

Open-R1

A Fully Open Reproduction of DeepSeek-R1

Guilherme Penedo

Hi, I'm Guilherme!



Falcon LLM team @ LightOn/TII: RefinedWeb dataset

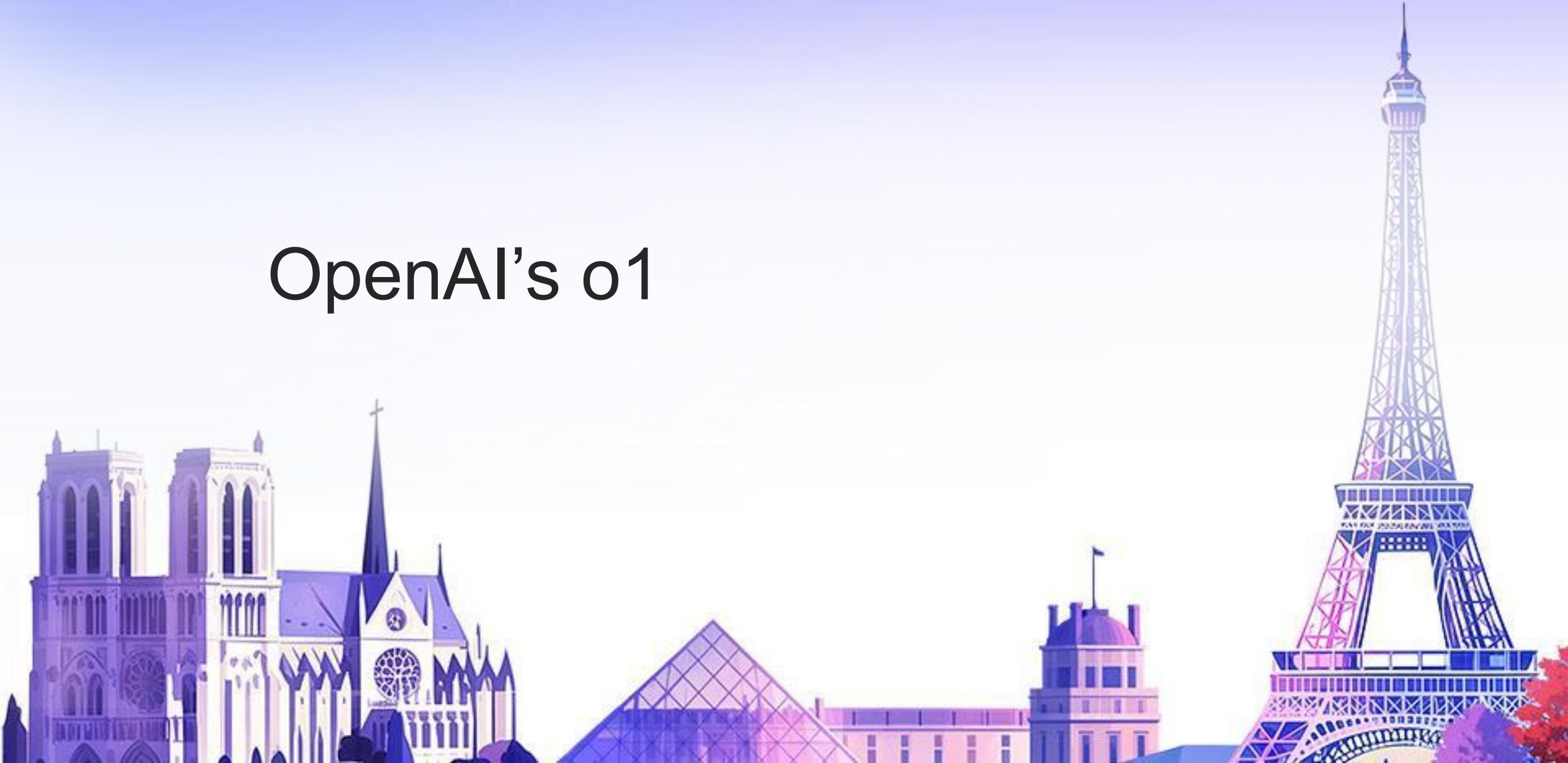
Pre-training @Hugging Face: FineWeb, FineWeb2 datasets

Open-R1, mostly on data & code reasoning

 😊 @guipenedo  @gui_penedo

1. OpenAI's o1
2. DeepSeek-R1
3. Open-R1

OpenAI's o1



o1: the 1st reasoning model



GOSIM

Noam Brown @polynoamial ⚡ ...

Today, I'm excited to share with you all the *fruit* of our effort at @OpenAI to create AI models capable of truly general reasoning: OpenAI's new o1 model series! (aka 🍓) Let me explain 🧠 1/

Competition Math (AIME 2024)

Model	accuracy
gpt4o	13.4
o1 preview	56.7
o1	83.3

Competition Code (CodeForces)

Model	percentile
gpt4o	11.0
o1 preview	62.0
o1	89.0

PhD-Level Science Questions (GPQA Diamond)

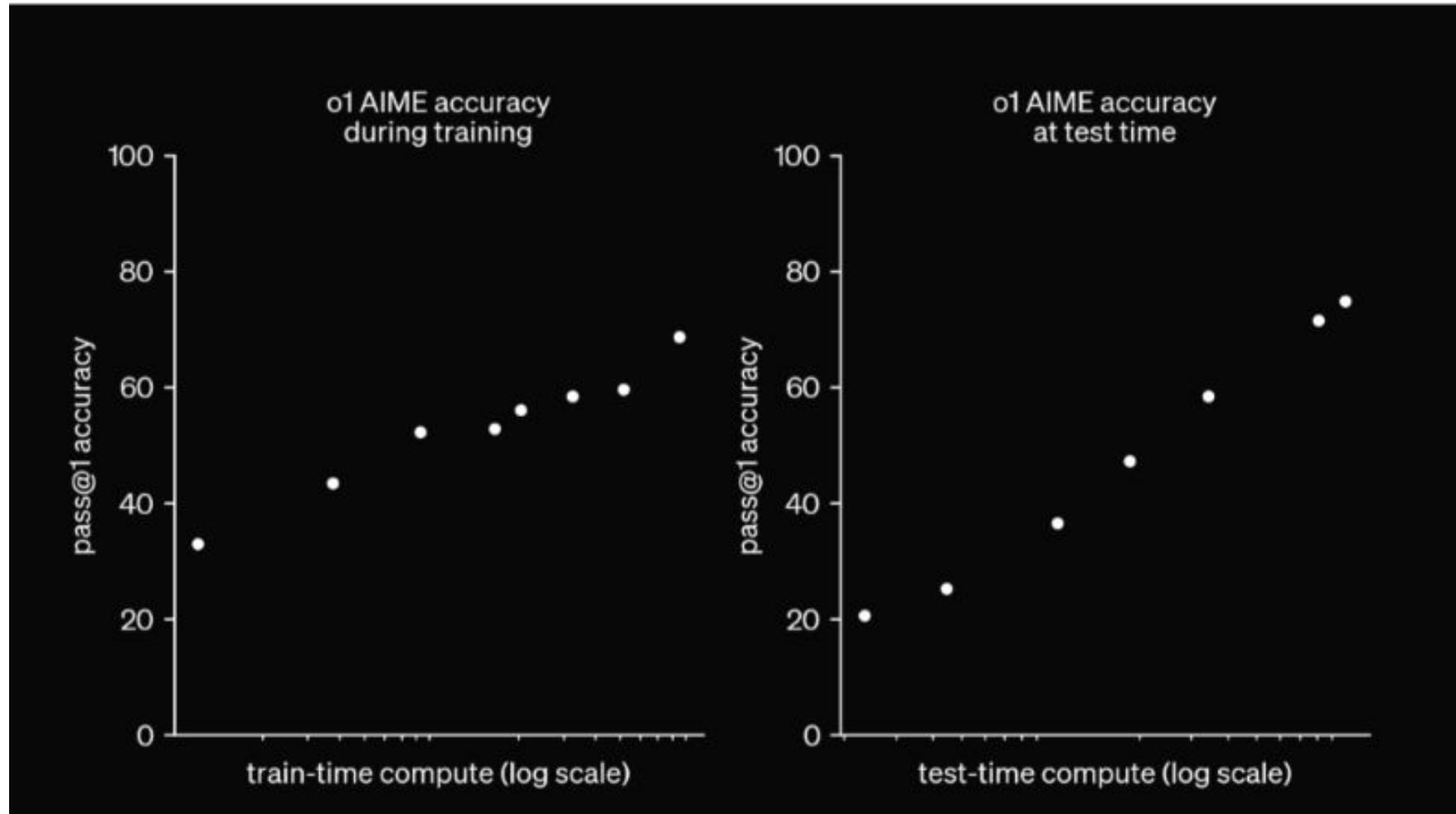
Model	accuracy
gpt4o	56.1
o1 preview	78.3
o1	78.0
expert human	69.7

7:17 PM · Sep 12, 2024 · 2.5M Views

o1: scaling *test-time* compute



GOSIM



[OpenAI, 2024]

o1: what we knew



GOSIM

“Think step by step”

model learns “by itself”
search

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.

