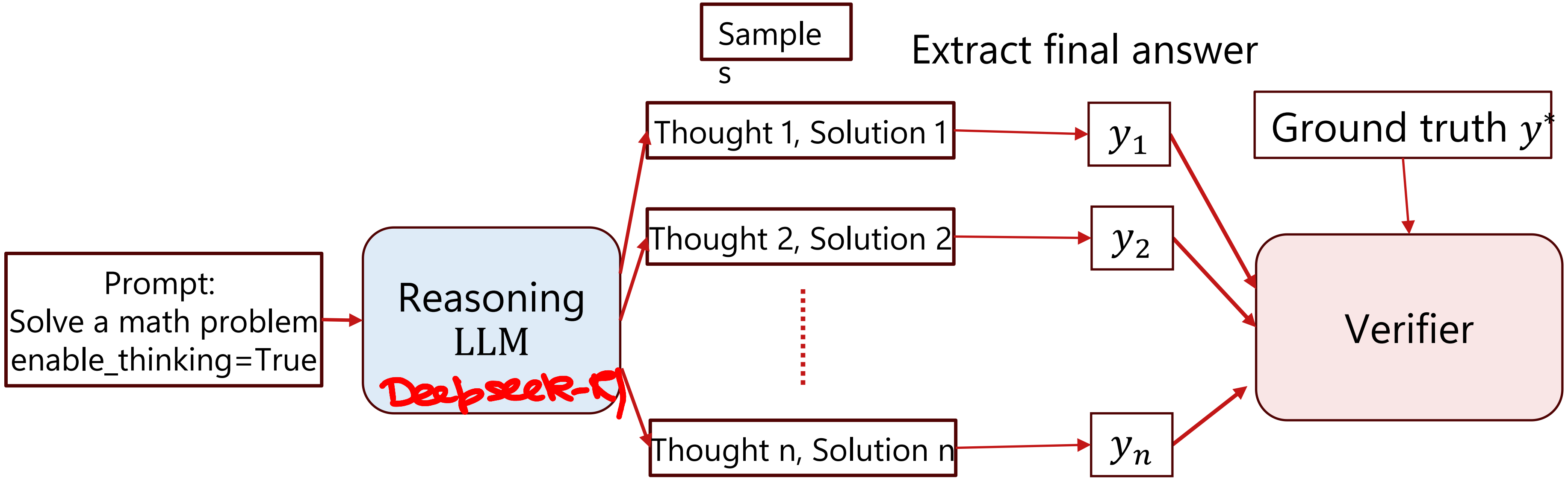


- Qwen-2.5-3B models already exhibit all the 4 behaviors at a much higher rate than Llama-70B.
- The initial policy must show the cognitive behavior for RL to exploit it.

Gandhi et al. Cognitive Behaviors that Enable Self-Improving Reasoners. arXiv 2025

Inducing reasoning patterns in weaker models

For verifiable problems



- Keep verified (prompt, thought, solution) triplets
- Perform SFT on these triplets

Distillation vs RL for weak models

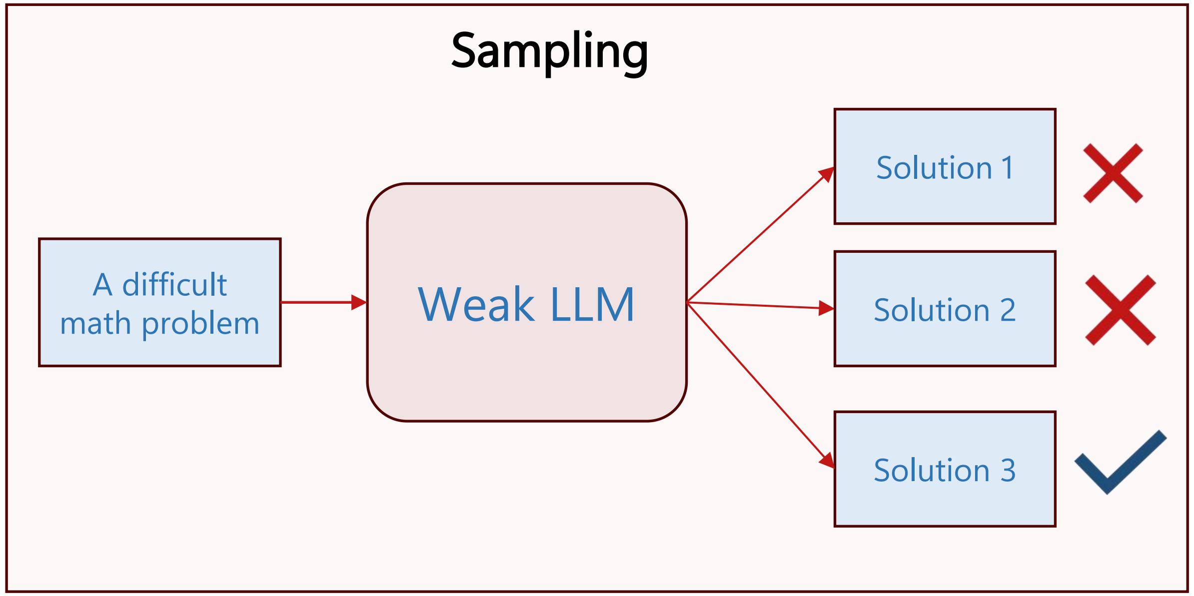
- Always prefer distillation from a powerful reasoning model.
- RL on weak models may never be able to find trajectories that have already been discovered by more powerful models

