Post-training open agents for long-horizon tasks

Open recipes with DR Tulu and TMax

INF 385T - Hamish Ivison - 10/28/26

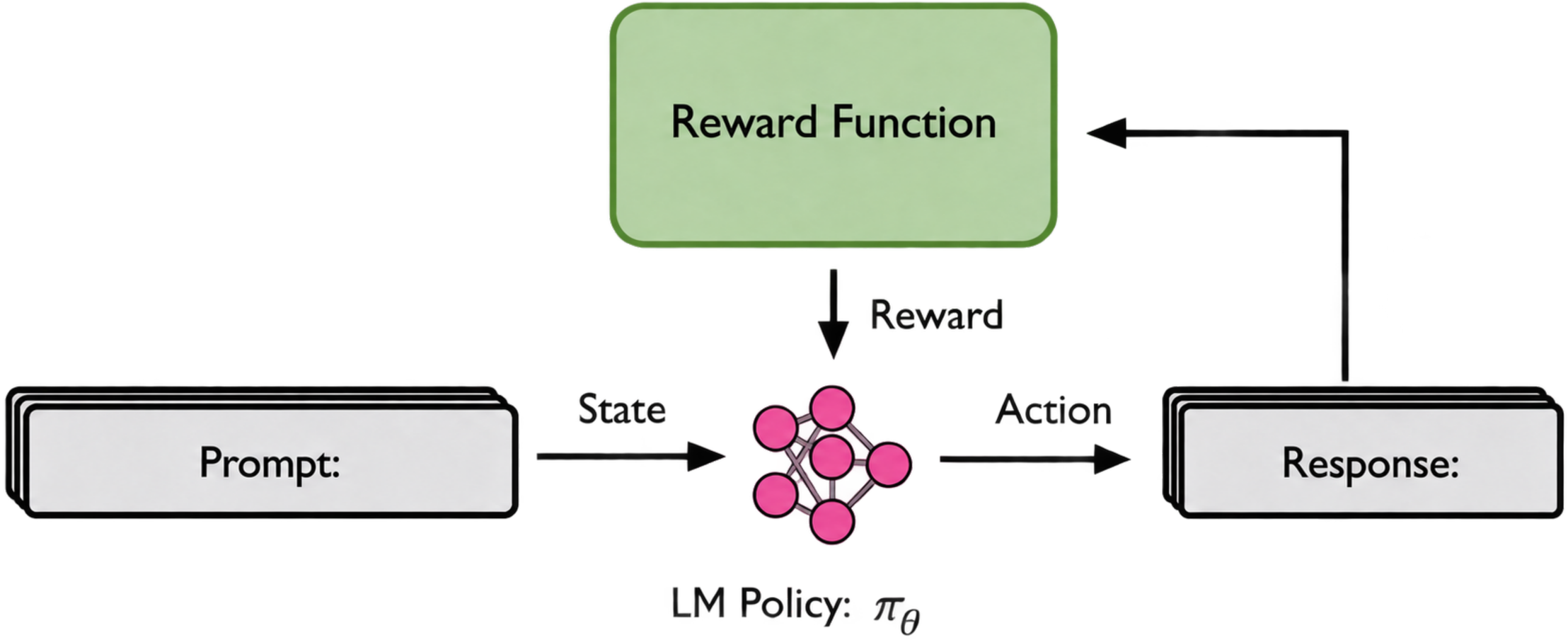


Talk Outline

1. Quick RL Refresher
2. Evolving rubrics for Deep Research with DR Tulu 
3. Training Terminal Agents with TMax 

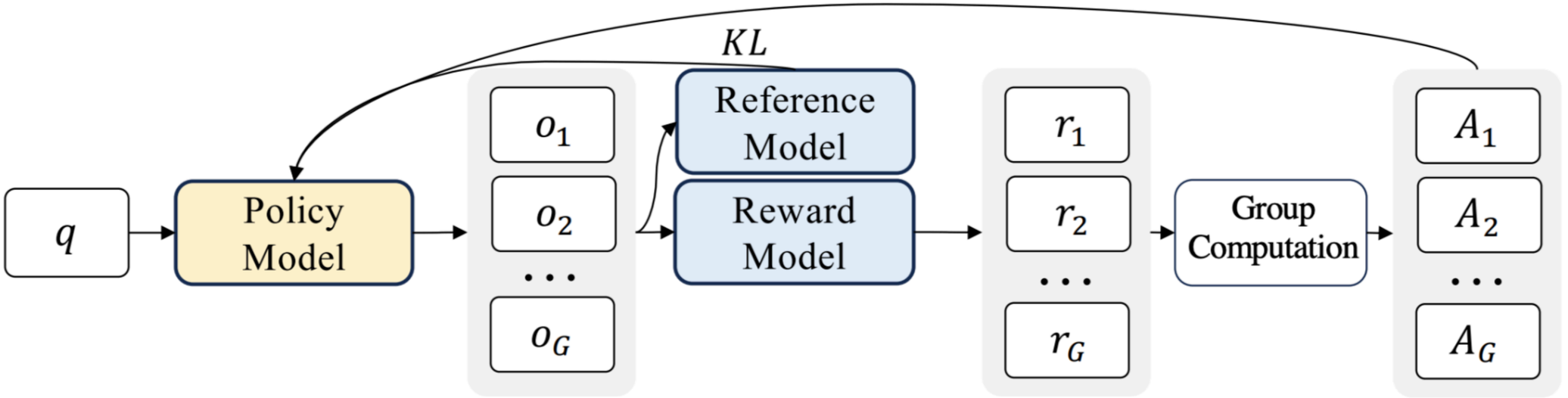
Quick RL Refresher!

We will be using **GRPO** (or variants thereof) throughout these works.



Quick RL Refresher!

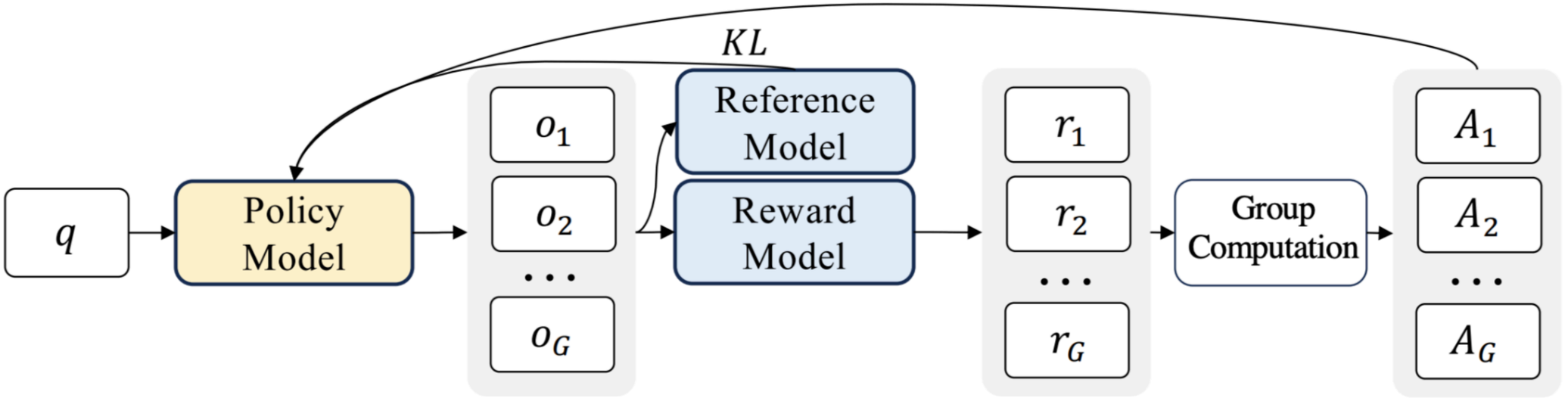
We will be using **GRPO** (or variants thereof) throughout these works.



DeepSeekMath: Pushing the Limits of Mathematical Reasoning in Open Language Models. Shao et el 2024.

Quick RL Refresher!

We will be using **GRPO** (or variants thereof) throughout these works.



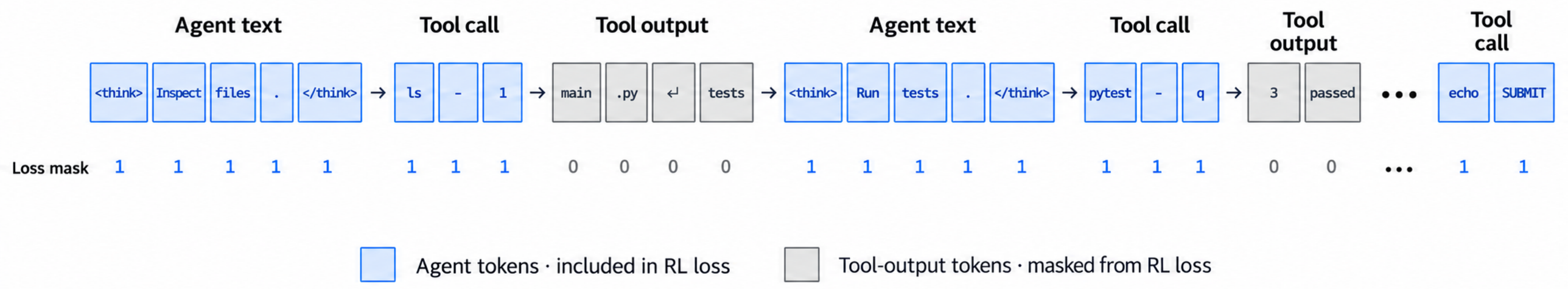
DeepSeekMath: Pushing the Limits of Mathematical Reasoning in Open Language Models. Shao et el 2024.

$$A_j = R(x, y_j) - \text{mean}(R_G)$$

Set KL to 0. See Olmo 3 paper for more.

Quick RL Refresher!

In particular, we will be applying RL over **long tool-using horizons**.
Turns out this ‘just’ works. We simply mask out tool-output tokens!



DR Tulu: Reinforcement Learning with Evolving Rubrics for Deep Research

Rulin Shao[1♡†], **Akari Asai**[2,3♡†], **Shannon Zejiang Shen**[4♡†], **Hamish Ivison**[1,2♡†], **Varsha Kishore**[1,2†], **Jingming Zhuo**[1†], **Xinran Zhao**[3], **Molly Park**[1], **Samuel G. Finlayson**[1,5], **David Sontag**[4], **Tyler Murray**[2], **Sewon Min**[2,6], **Pradeep Dasigi**[2], **Luca Soldaini**[2], **Faeze Brahman**[2], **Wen-tau Yih**[1], **Tongshuang Wu**[3], **Luke Zettlemoyer**[1], **Yoon Kim**[4], **Hannaneh Hajishirzi**[1,2], **Pang Wei Koh**[1,2]

[1]University of Washington, [2]Allen Institute for AI, [3]Carnegie Mellon University
[4]Massachusetts Institute of Technology, [5]Seattle Children's Hospital, [6]University of California, Berkeley



What is a deep research agent?