- OpenAI, 2024



o1: how did it actually work?



GOSIM

The thumbnail shows a black background with white text. At the top left is a white circular logo. To its right is a text box containing a paragraph about o1 learning strategies. Below the text box is a timer showing 47:56. The main title 'Speculations on Test-Time Scaling' is at the bottom left, followed by '(o1)'. To the right of the title is a small image of a person's face in a circular frame. Below the person's image is the name 'Sasha Rush' and some smaller text about her.

o1 learns to hone its chain of thought and refine the strategies it uses. It learns to recognize and correct its mistakes. It learns to break down tricky steps into simpler ones. It learns to try a different approach when the current one isn't working.

47:56

Speculations on Test-Time Scaling : (o1)

Sasha Rush
@srush_nlp · 11.3K subscribers
I give technical talks and courses
rush-nlp.com and 1 more link

Subscribe

The Suspects

- Guess + Check
- Process Rewards
- Search / AlphaZero
- Learning to Correct



DeepSeek-R1



DeepSeek-R1: Open-weights reasoning model



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

DeepSeek-AI

research@deepseek.com

Abstract

We introduce our first-generation reasoning models, DeepSeek-R1-Zero and DeepSeek-R1. DeepSeek-R1-Zero, a model trained via large-scale reinforcement learning (RL) without supervised fine-tuning (SFT) as a preliminary step, demonstrates remarkable reasoning capabilities. Through RL, DeepSeek-R1-Zero naturally emerges with numerous powerful and intriguing reasoning behaviors. However, it encounters challenges such as poor readability, and language mixing. To address these issues and further enhance reasoning performance, we introduce DeepSeek-R1, which incorporates multi-stage training and cold-start data before RL. DeepSeek-R1 achieves performance comparable to OpenAI-o1-1217 on reasoning tasks. To support the research community, we open-source DeepSeek-R1-Zero, DeepSeek-R1, and six dense models (1.5B, 7B, 8B, 14B, 32B, 70B) distilled from DeepSeek-R1 based on Qwen and Llama.

[cs.CL] 22 Jan 2025

GOSIM AI Paris 2025

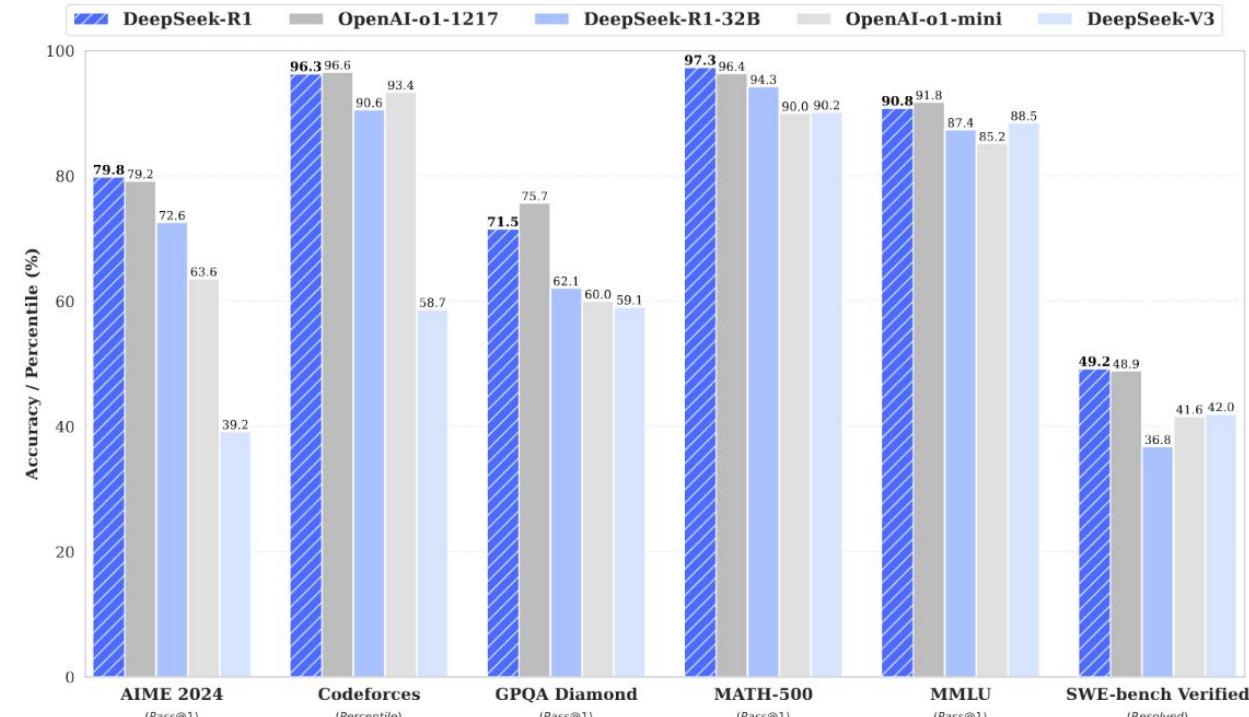


Figure 1 | Benchmark performance of DeepSeek-R1.



DeepSeek-R1: similar to o1



 **Mark Chen** 
@markchen90

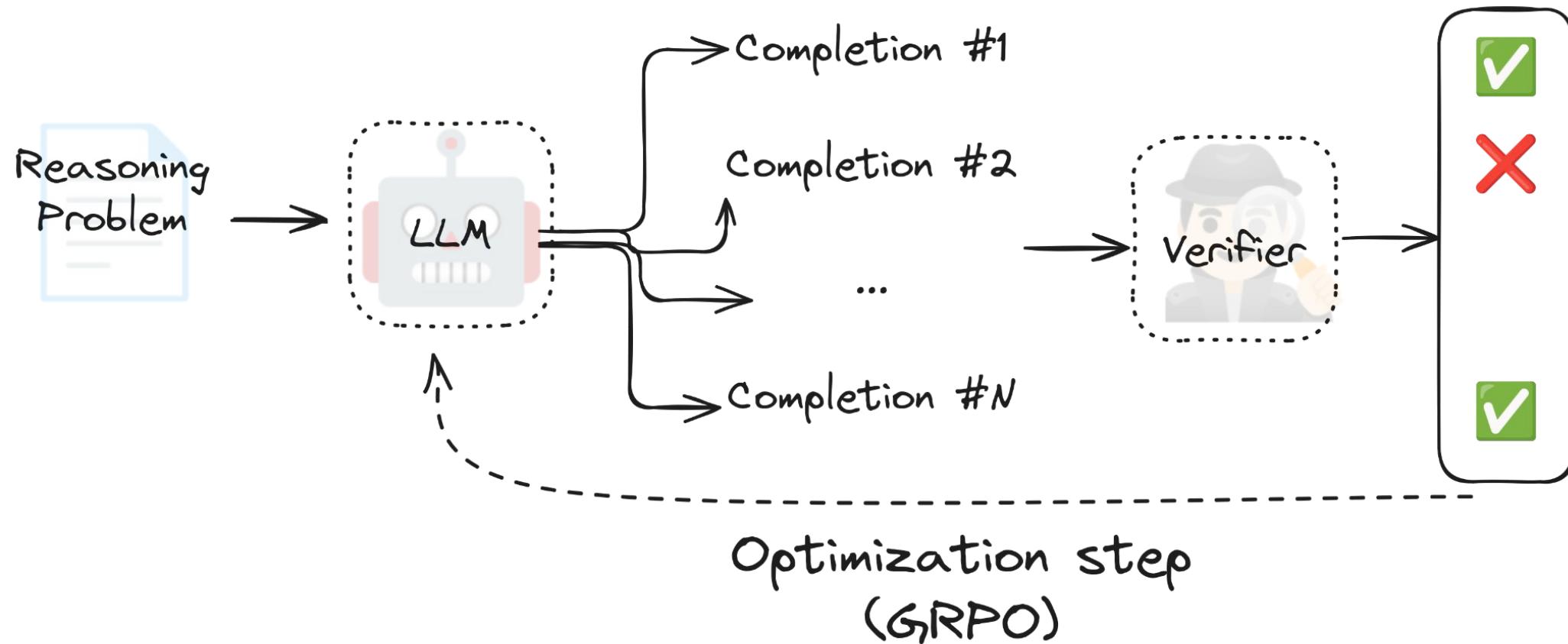
∅ ...

Congrats to DeepSeek on producing an o1-level reasoning model! Their research paper demonstrates that they've independently found some of the core ideas that we did on our way to o1.

7:11 PM · Jan 28, 2025 · **8.2M** Views



DeepSeek-R1: Simpler than expected!