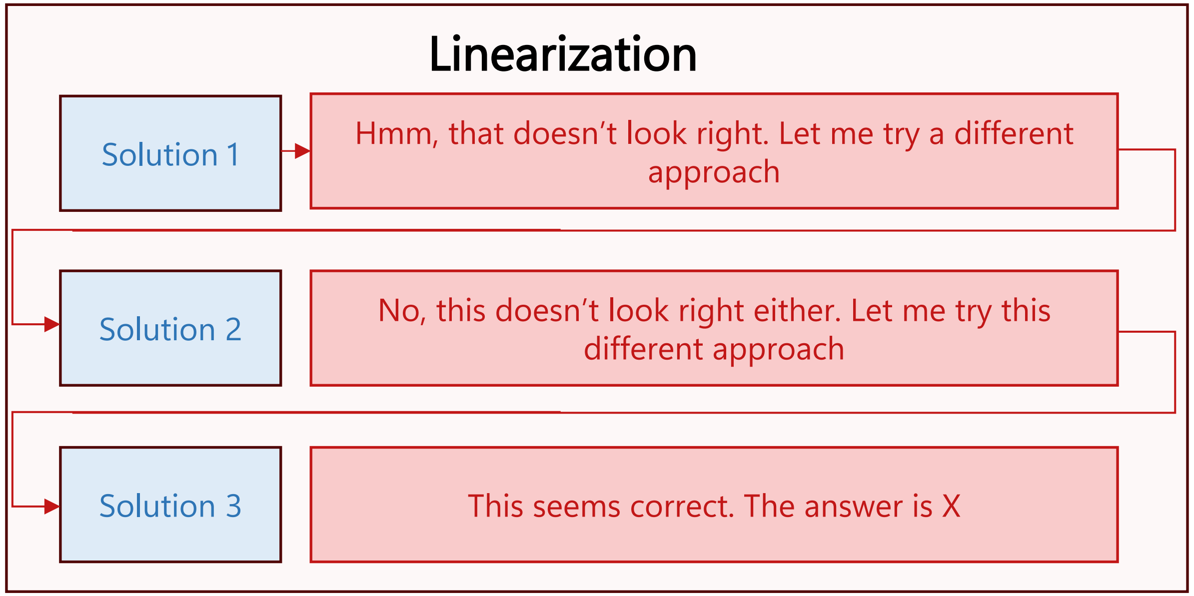
Model	AIME 2024		MATH-500	GPQA Diamond	LiveCodeBench
	pass@1	cons@64	pass@1	pass@1	pass@1
QwQ-32B-Preview	50.0	60.0	90.6	54.5	41.9
DeepSeek-R1-Zero-Qwen-32B	47.0	60.0	91.6	55.0	40.2
DeepSeek-R1-Distill-Qwen-32B	**72.6**	**83.3**	**94.3**	**62.1**	**57.2**

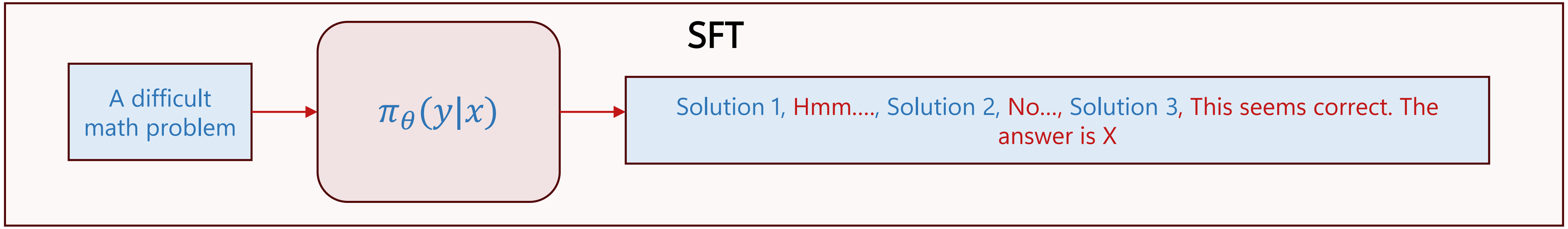
What if no teacher is available?