Deep Research (DR) agents

What are the differences between OLMo2 and OLMo3?

Differences Between OLMo2 and OLMo3 Language Models

I'm thinking through whether OLMo2 and OLMo3 are open language models from the Allen Institute, similar to GPT-like models, and noting the need to pinpoint their exact nature and versions.

Searched for OLMo2 and OLMo3 models

Read allenai.org

OK, let's see: Olmo 2's specifics and its differences from Olmo 3 are tricky. I'll review the technical report for more details.

Read arxiv.org

I'm looking at OLMo3's state-of-the-art models, their expansive capabilities, lifecycle management, and comparison with OLMo2. Emails about these topics are ongoing.

Read allenai.org

...

OLMo2 vs OLMo3: Key Differences and Comparison

Architecture and Model Size

- **Model Scale**: **OLMo2** debuted with 7B and 13B parameter models (later expanded to a 32B variant) infoq.com huggingface.co . **OLMo3** offers 7B and 32B models as its primary releases, making a significant jump in size at the high end (32B vs. OLMo2's 13B) infoq.com arxiv.org . This larger 32B "Think" model is the new flagship, providing substantially more capacity for complex tasks.
- **Transformer Architecture**: Both OLMos use a decoder-only transformer architecture. **OLMo2** introduced several architectural tweaks to improve stability and performance, such as using **RMSNorm** (instead of standard layer norm), applying **QK-normalization**, reordering layer norms, and switching from absolute to **rotary positional embeddings** allenai.org infoq.com . It also added techniques like **Z-loss regularization** to stabilize training allenai.org . **OLMo3** retains these improvements and further enhances the model – for example, it's explicitly designed for **long-context support** (trained to handle inputs up to ~65k tokens, far beyond OLMo2's 4k context window) huggingface.co venturebeat.com . OLMo3-"Think" models can even output step-by-step reasoning traces (chain-of-thought style), a new capability not present in OLMo2 venturebeat.com .

Training Data and Token Count

- **OLMo2 Training Data**: OLMo2 was trained on **up to ~5 trillion tokens** drawn from diverse high-quality sources infoq.com . Its two-stage training curriculum first used a broad **OLMo-Mix** (~3.9T tokens of data from the web, code repositories, etc.), then a **Dolmino-Mix** (~0.84T tokens) for fine-tuning on more curated domain-specific and instruction-like content infoq.com . ... gave OLMo2 a wide knowledge base with an emphasis on technical and math data to boost reasoning performance

Deep Research (DR) agents

How does it work under the hood?

Deep Research (DR) agents

An agent (LM) is a policy that samples the next action conditioned on the current state:

Sampled action

Current state (current context)

$$(a_i, \square) \sim \pi_\theta(\cdot \mid s_i)$$

Deep Research (DR) agents

An agent (LM) is a policy that samples the next action conditioned on the current state:

Sampled action

Current state (current context)

$$(a_i, \square) \sim \pi_\theta(\cdot \mid s_i)$$

Initialized with the system prompt and the user question.

Deep Research (DR) agents

Sampled action

Current state (current context)

$$(a_i, \square) \sim \pi_\theta(\cdot \mid s_i)$$

We consider 3 actions:

Deep Research (DR) agents

Sampled action

Current state (current context)

$$(a_i, \square) \sim \pi_\theta(\cdot \mid s_i)$$

We consider 3 actions:

1. `think` (`<think></think>`)

   generate intermediate reasoning

Deep Research (DR) agents

Sampled action

Current state (current context)

$$(a_i, \square) \sim \pi_\theta(\cdot \mid s_i)$$

We consider 3 actions:

1. `think` (`<think></think>`)
2. `tool` (`<call_tool></call_tool>`)

issue a search query & inspect retrieved content.

Deep Research (DR) agents

Sampled action

Arguments

Current state (current context)

$$(a_i, \zeta_i) \sim \pi_\theta(\cdot \mid s_i)$$

We consider 3 actions:

1. `think` (<think></think>)
2. `tool` (<call_tool></call_tool>)

issue a search query & inspect retrieved content.

Example: <call_tool name="google_search" k="10" lang="en">query</call_tool>

Deep Research (DR) agents

Sampled action

Current state (current context)

$$(a_i, \zeta_i) \sim \pi_\theta(\cdot \mid s_i)$$

Searched results will also be appended to the context

We consider 3 actions:

1. `think` (`<think></think>`)
2. `tool` (`<call_tool></call_tool>`)

issue a search query & inspect retrieved content.

Example: <call_tool name="google_search" k="10" lang="en">query</call_tool>

Deep Research (DR) agents

Sampled action

Current state (current context)

$$(a_i, \zeta_i) \sim \pi_\theta(\cdot \mid s_i)$$

We consider 3 actions:

1. `think` (<think></think>)
2. `tool` (<call_tool></call_tool>)
3. `answer` (<answer></answer>)

synthesize final response

Deep Research (DR) agents

Sampled action

Current state (current context)

$$(a_i, \zeta_i) \sim \pi_\theta(\cdot \mid s_i)$$

We consider 3 actions:

1. `think` (<think></think>)
2. `tool` (<call_tool></call_tool>)
3. `answer` (<answer></answer>)
   - `cite` (<cite id="SOURCE_ID"></cite>)

Overview

<cite id="8b9f3729-0">Netflix's adaptation avoided a literal replica of Macondo and instead fused real locations with meticulously built sets to honor the novel's essence while giving the show contemporary visual grammar.</cite><cite id="6188e90a-0">The production grounded magical realism in front-of-camera practical craft, relying on makeup, special effects, and 3D scanning to keep the fantastical tethered to tangible performances and environments.</cite>

Location Strategy and Production Design

…

Pacing and Narrative Focus

<cite id="4aaa422c-2">Major outlets judged the adaptation as faithful in spirit yet fresh in execution, crediting the production's design, effects, and performances for making García Márquez's epic legible on screen.</cite>

Deep Research (DR) agents

What are the differences between OLMo2 and OLMo3?

Differences Between OLMo2 and OLMo3 Language Models

I'm thinking through whether OLMo2 and OLMo3 are open language models from the Allen Institute, similar to GPT-like models, and noting the need to pinpoint their exact nature and versions.

Searched for OLMo2 and OLMo3 models

Read allenai.org

OK, let's see: Olmo 2's specifics and its differences from Olmo 3 are tricky. I'll review the technical report for more details.

Read arxiv.org

I'm looking at OLMo3's state-of-the-art models, their expansive capabilities, lifecycle management, and comparison with OLMo2. Emails about these topics are ongoing.

Read allenai.org

...

OLMo2 vs OLMo3: Key Differences and Comparison

Architecture and Model Size

- **Model Scale**: **OLMo2** debuted with 7B and 13B parameter models (later expanded to a 32B variant) infoq.com huggingface.co . **OLMo3** offers 7B and 32B models as its primary releases, making a significant jump in size at the high end (32B vs. OLMo2's 13B) infoq.com arxiv.org . This larger 32B "Think" model is the new flagship, providing substantially more capacity for complex tasks.
- **Transformer Architecture**: Both OLMos use a decoder-only transformer architecture. **OLMo2** introduced several architectural tweaks to improve stability and performance, such as using **RMSNorm** (instead of standard layer norm), applying **QK-normalization**, reordering layer norms, and switching from absolute to **rotary positional embeddings** allenai.org infoq.com . It also added techniques like **Z-loss regularization** to stabilize training allenai.org . **OLMo3** retains these improvements and further enhances the model – for example, it's explicitly designed for **long-context support** (trained to handle inputs up to ~65k tokens, far beyond OLMo2's 4k context window) huggingface.co venturebeat.com . OLMo3-"Think" models can even output step-by-step reasoning traces (chain-of-thought style), a new capability not present in OLMo2 venturebeat.com .

Training Data and Token Count

- **OLMo2 Training Data**: OLMo2 was trained on **up to ~5 trillion tokens** drawn from diverse high-quality sources infoq.com . Its two-stage training curriculum first used a broad **OLMo-Mix** (~3.9T tokens of data from the web, code repositories, etc.), then a **Dolmino-Mix** (~0.84T tokens) for fine-tuning on more curated domain-specific and instruction-like content infoq.com . ... gave OLMo2 a wide knowledge base with an emphasis on technical and math data to boost reasoning performance

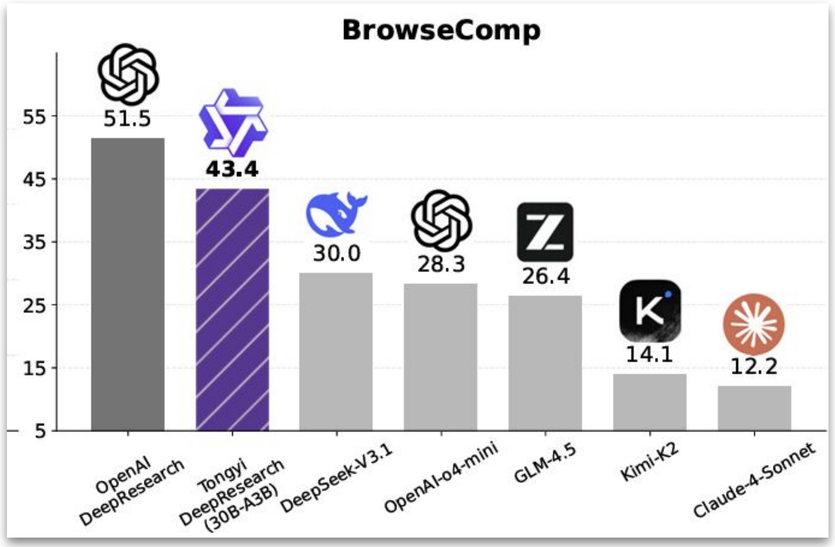
The open-source progress

Prior work focuses on short-form DR tasks

Please identify the fictional character who occasionally breaks the fourth wall with the audience, has a backstory involving help from selfless ascetics, is known for his humor, and had a TV show that aired between the 1960s and 1980s with fewer than 50 episodes.