DeepSeek-R1: Increased “thinking time”

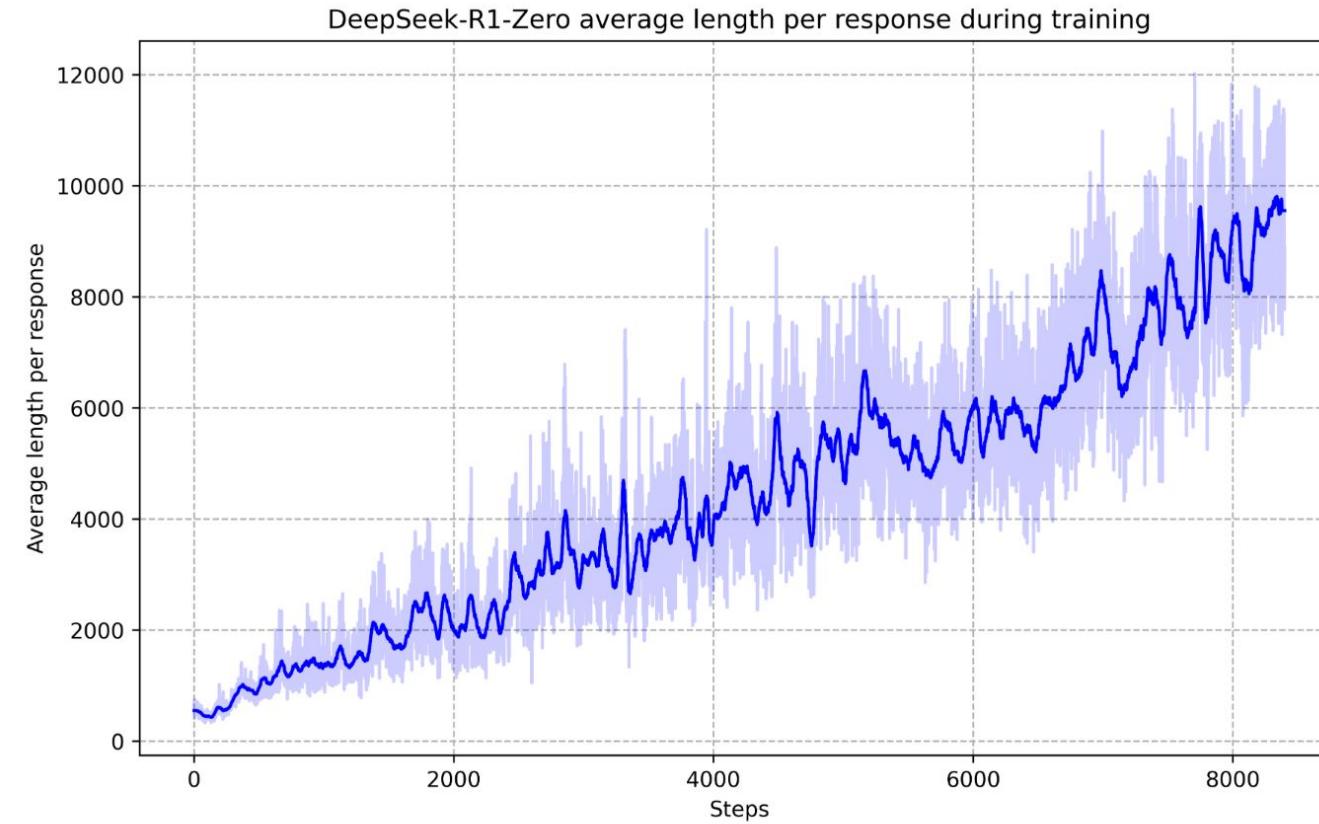


Figure 3 | The average response length of DeepSeek-R1-Zero on the training set during the RL process. DeepSeek-R1-Zero naturally learns to solve reasoning tasks with more thinking time.



DeepSeek-R1: “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 \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.



DeepSeek-R1: Distillation



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

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

We need strong base models!
Pre-training isn't dead 😊



Open-R1



Open-R1: why “open”?



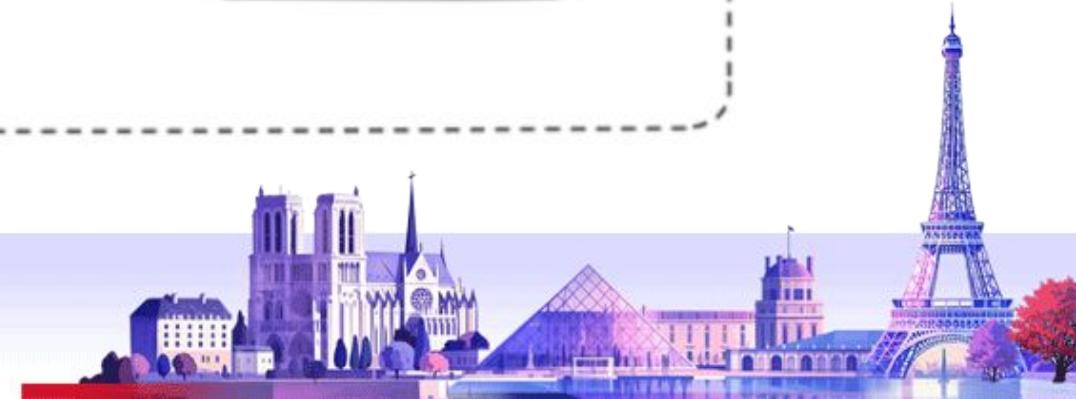
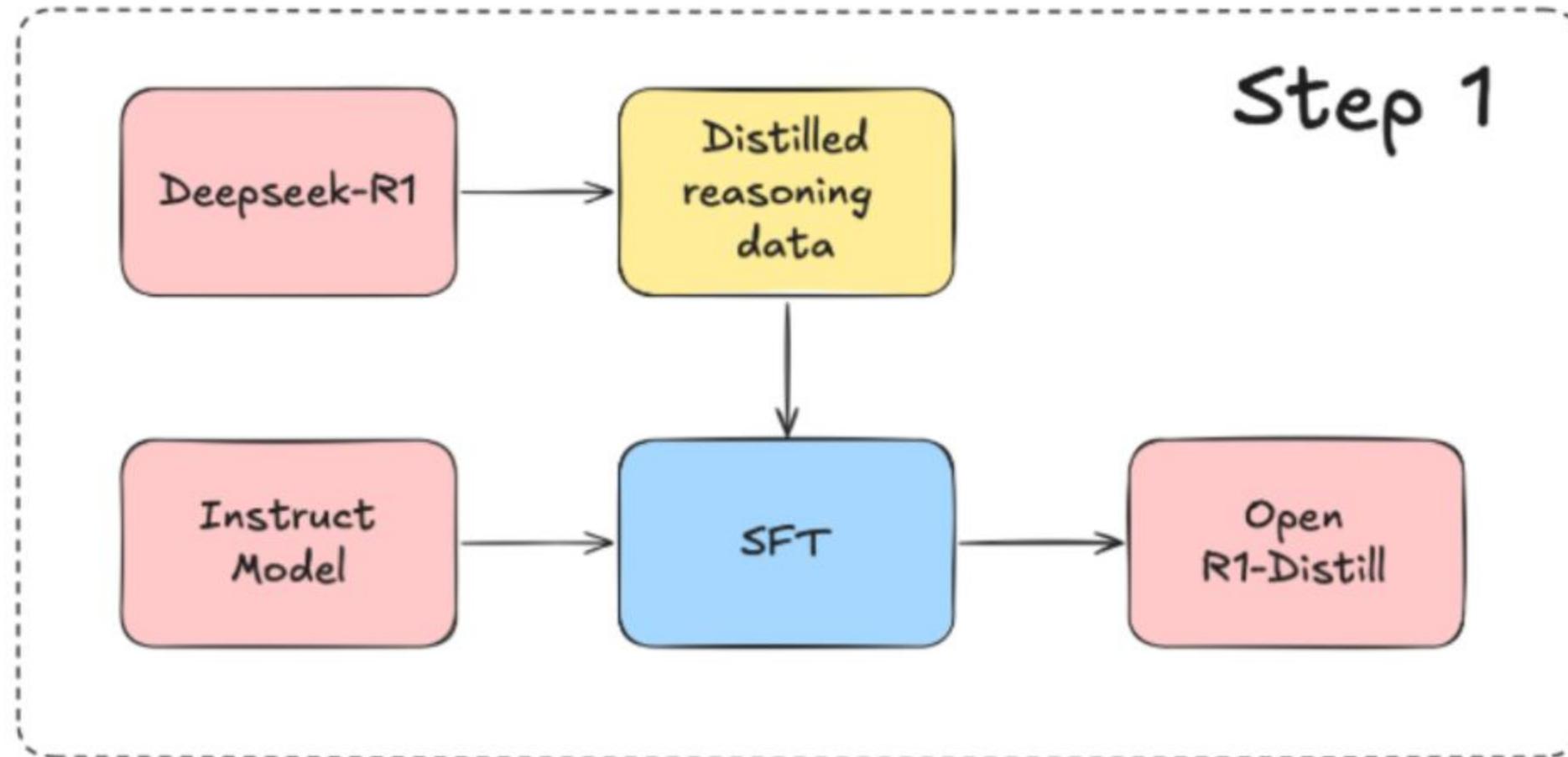
Isn't DeepSeek-R1 open?