- Find challenging math problems
- Artificially create solutions with backtracking, verification, restarts, etc.
- SFT the model with these solutions to induce such behaviors in the model
- Then, do RL on hard problems – **GRPO + RLVR**

Teaching restarts







What about backtracking?

This was a mistake

A box is 2 cm high, 5 cm long and can hold 40gm of clay. A second box with twice the height … What is n?

Step 1: Determine the dimension of the 2nd box.
…
The dimensions are 4,9 and 5 ..

Step 2: Calculate the volume of the 2nd box
…
The volume is 180 cc

Step 3: Calculate the volume of the 1st box
…
The volume is 30 cc

Step 4: Calculate n, the amount of clay that 2nd box can hold
…
The answer is 480 gm

Step 3: Calculate the volume of the 1st box
…
The volume is 30 cc

Step 4: Calculate the ratio of the volumes
…
The ratio is 6:1

Step 5: Calculate n, the amount of clay that 2nd box can hold
…
The answer is 240 gm

Rewriting steps

A box is 2 cm high, 5 cm long and can hold 40gm of clay. A second box with twice the height … What is n?

→ Lets determine the dimension of the 2nd box. … The dimensions are 4,9 and 5 ..

→ Next, lets calculate the volume of the 2nd box … The volume is 180 cc

→ Next, lets calculate the volume of the 1st box … The volume is 30 cc

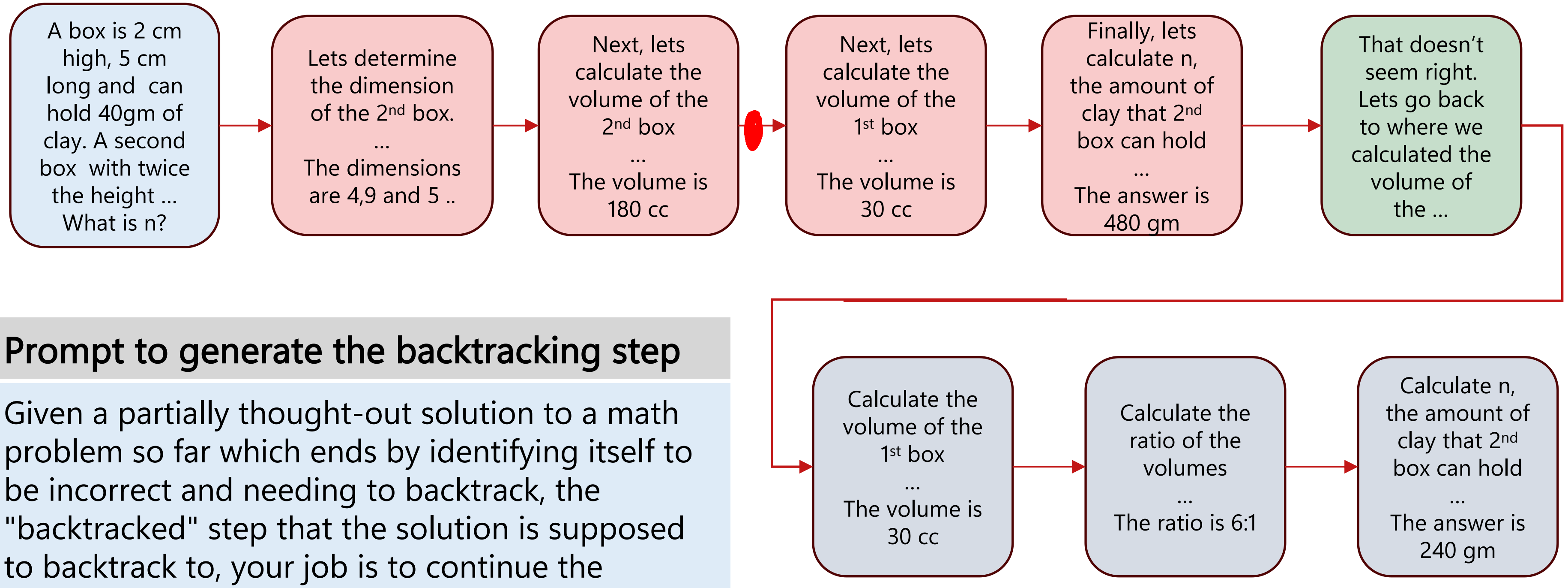
→ Finally, lets calculate n, the amount of clay that 2nd box can hold … The answer is 480 gm

Each step is rewritten by the LLM based on the previous steps

Prompt to rewrite each step

Given a partially thought-out solution to a math problem so far and the current step for solving the problem, your job is to rewrite the *current step* into a thought that smoothly continues the previous thoughts. This rewritten thought should only address the contents of the current step itself, nothing more or less.