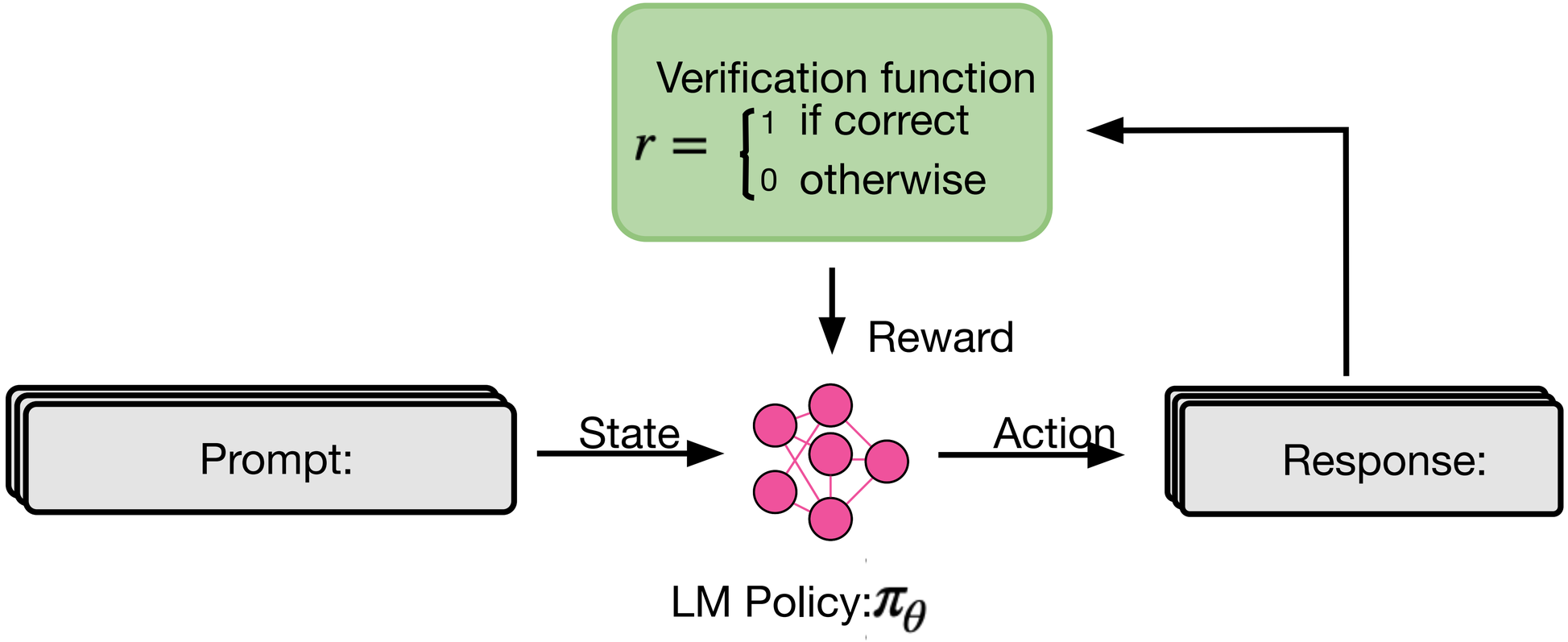
Plastic Man

BrowseComp: A Simple Yet Challenging Benchmark for Browsing Agents
Wei et al. 2025



Tongyi Deep Research Technical Report
Tongyi Deep Research Team. 2025

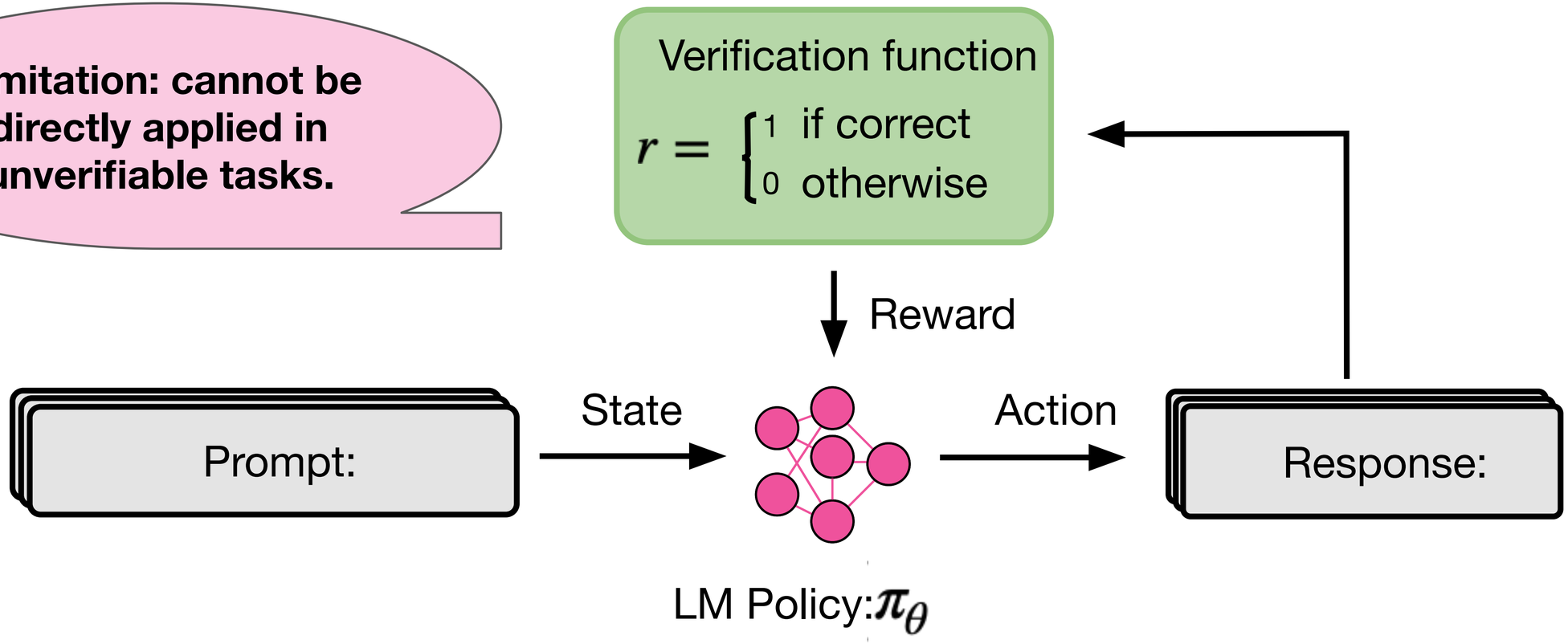
RL technique used in prior work — RLVR



Tulu 3: Pushing Frontiers in Open Language Model Post-Training
Lambert et al. 2024

RL technique used in prior work — RLVR

Limitation: cannot be directly applied in unverifiable tasks.



Tulu 3: Pushing Frontiers in Open Language Model Post-Training
Lambert et al. 2024

DR tasks are open-ended and unverifiable

What's the state-of-the-art in immunotherapy for cervical cancer?

Open Scholar

... Pembrolizumab can extend progression-free and overall survival in patients with persistent, recurrent, or metastatic PD -L1-positive cervical cancer [1] and locally advanced cervical cancer [2]...

[1] Colombo et al., KEYNOTE-826, 2021
[2] Lorusso et al., KEYNOTE-A18, 2024

Attribution

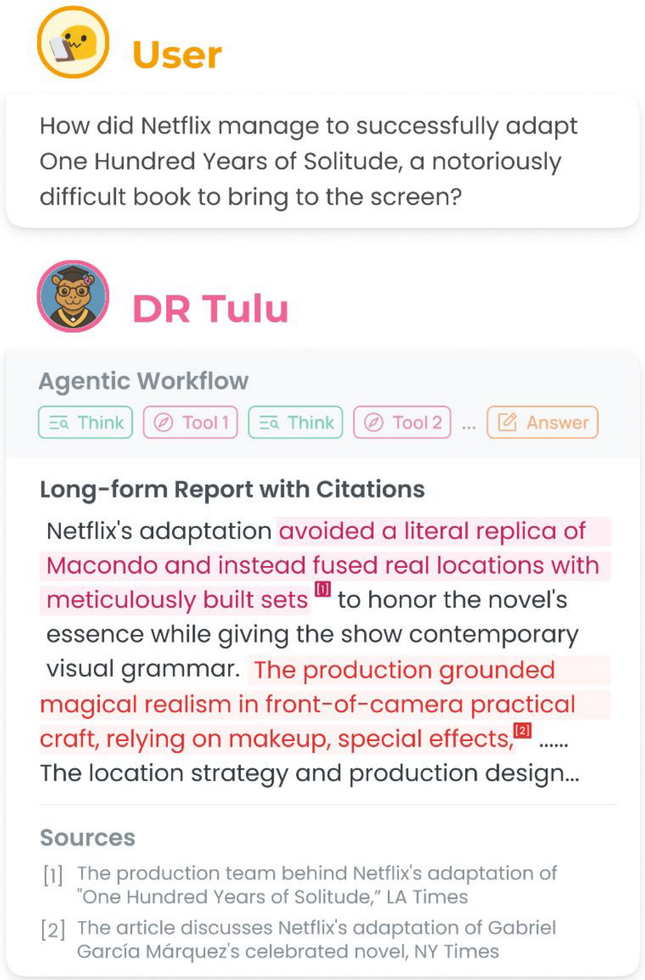
Credible sources

Up-to-date

OpenScholar: Synthesizing Scientific Literature with Retrieval-augmented LMs
Asai et al. Nature 2026.