Yes, but...

- Training code?
- Scaling and trade-offs?
- **Verifiers???**
- **Data???????????????**



Open-R1: Plan of attack #1 – Distilled models





🧠 Reasoning datasets ↗

Datasets with reasoning traces for math and code released by the community ↗

updated 17 days ago

bespokeLabs/Bespoke-Stratos-17k
Viewer • Updated Jan 31 • 16.7k • 22.3k • 307

Note 17k reasoning traces for coding and math distilled from DeepSeek-R1

open-thoughts/OpenThoughts-114k
Viewer • Updated 26 days ago • 228k • 28.6k • 693

Note 114k reasoning traces covering math, science, code, and puzzles distilled from DeepSeek-R1

open-r1/OpenThoughts-114k-math
Viewer • Updated Jan 30 • 89.1k • 1.02k • 81

Note Filtered version of OpenThoughts-114k based on correct answers

PrimeIntellect/NuminaMath-QwQ-CoT-5M
Viewer • Updated Jan 22 • 5.14M • 2.51k • 47

Note 5M reasoning traces for math problems distilled from QwQ-32B

ServiceNow-AI/R1-Distill-SFT
Viewer • Updated Feb 8 • 1.85M • 1.93k • 295

Note 1.7M reasoning traces for math problems distilled from DeepSeek-R1-Distill-Qwen-32B

cognitivecomputations/dolphin-r1
Viewer • Updated Jan 30 • 814k • 1.15k • 277

Note A mix of reasoning traces from DeepSeek-R1, Gemini Flash and Dolphin Chat

GAIR/LIMO
Viewer • Updated Feb 10 • 817 • 3.75k • 154

simplescaling/s1K
Viewer • Updated Feb 11 • 1k • 2.3k • 213

glaiveai/reasoning-v1-20m

Viewer • Updated Mar 19 • 22.2M • 9.35k • 205

BytedTsinghua-SIA/DAPO-Math-17k

Viewer • Updated 15 days ago • 1.79M • 4.22k • 67

nvidia/Llama-Nemotron-Post-Training-Dataset

Viewer • Updated 6 days ago • 3.91M • 9.01k • 433

Intelligent-Internet/II-Thought-RL-v0

Viewer • Updated Mar 28 • 342k • 2.28k • 49

SynthLabsAI/Big-Math-RL-Verified

Viewer • Updated Mar 25 • 251k • 5.75k • 175

virtuoussy/Multi-subject-RLVR

Viewer • Updated 17 days ago • 579k • 1.55k • 55

zwhe99/DeepMath-103K

Viewer • Updated 15 days ago • 103k • 18.8k • 162



Math: Reasoning problems



IMO 2024
65th International Mathematical Olympiad

English (eng), day 1

Tuesday, 16. July 2024

Problem 1. Determine all real numbers α such that, for every positive integer n , the integer

$$\lfloor \alpha \rfloor + \lfloor 2\alpha \rfloor + \cdots + \lfloor n\alpha \rfloor$$

is a multiple of n . (Note that $\lfloor z \rfloor$ denotes the greatest integer less than or equal to z . For example, $\lfloor -\pi \rfloor = -4$ and $\lfloor 2 \rfloor = \lfloor 2.9 \rfloor = 2$.)

Problem 2. Determine all pairs (a, b) of positive integers for which there exist positive integers g and N such that

$$\gcd(a^n + b, b^n + a) = g$$

holds for all integers $n \geq N$. (Note that $\gcd(x, y)$ denotes the greatest common divisor of integers x and y .)

Problem 3. Let a_1, a_2, a_3, \dots be an infinite sequence of positive integers, and let N be a positive integer. Suppose that, for each $n > N$, a_n is equal to the number of times a_{n-1} appears in the list a_1, a_2, \dots, a_{n-1} .

Prove that at least one of the sequences a_1, a_3, a_5, \dots and a_2, a_4, a_6, \dots is eventually periodic.

(An infinite sequence b_1, b_2, b_3, \dots is *eventually periodic* if there exist positive integers p and M such that $b_{m+p} = b_m$ for all $m \geq M$.)



GOSIM

Solution 1

To solve the problem, we need to find all real numbers α such that, for every positive integer n , the integer

$$S_n(\alpha) = \lfloor \alpha \rfloor + \lfloor 2\alpha \rfloor + \cdots + \lfloor n\alpha \rfloor$$

is divisible by n , i.e., $S_n(\alpha) \equiv 0 \pmod{n}$.

Step 1: Break Down α into Integer and Fractional Parts

Let $\alpha = m + f$, where $m = \lfloor \alpha \rfloor \in \mathbb{Z}$ and $f = \{\alpha\} \in [0, 1)$ is the fractional part of α .

Step 2: Express the Sum in Terms of m and f

Using this, we have:

$$\lfloor k\alpha \rfloor = \lfloor k(m + f) \rfloor = km + \lfloor kf \rfloor.$$

So, the sum becomes:

$$S_n(\alpha) = m \sum_{k=1}^n k + \sum_{k=1}^n \lfloor kf \rfloor = m \frac{n(n+1)}{2} + \sum_{k=1}^n \lfloor kf \rfloor.$$

Step 3: Modulo n Simplification

We are interested in $S_n(\alpha) \pmod{n}$:

$$S_n(\alpha) \equiv \left(m \frac{n(n+1)}{2} + \sum_{k=1}^n \lfloor kf \rfloor \right) \pmod{n}.$$

Since $m \frac{n(n+1)}{2}$ is divisible by n , the expression simplifies to:

$$S_n(\alpha) \equiv \sum_{k=1}^n \lfloor kf \rfloor \pmod{n}.$$

Step 4: Analyze the Fractional Part f

Our goal is to find all $f \in [0, 1)$ such that:

$$\sum_{k=1}^n \lfloor kf \rfloor \equiv 0 \pmod{n} \quad \text{for all } n \in \mathbb{N}.$$

Step 5: Test $f = 0$

If $f = 0$, then $\lfloor kf \rfloor = 0$ for all k , so:

$$\sum_{k=1}^n \lfloor kf \rfloor = 0,$$



Verifying Math



S Datasets: open-r1/OpenR1-Math-220k like 567 Following Open R1 1.49k

Modalities: Text Formats: parquet Languages: English Size: 100K - 1M Libraries: Data

License: apache-2.0

Dataset card Data Studio Files Community 13 Settings

Dataset Viewer Auto-converted to Parquet API Embed Data Studio

Subset (3) default · 93.7k rows Split (1) train · 93.7k rows

Search this dataset

problem string · lengths solution string

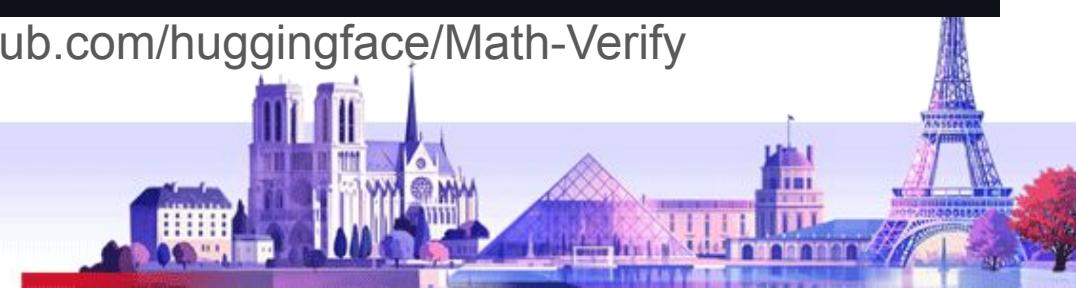
```
from math_verify import parse, verify

# Parse the gold and answer
# If you know that gold will only contain integers
# parse(gold, extraction_config=[LatexEquationParser()])
gold = parse("${1,3} \cup {2,4}")
answer = parse("${1,2,3,4}")

# Order here is important!
verify(gold, answer)
# >>> True
```

- Complete set theory support (Intervals, FiniteSets, set operations)
- Unicode symbol substitution support (e.g. $\beta \rightarrow \text{beta}$)
- Applies Latex fixes for common malformations (e.g. $\frac{1}{3} \rightarrow 1/3$)
- Equation and inequality parsing, with symbol assignment resolution (e.g. $x = 1 \rightarrow 1$)
- Percentage best effort conversion (e.g. $10\% \rightarrow 0.1$)
- Units in text handling (e.g. $10 \text{ cm} \rightarrow 10$)
- Exact representation of the input expressions (e.g. $0.333 \rightarrow \text{Float}(333, 3)$)

<https://github.com/huggingface/Math-Verify>



Math: Approaching DeepSeek-Distill



Model	MATH-500	AIME24	AIME25
DeepSeek-Distill-Qwen-7B	91.6	43.3	40
OpenR1-Qwen-7B	90.6	36.7	40
OpenThinker-7B	89.6	30.0	33.3

<https://huggingface.co/open-r1/OpenR1-Qwen-7B>



D. Array and GCD

time limit per test: 2 seconds

memory limit per test: 512 megabytes

You are given an integer array a of size n .