Linearization of backtracking



Prompt to generate the backtracking step

Given a partially thought-out solution to a math problem so far which ends by identifying itself to be incorrect and needing to backtrack, the "backtracked" step that the solution is supposed to backtrack to, your job is to continue the existing solution thoughts by backtracking to the part of the solution that corresponds to the "backtracked step".

Generating the tree – Markov Chain Tree Search

- Each node S_t represents a partial solution till that point
- The edges, representing the actions a_t in MCTS - a step of the solution

A box is 2 cm high, 5 cm long and can hold 40gm of clay. A second box with twice the height … What is n?

→

Step 1: Determine the dimension of the 2nd box.
…
The dimensions are 4,9 and 5 ..

→ **Step 2: Calculate the volume of the 2nd box**
…
The volume is 180 cc

Step 1: Determine the dimension of the 2nd box.
…
The dimensions are 4,9 and 5 ..
Step 2: Calculate the volume of the 2nd box
…
The volume is 180 cc

Too clumsy !!

Generating the tree – Markov Chain Tree Search

- Each node represents a partial solution till that point
- The edges correspond to an action - a step

A box is 2 cm high, 5 cm long and can hold 40gm of clay. A second box with twice the height … What is n? → **Step 1: Determine the dimension of the 2nd box.** … The dimensions are 4,9 and 5 .. → **Step 2: Calculate the volume of the 2nd box** … The volume is 180 cc

Easy to view

S_0 $\quad$ s_1

Partial solution so far

- Which node should I expand next?
- Which node is most likely to reach the correct solution?
- We need a way to evaluate the nodes.

$S_{t+1} = (S_t, a_t)$

Reward at a node

- **Given a node S_t representing a partial solution**
  - Sample multiple completions $\Pi_{LM,j}(S_t)$ of the partial solution.
  - Compare the answer against a ground-truth G using a verifier Ver
  - The reward of the node

$$R(S_t) = \frac{1}{N}\sum_{j=1}^{N} \mathrm{Ver}(\Pi_{LM,j}(S_t)\,,\ G)$$

$R_2 = \frac{2}{4} = 0.5$

A box is 2 cm high, 5 cm long and can hold 40gm of clay. A second box with twice the height ... What is n?

→ **Step 1: Determine the dimension of the 2nd box.** ... The dimensions are 4,9 and 5 ..

→ **Step 2: Calculate the volume of the 2nd box** ... The volume is 200 cc

$R_1 = 0.9,\ N_1 = 1$

$R_2 = 0,\ N_2 = 0$

Comp1 ✓

Comp2 ✓

Comp3 X

Comp4 X

Q-value of an action

- Given the state-action pair (S_{t-1}, a)
- The next state after taking a at S_{t-1} is S_t
- A weighted average
  - Reward of the next state
  - Q-value of taking each action at the next state
  - State-action pairs that have been traversed more contribute more

$$Q(S_{t-1}, a) = \frac{\sum_{a'} Q(S_t, a')N(S_t, a') + R(S_t)}{\sum_{a'} N(S_t, a') + 1}$$

Q-value for stepwise generation from LLMs

- A weighted average
  - Its own reward
  - Value of its children

$$V(S_t) = \frac{\sum_{S \in \mathcal{N}(S_t)} V(S) N(S) + R(S_t)}{\sum_{S \in \mathcal{N}(S_t)} N(S) + 1}$$

children of S_t

→ reward of node itself

no. of times the states has been selected

A box is 2 cm high, 5 cm long and can hold 40gm of clay. A second box with twice the height … What is n?

Node 1

Step 1: Determine the dimension of the 2nd box.
…
The dimensions are 4,9 and 5 ..