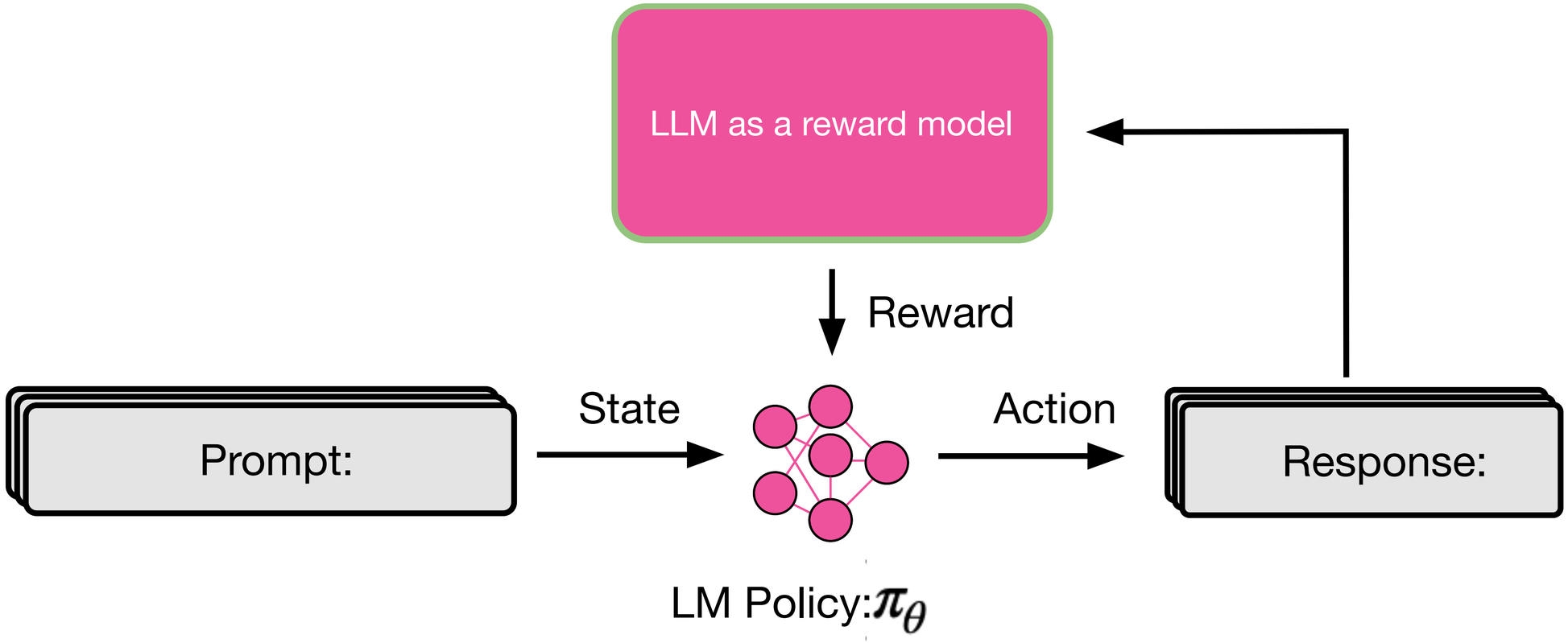
How do we train open DR agents for tasks without simple verification?

DR Tulu: Open DR Agents for Open-ended Tasks



- **Performs both open-ended, long-form tasks and short-form tasks**
- **Reason to choose effective search MCP tools**
- **Provide snippet-level citations for each claim from credible sources**

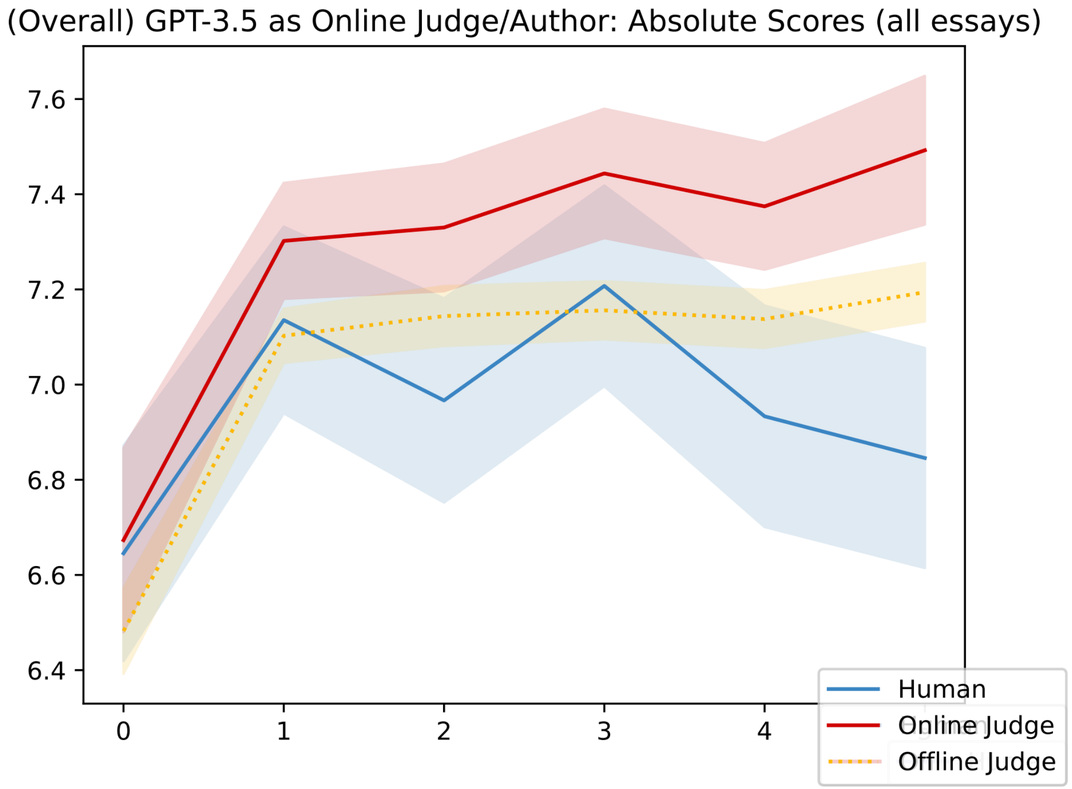
Naive solution: use LLM as a reward model



Tulu 3: Pushing Frontiers in Open Language Model Post-Training
Lambert et al. 2024

Naive solution: use LLM as a reward model

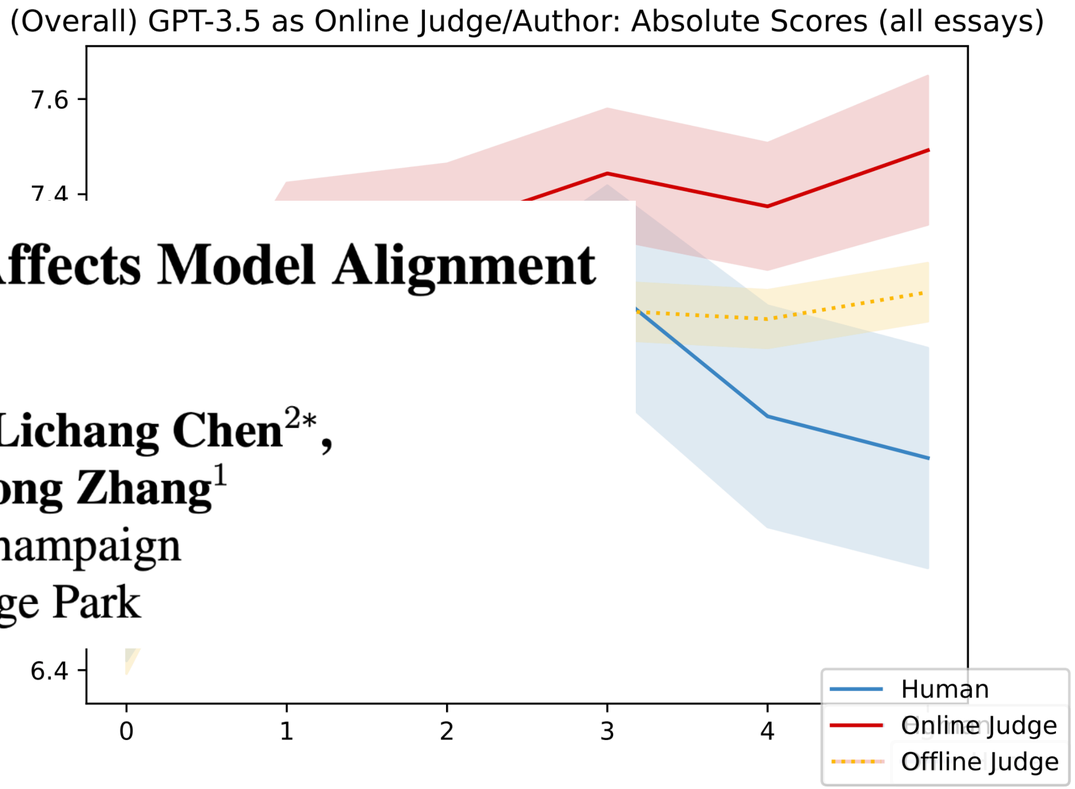
Reward hacking 🙅



Spontaneous Reward Hacking in Iterative Self-Refinement
Jane Pan, et al. 2024

Naive solution: use LLM as a reward model

Reward hacking 🙅



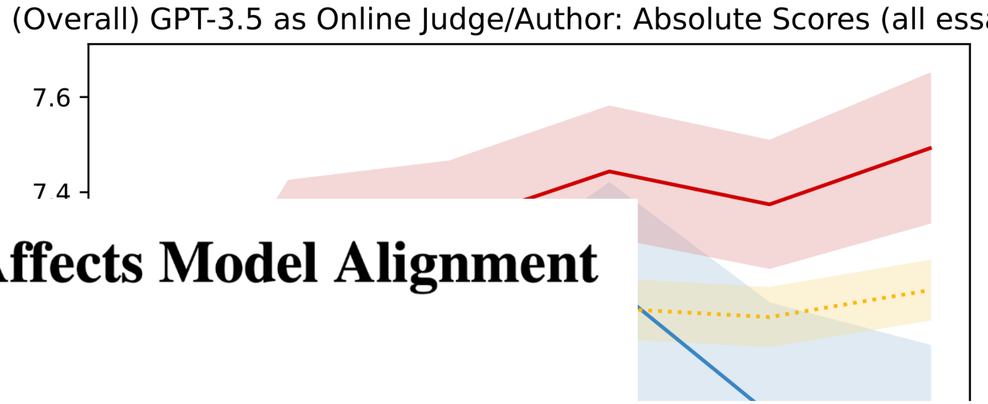
From Lists to Emojis: How Format Bias Affects Model Alignment

Xuanchang Zhang[*] [1], **Wei Xiong**[1*], **Lichang Chen**[2*],
Tianyi Zhou[2], **Heng Huang**[2], **Tong Zhang**[1]
[1]University of Illinois Urbana-Champaign
[2]University of Maryland College Park