You can perform the following operations any number of times (possibly, zero):

- pay one coin and increase any element of the array by 1 (you must have at least 1 coin to perform this operation);
- gain one coin and decrease any element of the array by 1.

Let's say that an array is *ideal* if both of the following conditions hold:

- each element of the array is at least 2;
- for each pair of indices i and j ($1 \leq i, j \leq n; i \neq j$) the greatest common divisor (GCD) of a_i and a_j is equal to 1. If the array has less than 2 elements, this condition is automatically satisfied.

Let's say that an array is *beautiful* if it can be transformed into an ideal array using the aforementioned operations, provided that you initially have no coins. If the array is already ideal, then it is also beautiful.

The given array is not necessarily beautiful or ideal. You can remove any elements from it (including removing the entire array or not removing anything at all). Your task is to calculate the minimum number of elements you have to remove (possibly, zero) from the array a to make it **beautiful**.

Input

The first line contains a single integer t ($1 \leq t \leq 10^4$) — the number of test cases.

The first line of each test case contains a single integer n ($1 \leq n \leq 4 \cdot 10^5$).

The second line contains n integers a_1, a_2, \dots, a_n ($2 \leq a_i \leq 10^9$).

Additional constraint on the input: the sum of n over all test cases doesn't exceed $4 \cdot 10^5$.

Output

For each test case, print a single integer — the minimum number of elements you have to remove (possibly, zero) from the array a to make it **beautiful**.

Example

input

```
5
3
5 5 5
4
2 3 2 4
1
3
3
2 100 2
5
2 4 2 11 2
```

[Copy](#)

output

```
0
2
0
0
0
1
```

[Copy](#)

Note

In the first example, you don't need to delete any elements, because the array is already beautiful. It can be transformed into an ideal array as follows: $[5, 5, 5] \rightarrow [4, 5, 5] \rightarrow [4, 4, 5] \rightarrow [4, 3, 5]$ (you end up with 3 coins).

In the second example, you need to remove 2 elements so that the array becomes beautiful. If you leave the elements $[2, 3]$ and delete the other elements, then the given array is already ideal (and therefore, beautiful).

In the third example, you don't need to delete any elements because the array is already ideal (and thus, beautiful).

In the fourth example, the array is beautiful. It can be transformed into an ideal array as follows:
 $[2, 100, 2] \rightarrow [2, 99, 2] \rightarrow [2, 99, 3] \rightarrow [2, 98, 3] \rightarrow [2, 97, 3]$ (you end up with 2 coins).

Contest status

#	When	Who	Problem	Lang	Verdict	Time	Memory
318205236	May/03/2025 12:07 UTC+2	23CS02002	D - Array and GCD	C++23 (GCC 14-64, msys2)	Running on test 15	0 ms	0 KB
318205233	May/03/2025 12:07 UTC+2	SajidZakaria <small>00:57</small>	D - Array and GCD	C++20 (GCC 13-64)	Wrong answer on test 4	453 ms	134100 KB
318205003	May/03/2025 12:04 UTC+2	23CS02002	D - Array and GCD	C++23 (GCC 14-64, msys2)	Runtime error on test 4 <small>⚠</small>	406 ms	74800 KB
318205011	May/03/2025 12:04 UTC+2	Kirtyapriya	D - Array and GCD	C++23 (GCC 14-64, msys2)	Wrong answer on test 5	93 ms	7800 KB
318204841	May/03/2025 12:03 UTC+2	23CS02002	D - Array and GCD	C++23 (GCC 14-64, msys2)	Runtime error on test 4 <small>⚠</small>	202 ms	43500 KB
318204372	May/03/2025 11:57 UTC+2	RaduM <small>00:37</small>	D - Array and GCD	C++17 (GCC 7-32)	Accepted	187 ms	5500 KB
	May/03/2025		D - Array and GCD	C++23 (GCC 14-64)			

Code: problem dataset & distillation



GOSIM

Datasets: open-r1/codeforces-cots like 152 Follow Open R1 1.49k

Datasets: open-r1/codeforces like

Modalities: Tabular Text parquet

Subset (11) solutions · 47.8k rows

Split (1) train · 47.8k rows

Dataset card Data Studio Files and versions

Dataset Viewer (First 5GB)

Split (2) train · ~9.55k rows (showing the first 9.56k)

Search this dataset

prompt	generation	finish_reason	api_metadata	interaction_format	messages
string · lengths 1.73k-2.9k 51.9%	string · lengths 1.28k-14.2k 12%	stop 95.7%	dict stop	string · classes null 98%	list · lengths [{ "content": "You will be given a competitive programming problem. Please reason step by step about the solution, then provide a complete implementation in C++17." }] null
string · lengths 3 8 1 19 0 1	You will be given a competitive programming problem. Please reason step by step about the solution, then provide a complete implementation in C++17. <think> Okay, let's try to figure out how to solve this problem. The problem is that we have an array of odd length, and we can repeatedly remove pairs of adjacent elements until there's one left. We need to find the maximum possible remaining element after all possible operations. Your solution must read input from standard input (cin), write output to standard output (cout). Do not include any debug prints or additional output. Put your final solution within a single code block: ```cpp <your code here> ``` # Problem You are given an array \$\$\\$a\$\$\$ of \$\$\\$n\$\$\$ integers, where \$\$\\$n\$\$\$ is odd. In one operation, you will remove two adjacent elements from the array \$\$\\$a\$\$\$. and then	stop Okay, let's try to figure out how to solve this problem. The problem is that we have an array of odd length, and we can repeatedly remove pairs of adjacent elements until there's one left. We need to find the maximum possible remaining element after all possible operations. Your solution must read input from standard input (cin), write output to standard output (cout). Do not include any debug prints or additional output. Put your final solution within a single code block: ```cpp <your code here> ``` # Problem You are given an array \$\$\\$a\$\$\$ of \$\$\\$n\$\$\$ integers, where \$\$\\$n\$\$\$ is odd. In one operation, you will remove two adjacent elements from the array \$\$\\$a\$\$\$. and then	dict stop { "completion_tokens": 2504, "prompt_tokens": 878, "prompt_tokens_details": null, "total_tokens": 3382 }	string · classes null	list · lengths [{ "content": "You will be given a competitive programming problem. Please reason step by step about the solution, then provide a complete implementation in C++17." }] null

Code: OlympicCoder



IOI 2024 Eg

Practice Tasks

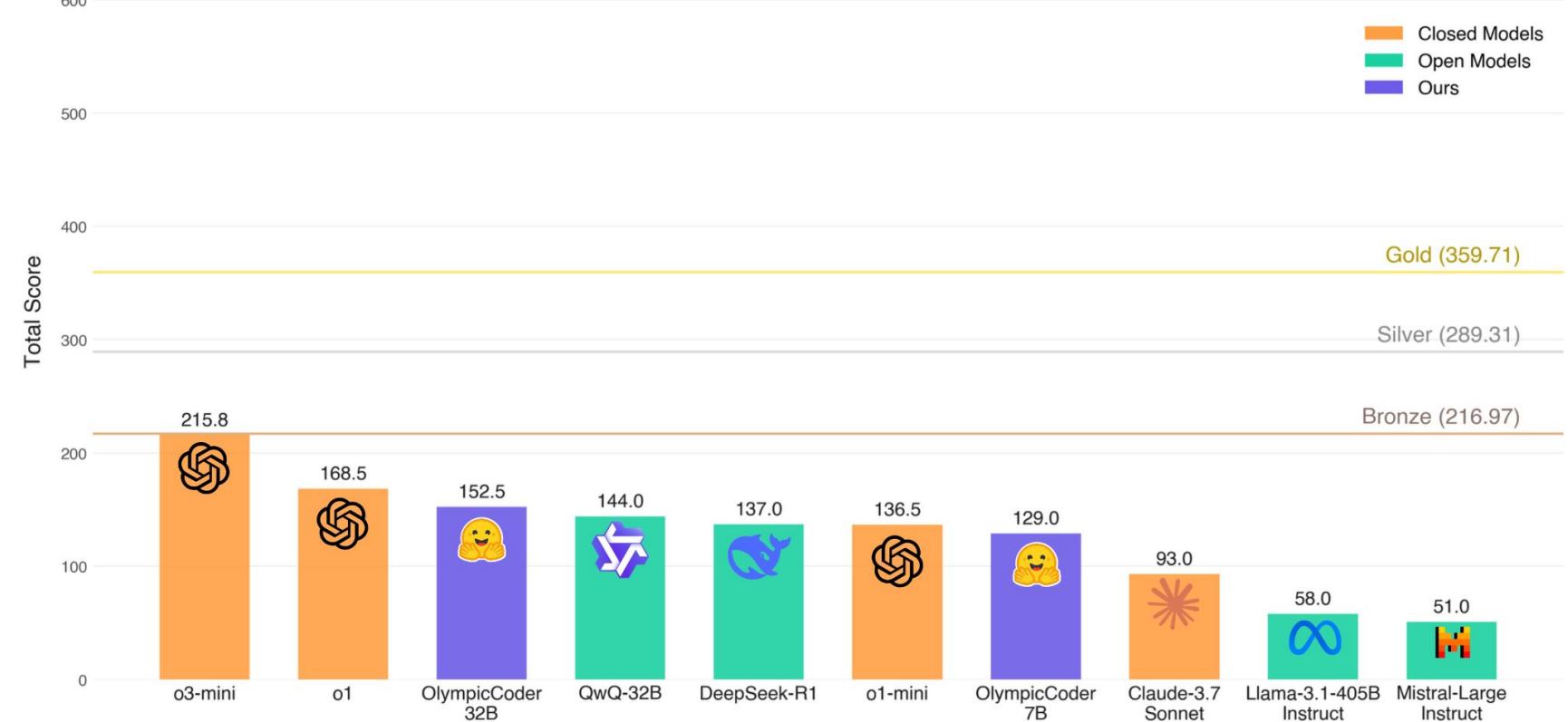
- **aplus**
 - Statement: [Official English](#)
 - [Task attachments](#)
 - [Test data](#)
- **machine**
 - Statement: [Official English](#)
 - [Task attachments](#)
 - [Test data](#)
- **pyramids**
 - Statement: [Official English](#)
 - [Task attachments](#)
 - [Test data](#)
- **treasure**
 - Statement: [Official English](#)
 - [Task attachments](#)
 - [Test data](#)