$R_1 = 0.9,\ N_1 = 1$

$V_3 = \ ??$

Step 2: Calculate the volume of the 2nd box
…
The volume is 200 cc

$R_2 = 0,\ V_2 = 0,\ N_2 = 1$

How many times this node has been seen

Step 2: Calculate the volume of the 2nd box
…
The volume is 200 cc

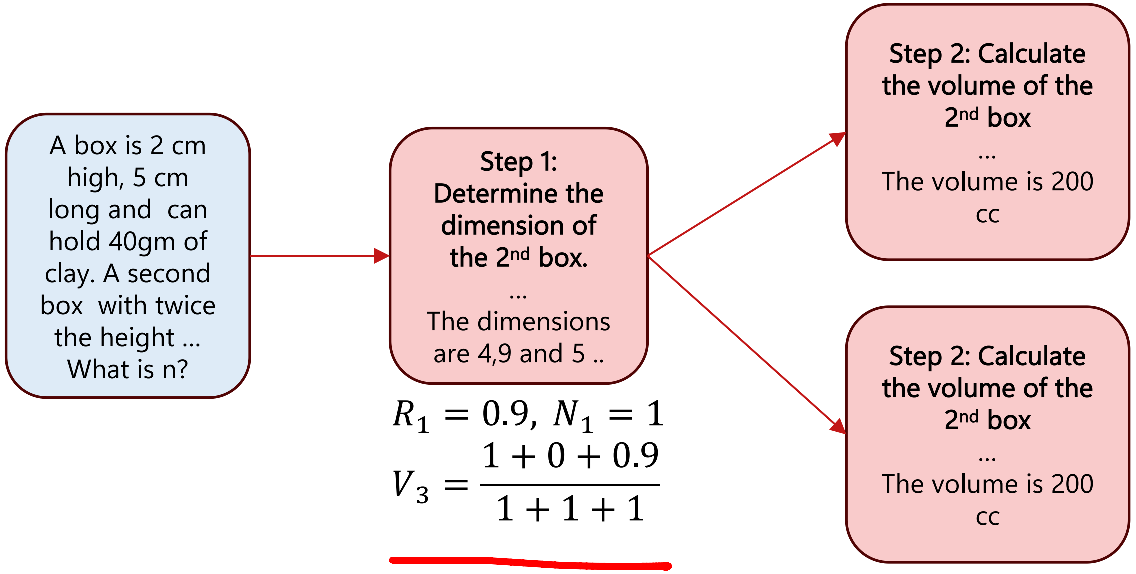
$R_3 = 1,\ V_3 = 1,\ N_3 = 1$

$$\frac{0.9 + 0 \times 1 + 1 \times 1}{1+1+1} = \frac{1.9}{3}$$

Value at a node

- A weighted average
  - Its own reward
  - Value of its children

$$V(S_t) = \frac{\sum_{S \in \mathcal{N}(S_t)} V(S)N(S) + R(S_t)}{\sum_{S \in \mathcal{N}(S_t)} N(S) + 1}$$



$R_1 = 0.9,\ N_1 = 1$

$$V_3 = \frac{1 + 0 + 0.9}{1 + 1 + 1}$$

$R_2 = 0,\ V_2 = 0,\ N_2 = 1$

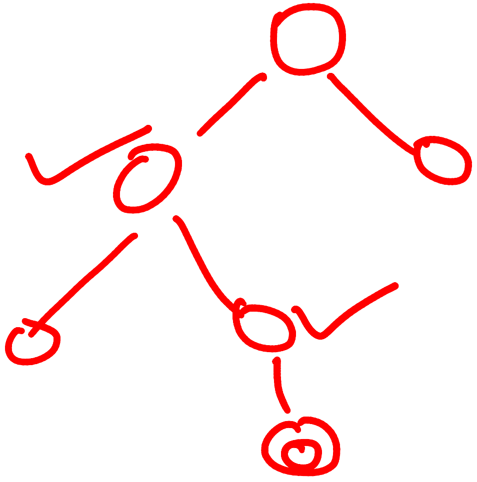
$R_3 = 1,\ V_3 = 1,\ N_3 = 1$

The 4 steps of MCTS

- Selection
  - Which node should I select for expansion?
- Expansion
  - How do I sample the next step? → Sample the next step from a selected node?
- Evaluation
  - How do I evaluate the node? → Compute the reward.
- Backpropagation
  - How do I propagate the reward upwards?