Spontaneous Reward Hacking in Iterative Self-Refinement
Jane Pan, et al. 2024

Naive solution: use LLM as a reward model

Reward hacking 🤦



From Lists to Emojis: How Format Bias Affects Model Alignment

Xuanchar
Tian
[1]U
2

One Token to Fool LLM-as-a-Judge

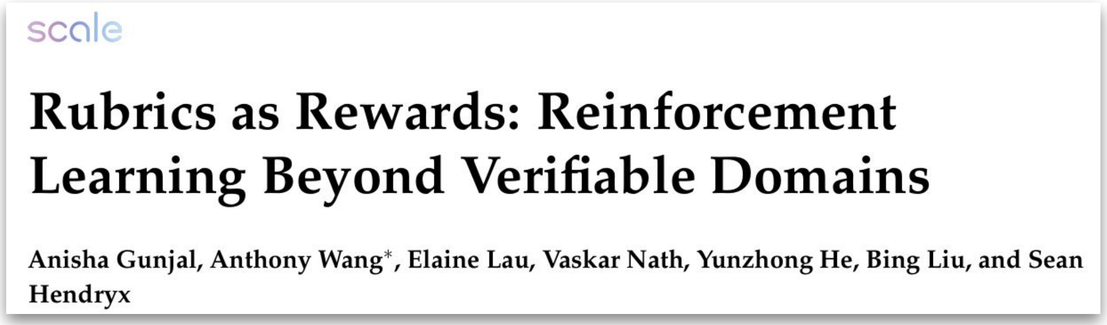
Yulai Zhao*,[1,2] , Haolin Liu*,[1,3] , Dian Yu[1] , S.Y. Kung[2] , Haitao Mi[1] , and Dong Yu[1]

[1]Tencent AI Lab
[2]Princeton University
[3]University of Virginia

Spontaneous Reward Hacking in Iterative Self-Refinement
Jane Pan, et al. 2024

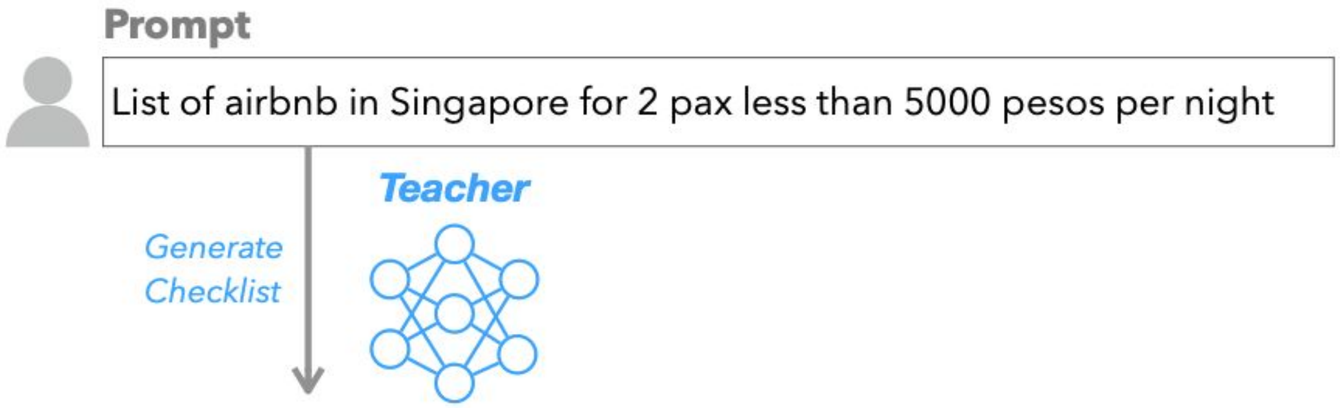
Background: RL with Rubric Rewards

Rubrics alleviate it by making the evaluation less subjective.





Background: RL with Rubric Rewards

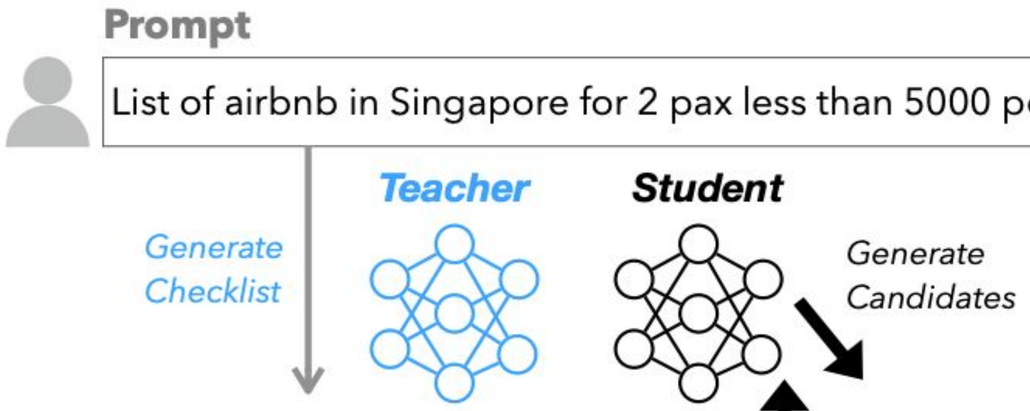


rubrics

1. Does the text provide a list of Airbnbs in Singapore? *(weight: 100/100)*
2. Does each listing accommodate 2 people? *(weight: 100/100)*
3. Does the text provide useful tips if real-time listings are not available? *(weight: 75/100)*

Checklists Are Better Than Reward Models For Aligning Language Models
Viswanathan et al. NeurIPS 2025.

Background: RL with Rubric Rewards



rubrics

1. Does the text provide a list of Airbnbs in Singapore? *(weight: 100/100)*
2. Does each listing accommodate 2 people? *(weight: 100/100)*
3. Does the text provide useful tips if real-time listings are not available? *(weight: 75/100)*

1. 47.9 × 100
2. 0.0 × 100
3. 100.0 × 75

Score 77.0

1. "To find Airbnb's in Singapore for 2 pax (2 people) […]
   Below are a few hypothetical examples …"

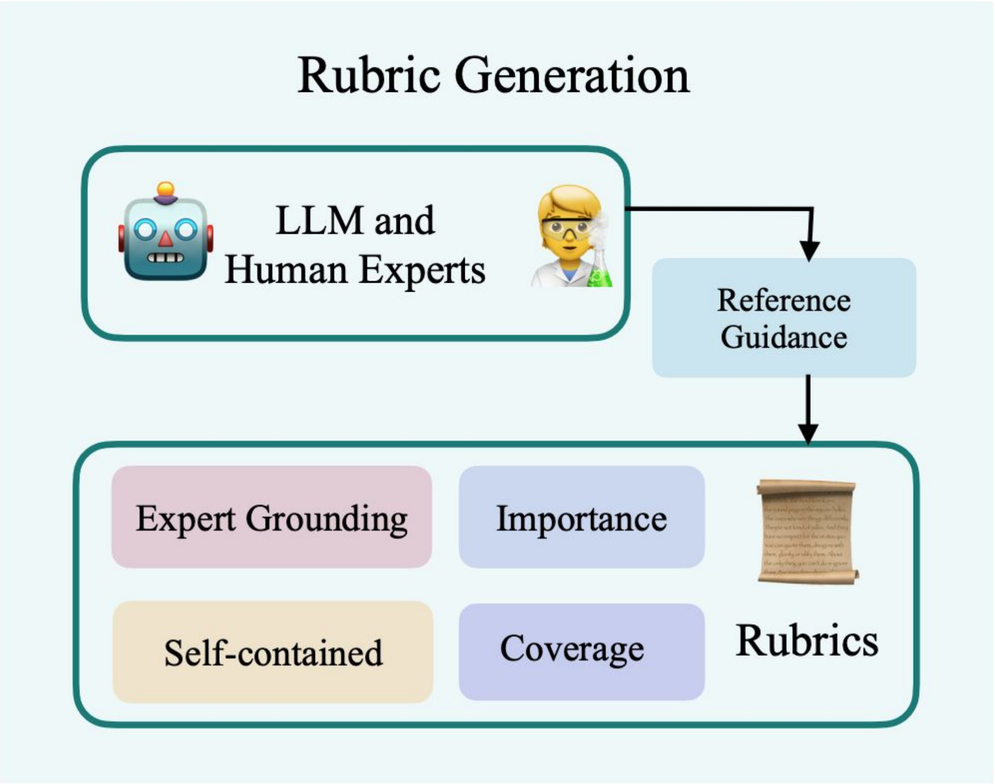
1. 0.0 × 100
2. 85.2 × 100
3. 60.6 × 75

Score 16.5

2. "I apologize, but using pesos (PHP) to describe prices in Singapore may be confusing …"

Checklists Are Better Than Reward Models For Aligning Language Models
Viswanathan et al. NeurIPS 2025.

Problem of static rubrics



- **Require human experts or a strong teacher model**
- **Not grounded on up-to-date knowledge**
- **Not adaptive to the policy distribution**

Rubrics as Rewards: Reinforcement Learning Beyond Verifiable Domains
Gunjal et al. NeurIPS 2025.

RL with Evolving Rubrics (RLER)

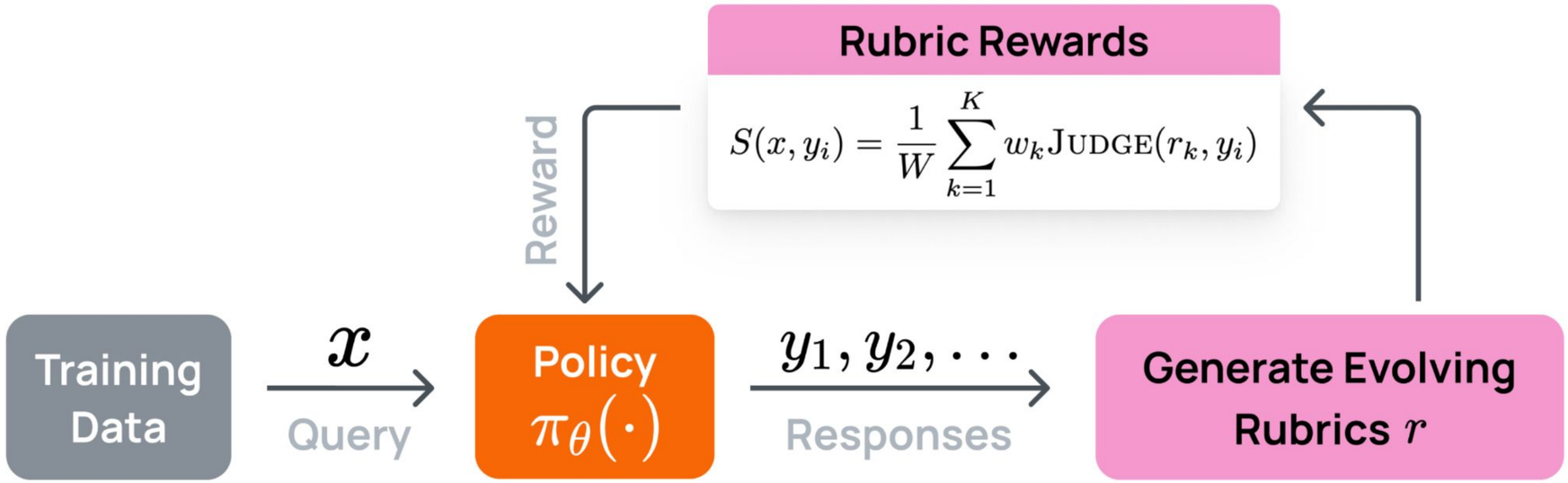
RL with Evolving Rubrics (RLER)