GOSIM

International Olympiad in Informatics (2024)

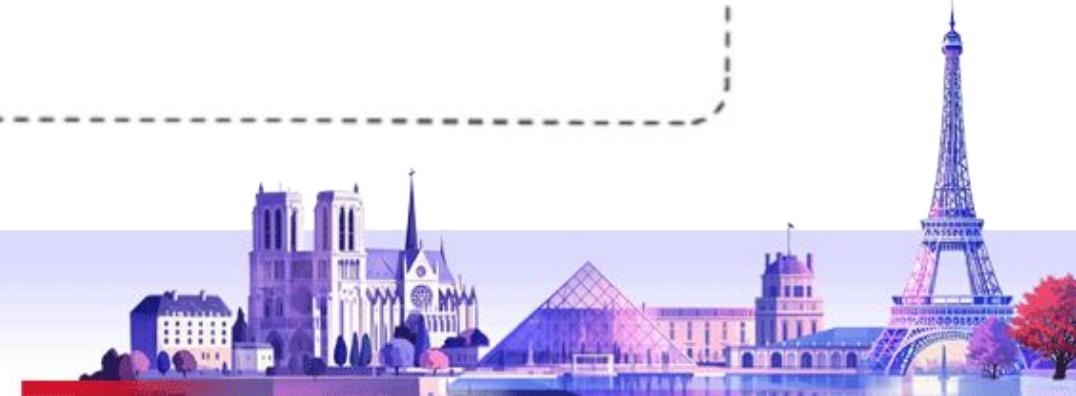
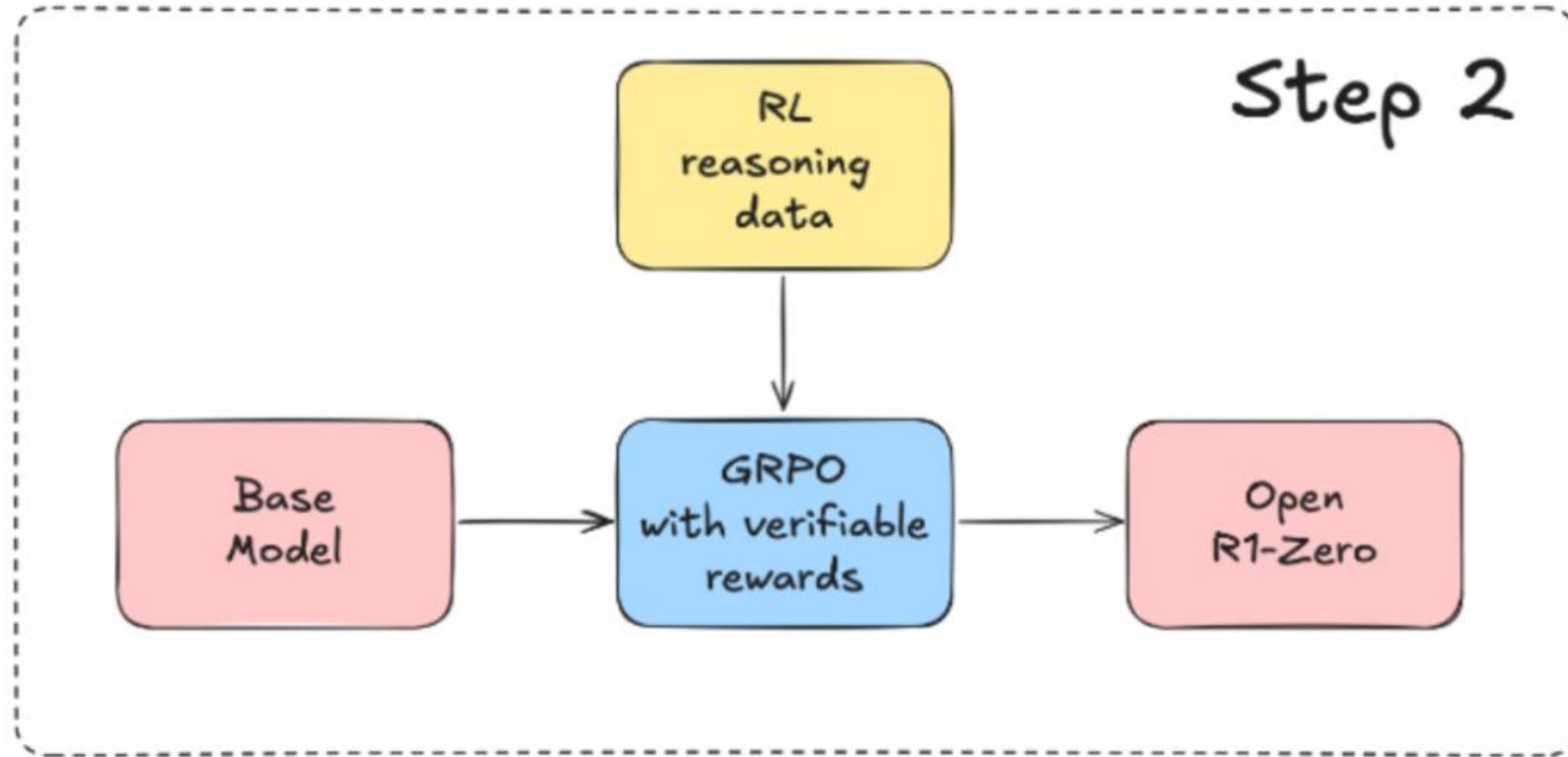
Performance of 10 selected models across 50 submissions



<https://huggingface.co/datasets/open-r1/ioi>

GOSIM AI Paris 2025

Open-R1: Plan of attack #2 – Open-R1-Zero



GRPO in TRL

Quentin Gallouédec ✅
@QGallouedec

TRL 0.14 – Featuring GRPO! 🎉

TRL 0.14 brings *GRPO*, the RL algorithm behind 🐳 DeekSeek-R1.

⚡ Blazing fast generation with vLLM integration.

📈 Optimized training with DeepSpeed ZeRO 1/2/3.

TRL v0.14
GRPO X deepspeed vLLM

```
pip install trl
from trl import GRPOTrainer
trainer = GRPOTrainer("Qwen/Qwen2.5-7B", reward_funcs, train_dataset)
```

3:53 PM · Jan 30, 2025 · 12.7K Views

GRPOTrainer

`GRPOTrainer` implements the [Group Relative Policy Optimization \(GRPO\) algorithm](#) that is more memory-efficient than PPO and was used to train [Deepseek AI's R1](#).

```
from datasets import load_dataset
from trl import GRPOTrainer

dataset = load_dataset("trl-lib/tldr", split="train")

# Dummy reward function: count the number of unique characters
def reward_num_unique_chars(completions, **kwargs):
    return [len(set(c)) for c in completions]

trainer = GRPOTrainer(
    model="Qwen/Qwen2-0.5B-Instruct",
    reward_funcs=reward_num_unique_chars,
    train_dataset=dataset,
)
trainer.train()
```