Computing the value function starting from leaves

Selection



- Exploitation only
  - Select the node that has the highest value function $V(S_{t+1})$
- Exploration only
  - Select the node that has not been well-explored

$$\left\{ \frac{\sqrt{N(S_t)}}{1+N(S_{t+1})} \right\} \qquad \frac{1}{N(S_{t+1})}$$

- Exploration vs Exploitation

$$S_{t+1}^{*} = argmax_{S_{t+1}=S_t \to a_i}\left[\mathrm{V}(S_{t+1}) + c_{\mathrm{puct}} \cdot \Pi_{LM}(a_i|S_t) \cdot \frac{\sqrt{N(S_t)}}{1+N(S_{t+1})} \right]$$

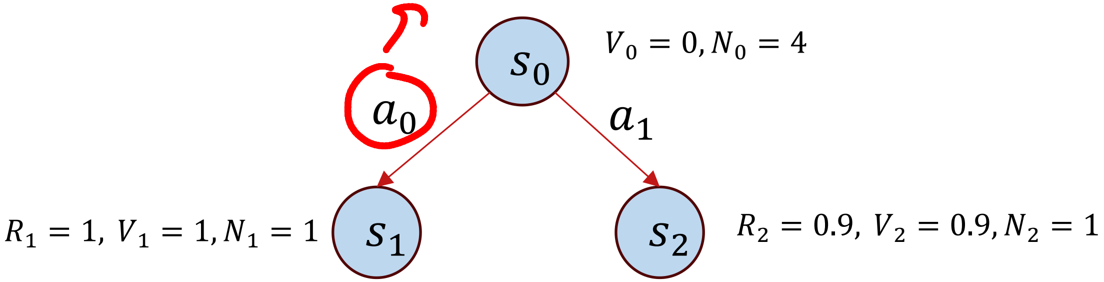
Predictor+Upper Confidence bounds applied to Trees (PUCT, Silver et al. (2016))

Expansion, Evaluation and Backpropagation

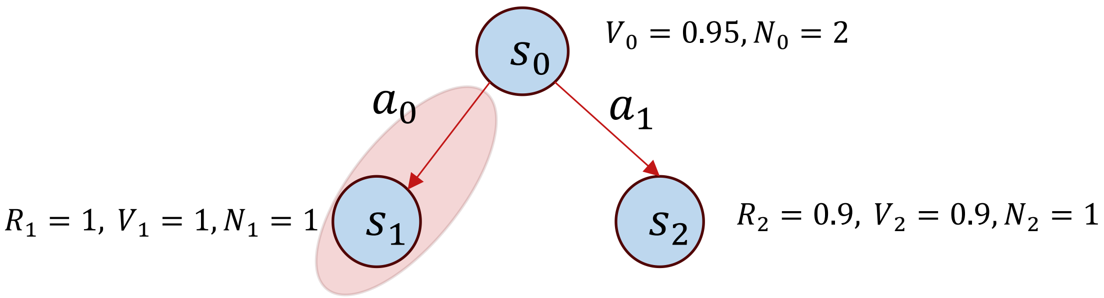
- The selected node corresponds to a partial solution
- Sample k next steps given the partial solution → Expansion
- Compute the reward of each partial solution with the next step added. → Evaluation
- Update the value at each node from the reward and value of its children → Backtracking