Key idea: Generate rubrics on the fly

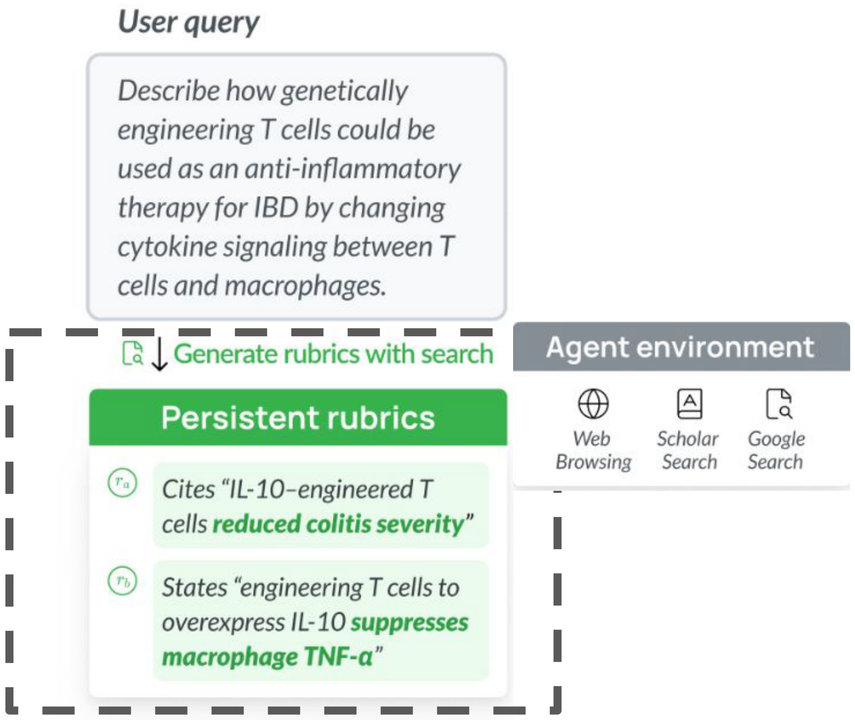
1. Ground rubric generation on **up-to-date searched information.**
2. Generate fine-grained rubrics by **contrasting model responses.**
3. Make rubrics **co-evolve with the policy model**

.

RL with Evolving Rubrics (RLER)

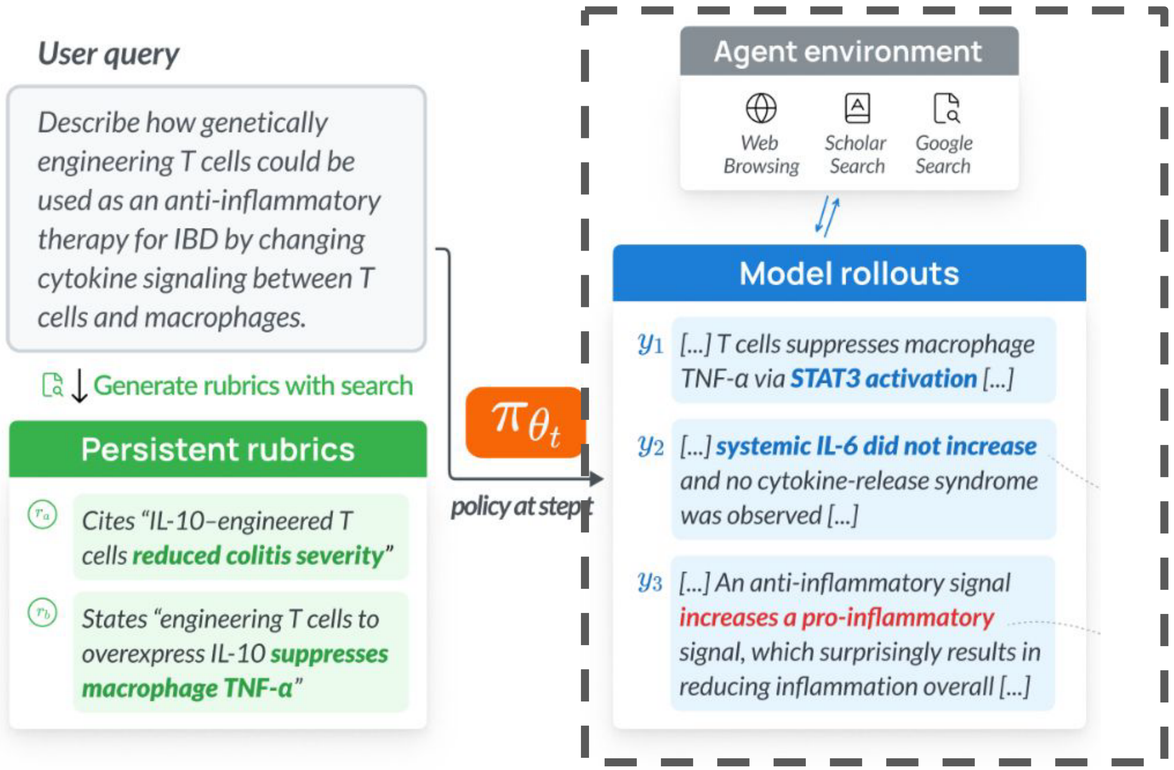


RL with Evolving Rubrics (RLER)



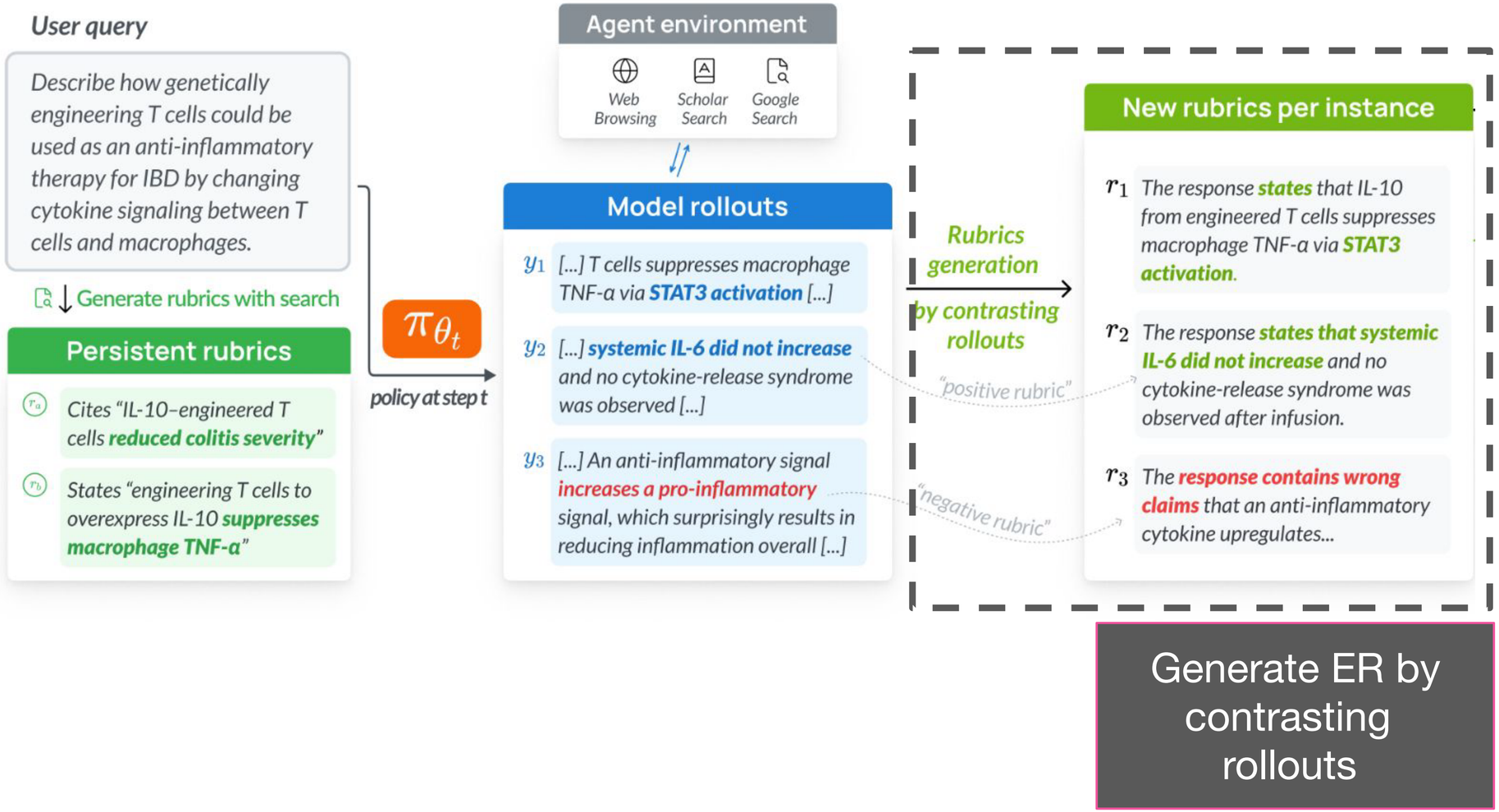
Init. rubrics

RL with Evolving Rubrics (RLER)