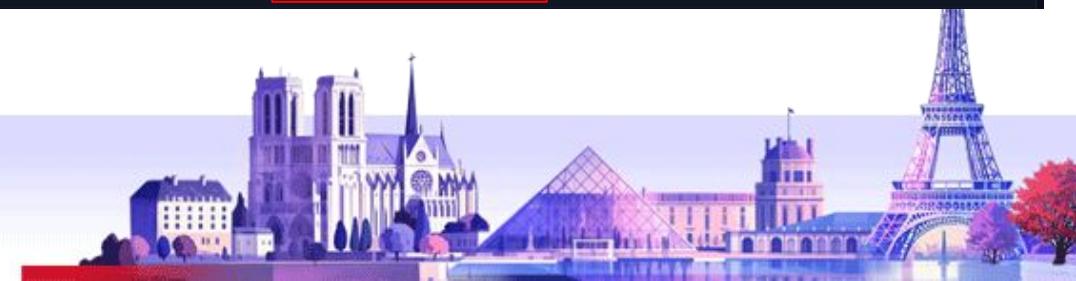
Code verifiability crisis



“verifiable” problems with weak tests only

5	
97	
118	
535	
987	
74	
918	
671	
15	
15	
13	
223	...
Answer	
2298	
48	

Datasets: deepmind/code_contests 0 like 166 Follow Deepmind 275	Dataset c
Split (3) train · 3.76k rows	
Filtered by source with 2 ×	Share results 2,207 rows (limited to the first 5GB)
public_tests sequence	private_tests sequence
{ "input": ["5 5\n1 2 3 4 5\n", "3 1\n1 1 2\n", "5 2\n1 1 2 1 3\n"], "output": ["16", "3", "14"] }	{ "input": ["50 1\n50 8 46 9 12 38 41 18 49 10 23 15 16 3 13 17 48 8 31 32 6 31 31 49 9 58 31 61 4 59 5 55 33 41 81 55 58 23 95 98 60 62 54 94 47 33 20 67 31 67 34 26 47 96 96 1"] }
{ "input": ["6 4\n2 2 5 5 6 6\n1 3\n2 4\n3 5\n4 6\n", "3 3\nn314 159 265\n1 2\n1 3\n3 2\n", "4 2\n2 1 1 5\n1 2\n3 4\n"], "output": ["LOSE\n", "WIN\n0 0\n", "WIN\n1 5 1\n", "LOSE\n"] }	{ "input": ["2 1\n1000000000 100000000\n2 1\n", "3 2\nn123 345 567\n1 2\n3 2\n", "3 0\n\n", "WIN\n0 0\n", "WIN\n678678678 395063145 1073741823 2 3 1\n", "WIN\n0 4 4\n", "LO"] }
{ "input": ["3 10\n2 3 2\n6\n1 5\n2 1 10\n2 2 9\n2 5 6\n3 1 7 8\n4 1 2 3 4\n"], "output": ["\nYES\nYES\nNO\nNO\nYES\n"] }	{ "input": ["", "output": []] }
{ "input": ["2\n1 2\n", "3\n1 2\n2 3\n", "5\n1 2\n1 3\n1 4\n2 5\n", "6\n1 2\n1 3\n1 4\n2 5\n6\n"], "output": ["YES", "NO", "NO", "YES"] }	{ "input": ["50\n16 4\n17 9\n31 19\n22 10\n8 1\n40 30\n3 31\n20 29\n47 27\n22 25\n34 35\n24 41\n30 15\n31 27\n8 44\n26 5\n26 48\n8 32\n53 17\n3 34\n3 51\n30 28\n47 10\n53 60"] }
{ "input": ["3\n0 0 0\n", "5\n0 1 1 1 10\n", "2\n0 1\n"], "output": ["3 2 1 ", "1 4 3 2 5 ", "1 2 "] }	{ "input": ["100\n0 57 121 57 0 19 251 19 301 19 160 57 578 664 57 19 50 0 621 91 5 289 42 31 36 5 68 37 10 49 24 7 32 65 59 14 12 11 6 27 34 91 72 21 87 98 3 97 25 100 46 8"] }
{ "input": ["3\n10 20 10\n6\n1 1\n2 1\n2 3 1\n3 2\n3 3\n", "7\n1 2 1 3 1 2 1\n9\n2 1\n2 3 1\n3 2\n3 3\n1\n1\n7 1\n7\n7 4\n"], "output": ["20\n10\n20\n10\n20\n10\n", "2\n3\n2\n3\n1\n2\n3\n4\n1\n2\n1\n2\n1\n"] }	{ "input": ["2\n1 10\n3\n2 2\n1\n1 1\n", "2\n3922 3922\n3\n2\n2\n1\n1\n", "1\n1000"] }
{ "input": ["6 3\nn3 1 2 3 2 3\n", "10 1\n1 1 1 1 1 1 1\n", "12 4\n1 2 3 1 2 3 4 1 2 1 2 1\n"], "output": ["\n6\n", "\n10\n", "\n12\n"] }	{ "input": ["", "output": []] }
{ "input": ["3\n1 2 3\n", "5\n5 4 3 2 1\n"], "output": ["0 0\n", "0 1 3 6 10\n"] }	{ "input": ["1\n1\n", "100\n98 52 63 2 18 96 31 58 84 40 41 45 66 100 46 71 26 48 81 20 1364 1377 1409 1445 1455 1461 1534 1553 1565 1572 1581 1664 1706 1715 1779 1787 1837 184"] }
{ "input": ["5 3\nn0011\n3\n1 2 3\n1\n4\n3\n4 5\n", "8 6\nn00110011\n3\n1 3 8\n5\n1 2 5 6 7\n2\n6 8\n2\n3 5\n2\n4 7\n1\n1\n2\n"], "output": ["19 5\nn00100100110000110\n2 3\n2 5\n6\n2 8 9\n5\n12 13 14 15 16\n1\n19\n", "7 3\nn001100\n3\n1 4 6\n3\n"] }	{ "input": ["1 1\n1\n", "0\n"] }
{ "input": ["3\n1 3 2\nn-100 2 3\n", "2\n2 1\n-n-3 0\n", "5\n2 1 4 3 5\n2 2 3 4\n"], "output": ["3\n", "0\n", "19\n"] }	{ "input": ["", "output": []] }



Code verifiability crisis



GOSIM

Problem	Lang	Verdict	Time	Memory
1246E - To Make 1	PyPy 3-64	Wrong answer on test 4	93 ms	0 KB
1148G - Gold Experience	PyPy 3-64	Wrong answer on test 14	687 ms	124500 KB
1148H - Holy Diver	PyPy 3-64	Time limit exceeded on test 26	3000 ms	72600 KB
1515H - Phoenix and Bits	PyPy 3-64	Time limit exceeded on test 7	4000 ms	65400 KB
1056H - Detect Robots	PyPy 3-64	Wrong answer on test 3	77 ms	0 KB
1515H - Phoenix and Bits	Python 3	Time limit exceeded on test 7	4000 ms	59600 KB
1491I - Ruler Of The Zoo	Python 3	Wrong answer on test 4	406 ms	900 KB



Code verifiability crisis



2025-01-06



GOSIM

CODEELO: Benchmarking Competition-level Code Generation of LLMs with Human-comparable Elo Ratings

B. Replace Character

time limit per test: 1 second

memory limit per test: 256 megabytes

You're given a string s of length n , consisting of only lowercase English letters.

You must do the following operation exactly once:

- Choose any two indices i and j ($1 \leq i, j \leq n$). You can choose $i = j$.
- Set $s_i := s_j$.

You need to minimize the number of distinct permutations[†] of s . Output any string with the smallest number of distinct permutations after performing **exactly one** operation.

[†] A permutation of the string is an arrangement of its characters into any order. For example, "bac" is a permutation of "abc" but "bcc" is not.

Input

Each test contains multiple test cases. The first line contains the number of test cases t ($1 \leq t \leq 500$). The description of the test cases follows.

The first line of each test case contains n ($1 \leq n \leq 10$) — the length of string s .

A clear indication that needs a special judge

The second line of each test case contains s of length n . The string contains only lowercase English letters.

Output

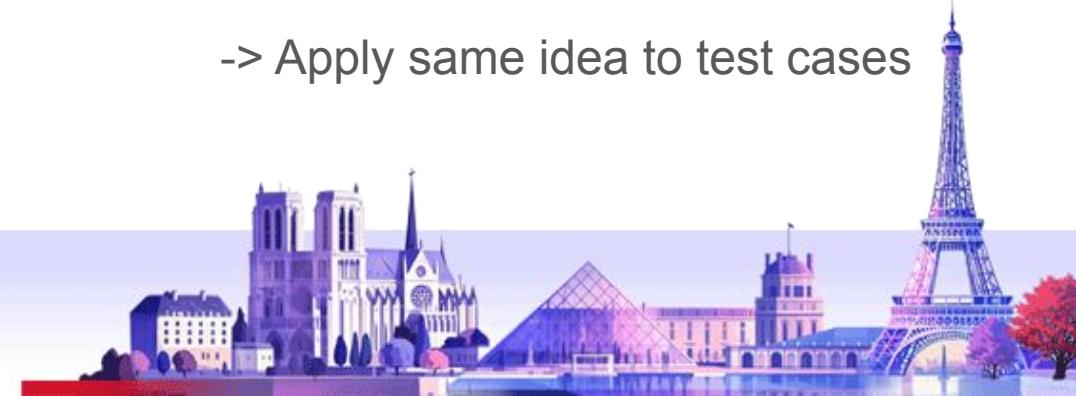
For each test case, output the required s after applying exactly one operation. If there are multiple solutions, print any of them.