An illustration

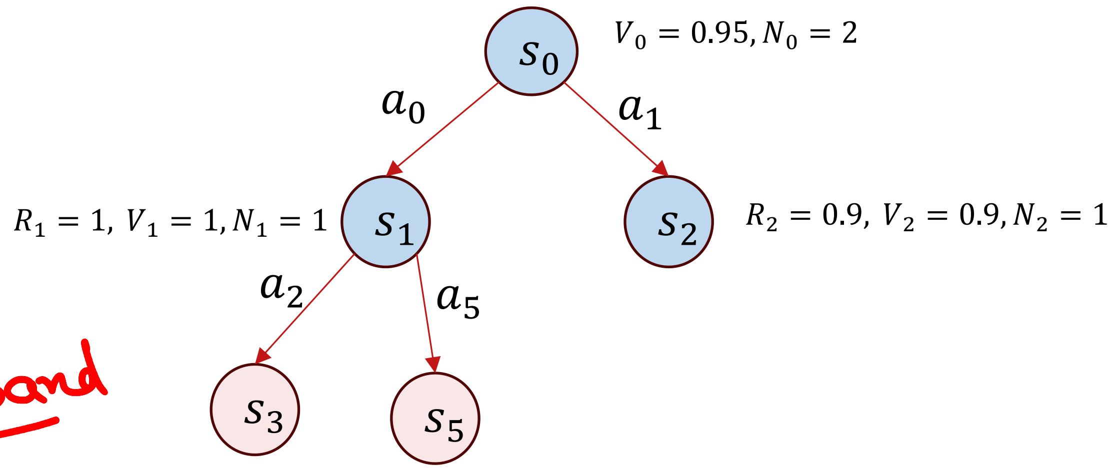
one step of the solution



Selection

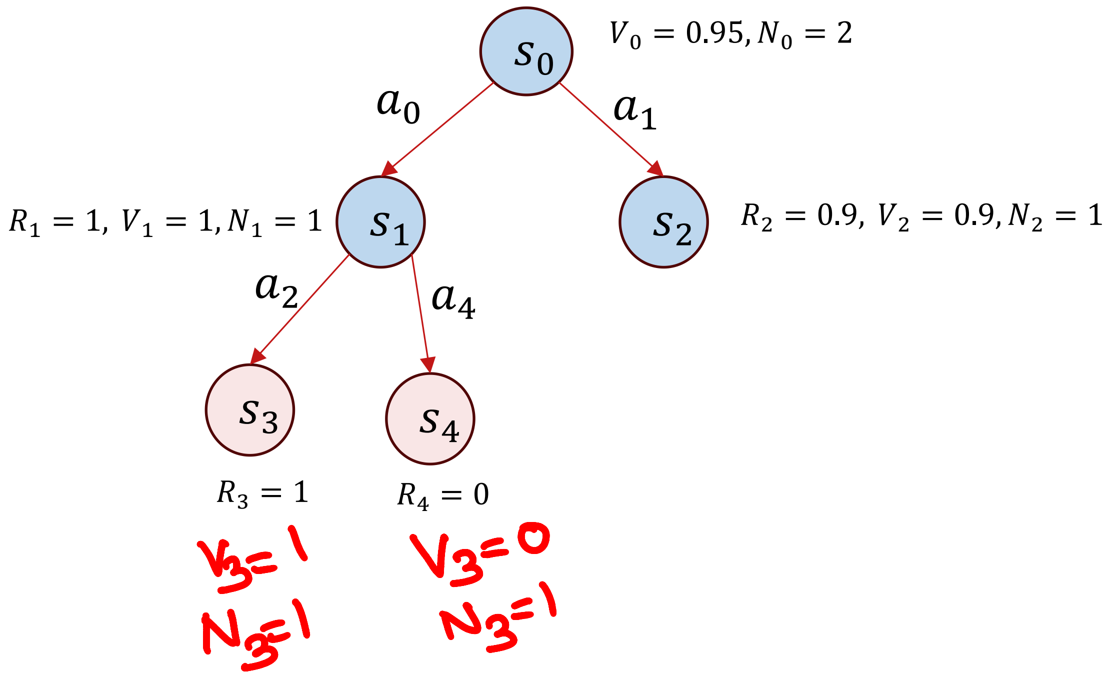


Expansion



Expand

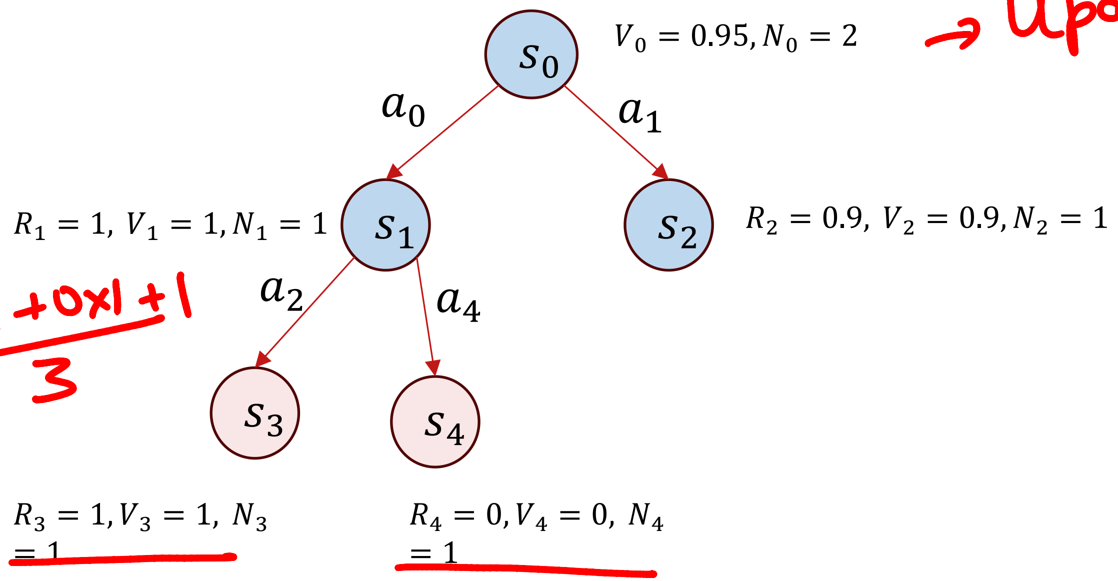
Evaluation



s_0
$V_0 = 0.95, N_0 = 2$
a_0
a_1
$R_1 = 1,\ V_1 = 1, N_1 = 1$
s_1
s_2
$R_2 = 0.9,\ V_2 = 0.9, N_2 = 1$
a_2
a_4
s_3
s_4
$R_3 = 1$
$R_4 = 0$
$V_3 = 1$
$N_3 = 1$
$V_3 = 0$
$N_3 = 1$