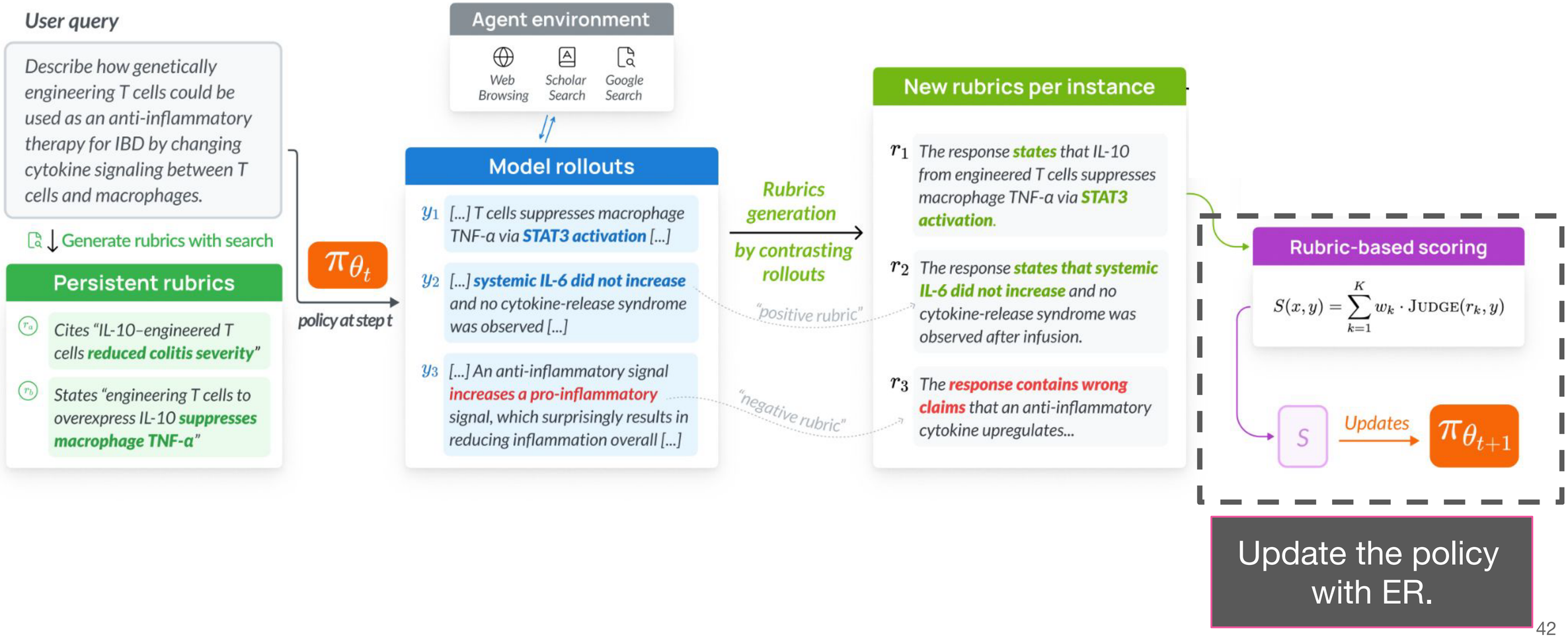
Rollout with search

RL with Evolving Rubrics (RLER)



Generate ER by contrasting rollouts

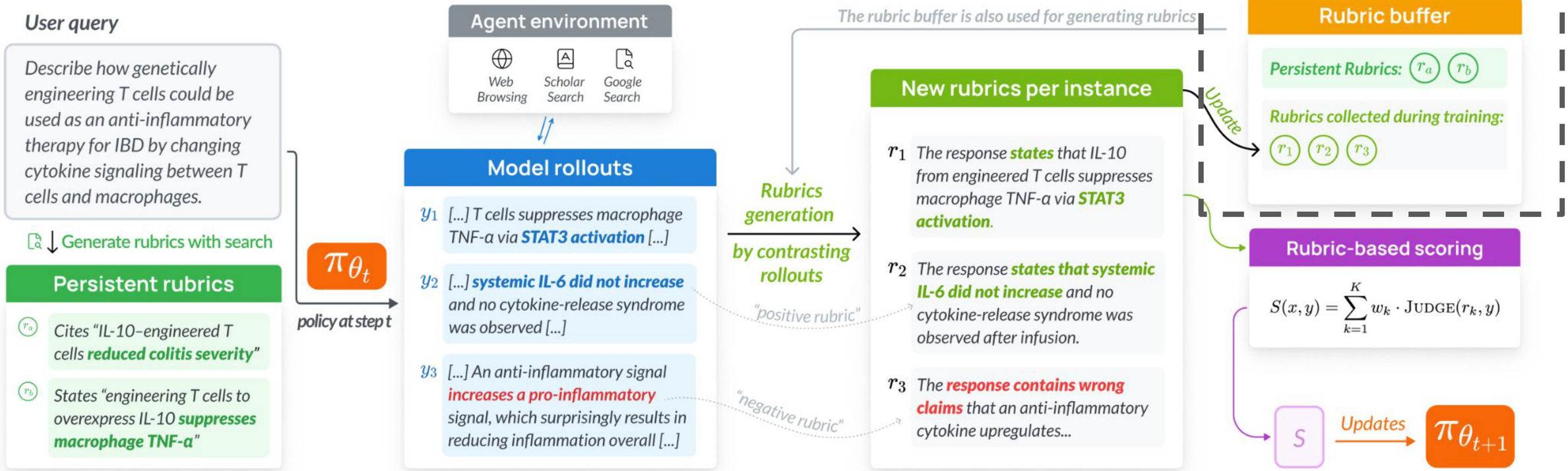
RL with Evolving Rubrics (RLER)



Update the policy with ER.

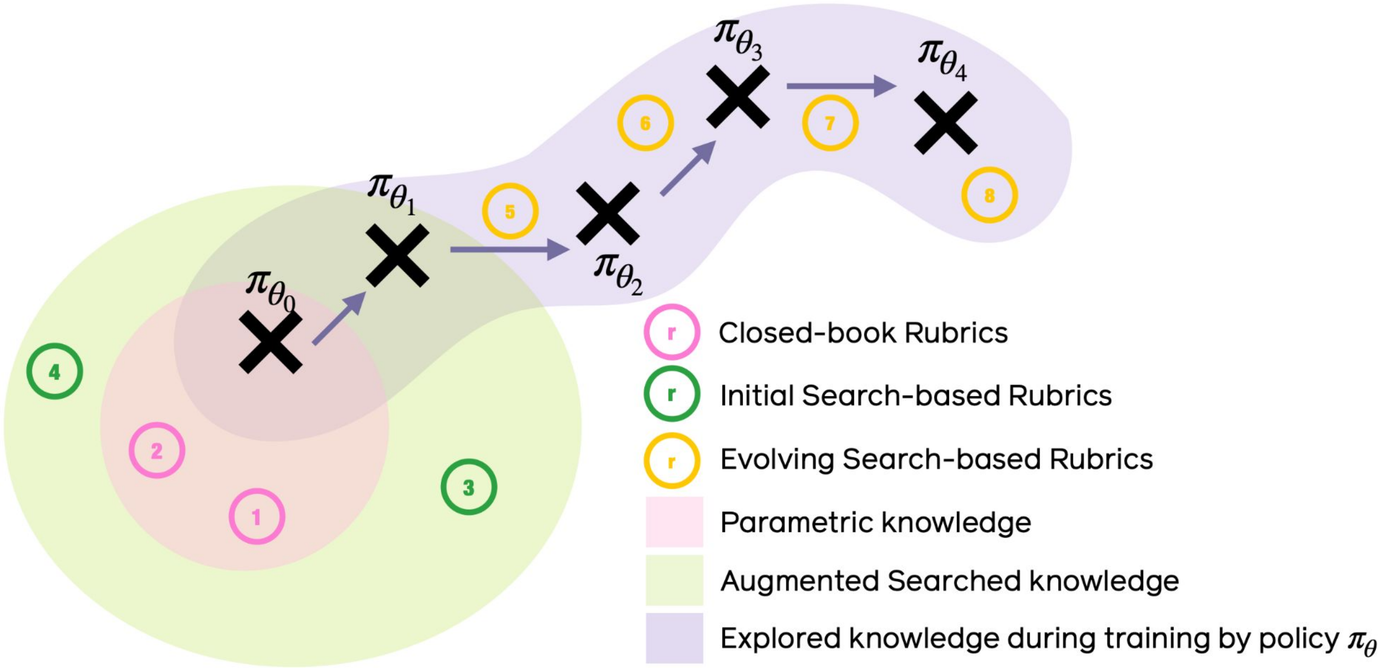
RL with Evolving Rubrics (RLER)

Maintain a rubric buffer.



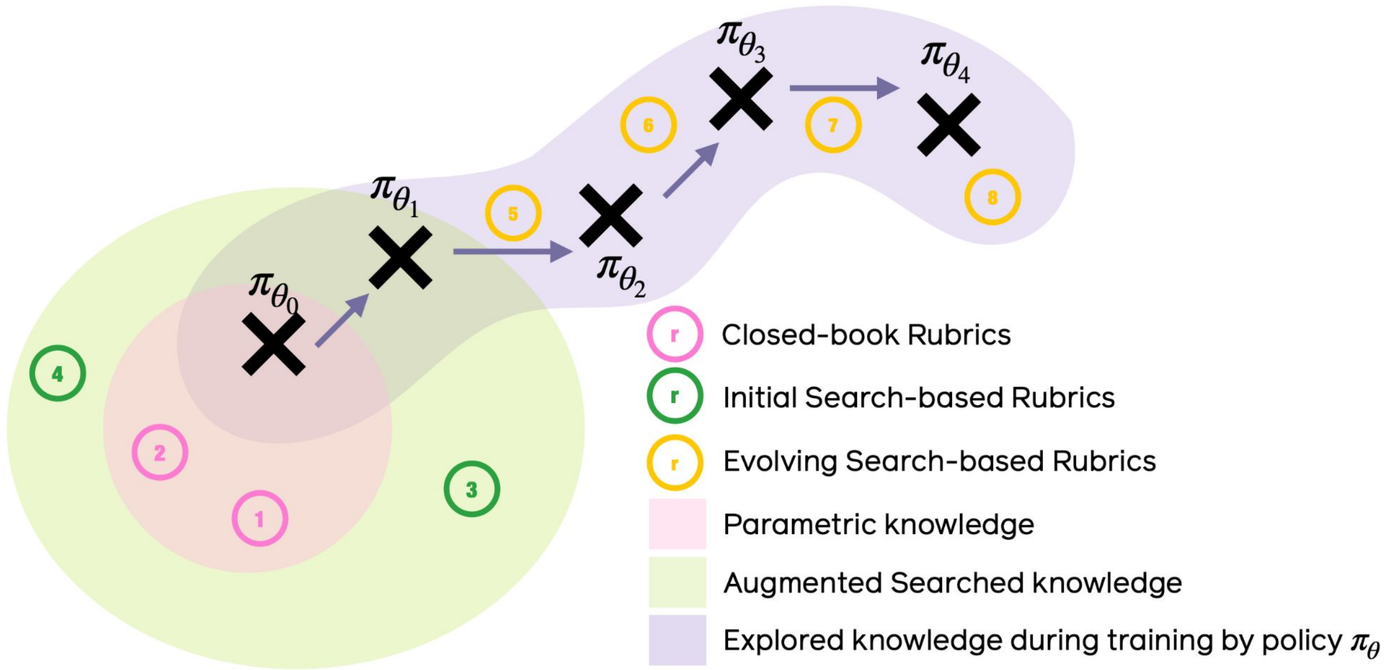
Rubrics co-evolve with the policy

- The knowledge coverage of the rubrics **co-evolves** with the policy's.