~30% require a special checker

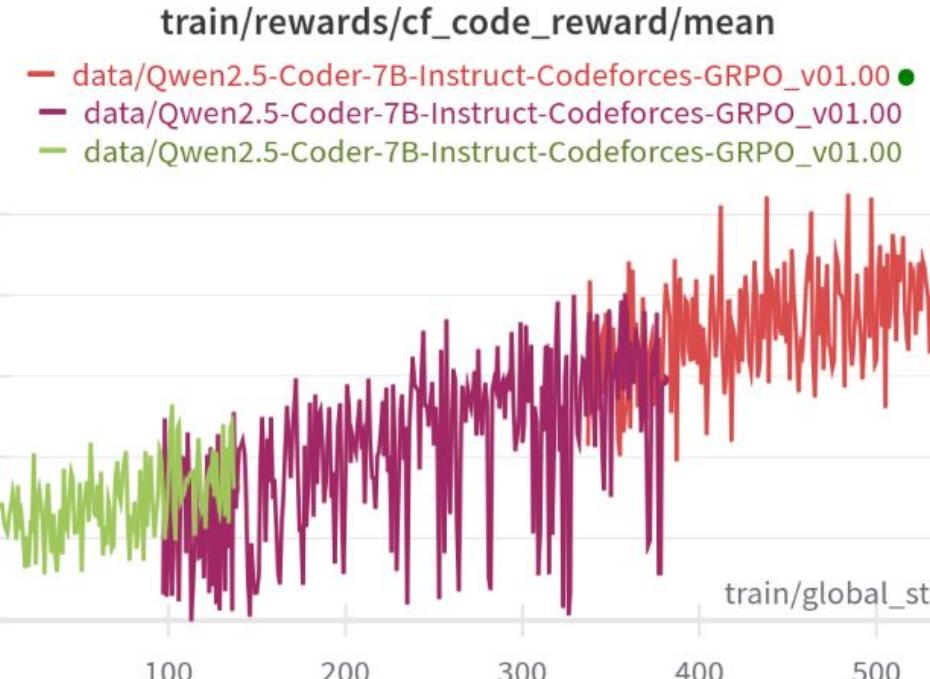
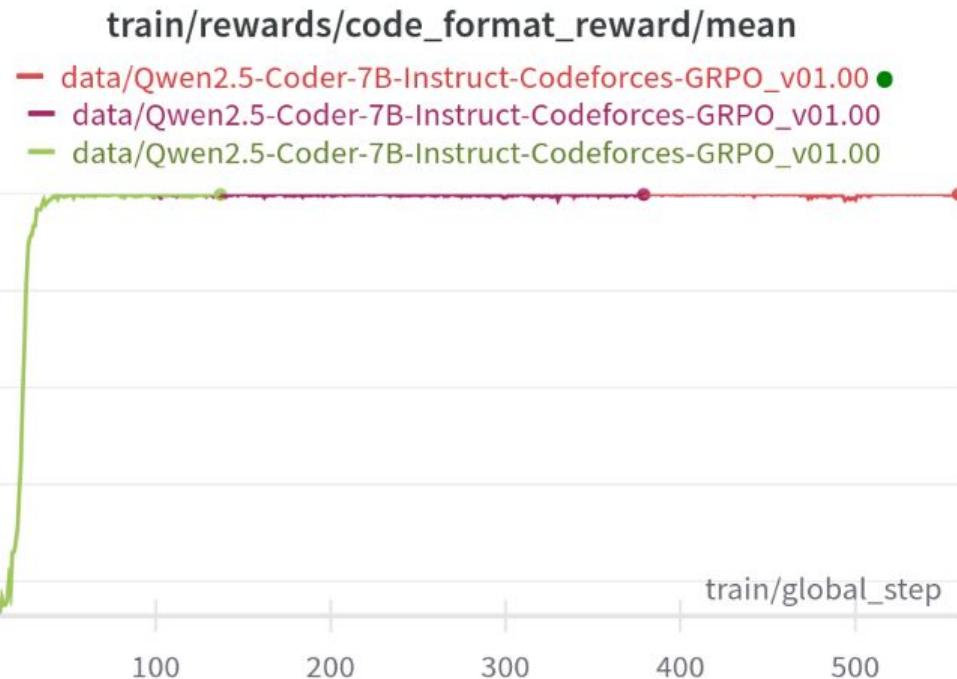
R1 generated checkers working for 60%+ of them

Validated with correct and incorrect solutions

-> Apply same idea to test cases

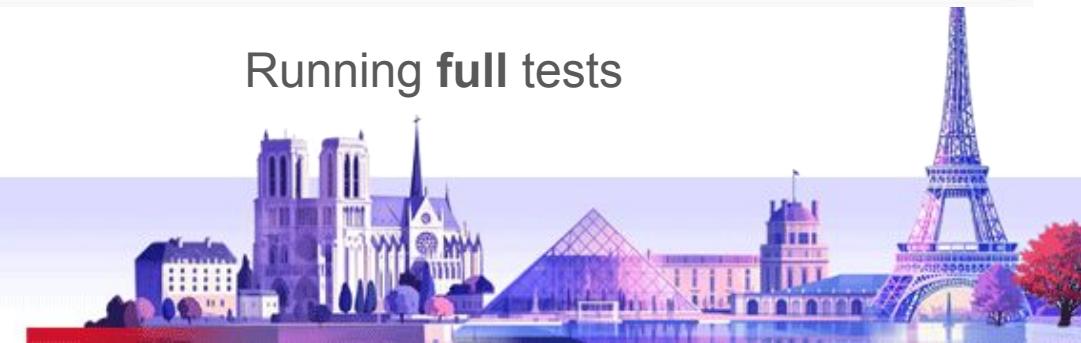


Code reward curves

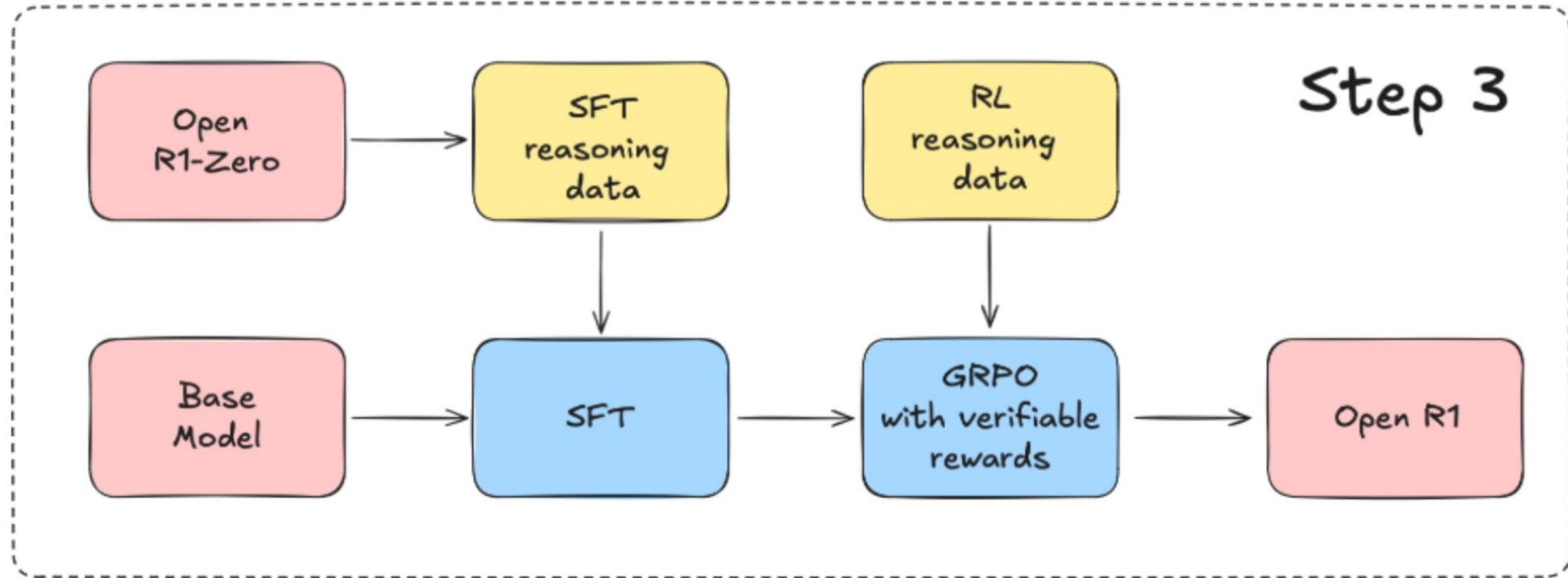


```
rf"^(?<think>\n.*?\n)</think>\n<answer>\n.*?\`{language}.*?\`.*?\n)</answer>$"
```

Running **full** tests



Open-R1: Plan of attack #3 – Open-R1



THANK YOU

 github.com/huggingface/open-r1

 hf.co/open-r1