Backpropagation

→ Upate V_0 based on updated V_1 and V_2



$V_1 = \frac{1\times1 + 0\times1 + 1}{3}$

$\frac{2}{3}$

Training

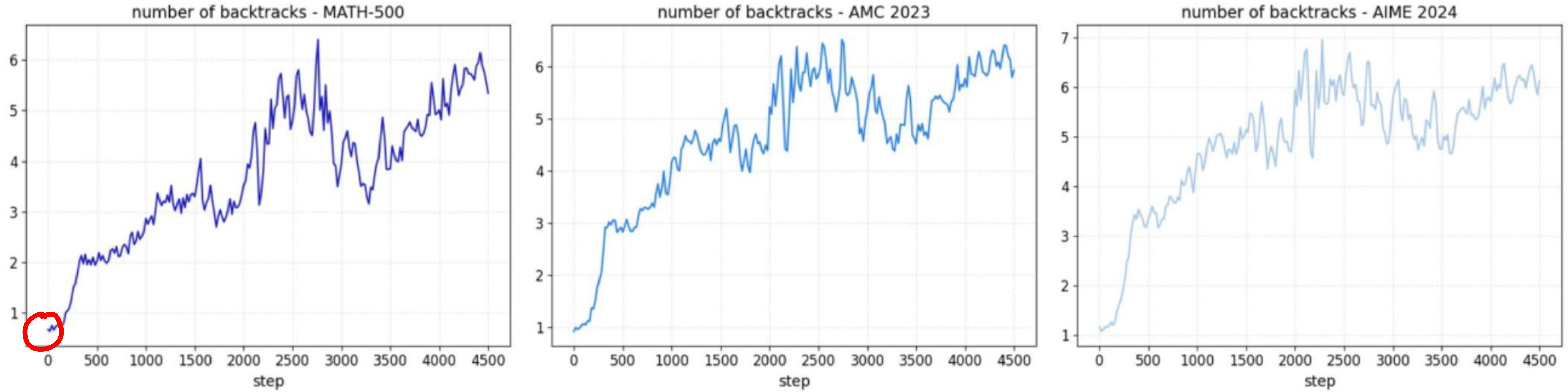
- The SFT model is trained on linearized traces
- This teaches the model how to reason

SFT

A difficult math problem → $\pi_\theta(y|x)$ → Solution 1, Hmm...., Solution 2, No..., Solution 3, This seems correct. The answer is X

- This is followed by reinforcement learning – GRPO + RLVR

Does backtracking increase with RL training?



ASTRO: Teaching Language Models to Reason by Reflecting and Backtracking In-Context

Direct vs backtracked solutions

Non-linearization - Pick the correct path

Checkpoint	MATH-500	AMC 2023		AIME 2024	
	pass@1	pass@1	maj@8	pass@1	maj@8
Llama-3.1-70B-Direct-SFT	**65.8**	**45.2**	**58.0**	**16.7**	**23.3**
Llama-3.1-70B-ASTRO-SFT	69.6	51.9	63.0	13.3	16.7
Llama-3.1-70B-Direct-RL	**79.8**	**60.5**	**67.8**	**27.1**	**30.3**
Llama-3.1-70B-ASTRO-RL	**81.8**	**64.4**	**68.8**	**30.0**	**32.0**

ASTRO: Teaching Language Models to Reason by Reflecting and Backtracking In-Context

References & Further Reading

- **Chain-of-Thought Prompting Elicits Reasoning in LLMs** (2022) — Prompt-only emergence of multi-step solutions.
- **STaR: Bootstrapping Reasoning With Rationales** (2022) — Self-generate explanations → filter → SFT.
- **Thinking LLMs: Thought Preference Optimization (TPO)** (2024) — Optimize preferences over thoughts, not just finals.
- **DeepSeek-R1: Incentivizing Reasoning via RL** (2025) — GRPO + verifiers; emergence of planning/backtracking.
- **Cognitive Behaviors that Enable Self-Improving Reasoners** (2025) — behavior presence predicts RL self-improvement
- **ASTRO: Teaching Language Models to Reason by Reflecting and Backtracking In-Context** (2025) – incorporating reasoning patterns by MCTS