Rubrics co-evolve with the policy

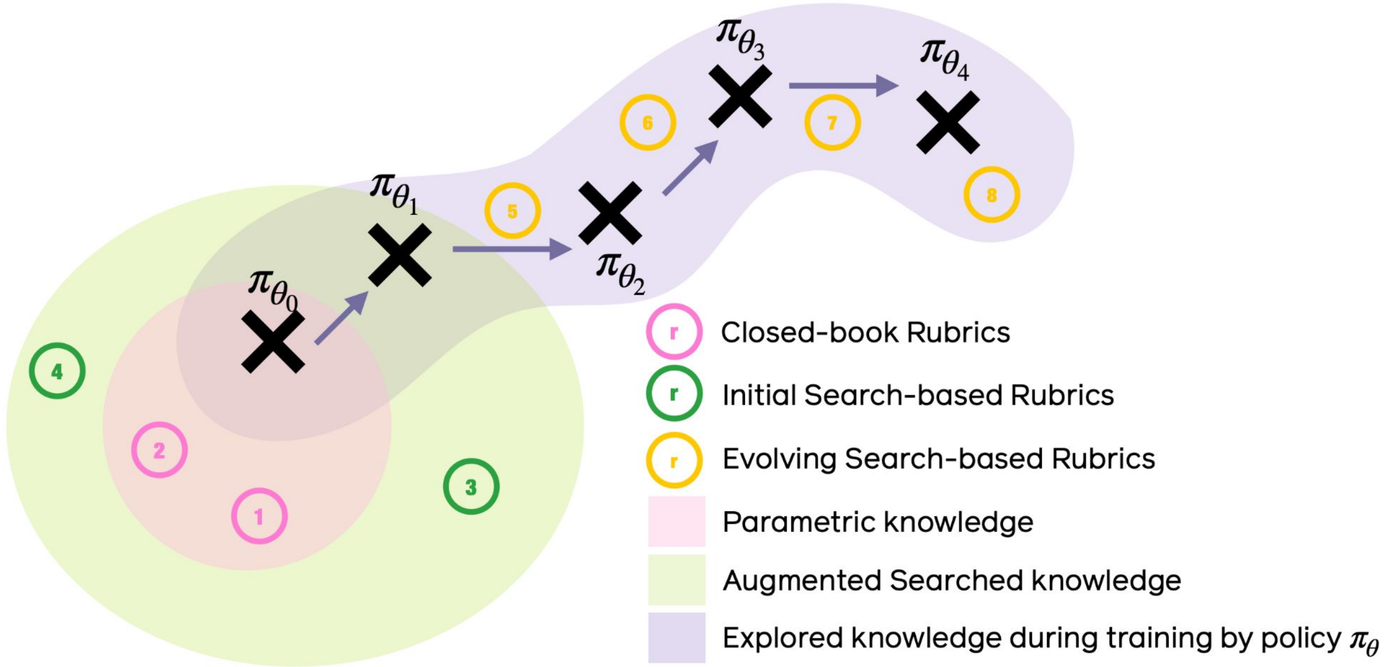
- The knowledge coverage of the rubrics **co-evolves** with the policy's.
- Rewards are adaptive to on-policy rollouts.



Rubrics co-evolve with the policy

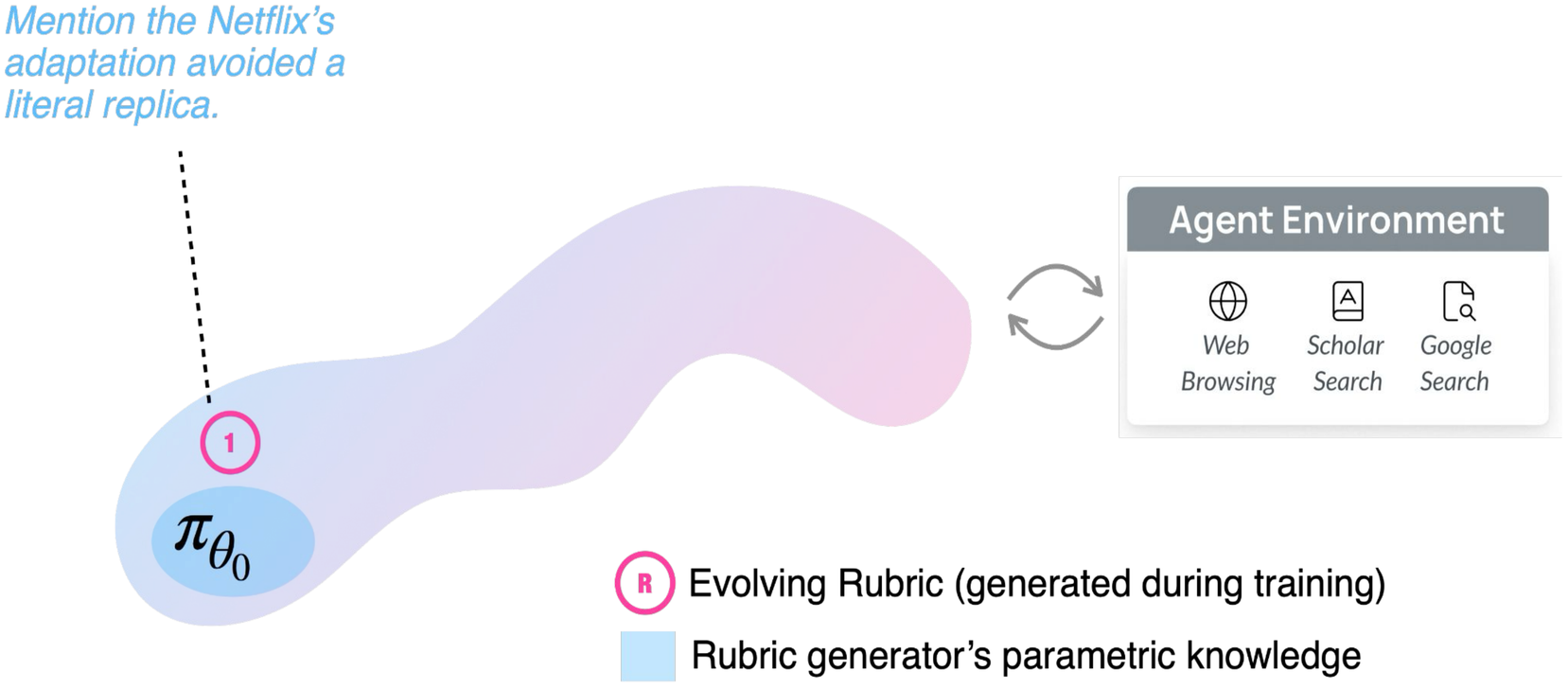
Weak-to-strong generalization by utilizing 2 types of privileged information:

- **Searched information** (breaking the limit of a closed environment)
- **Contrasting multiple rollouts** (comparing the rollouts is an easier task)



Rubrics co-evolve with the policy

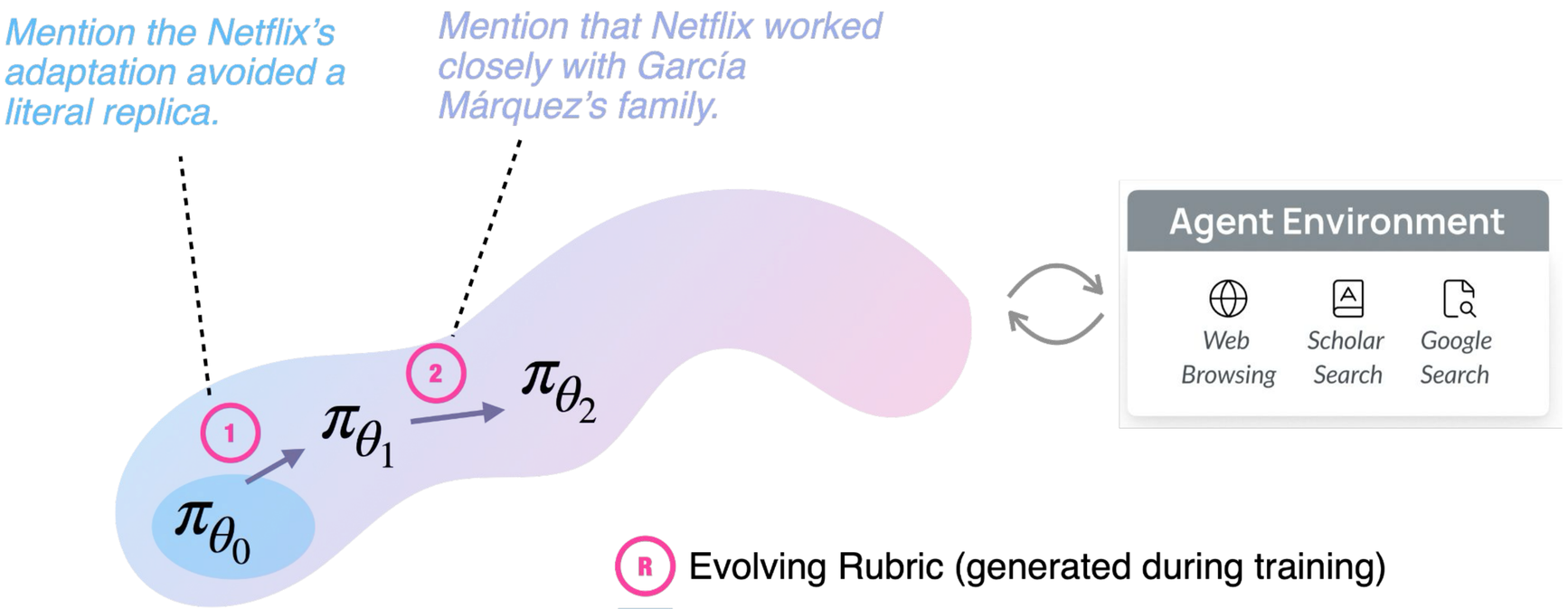
How did Netflix manage to successfully adapt One Hundred Years of Solitude, a notoriously difficult book to bring to the screen?



epoch=0

Rubrics co-evolve with the policy

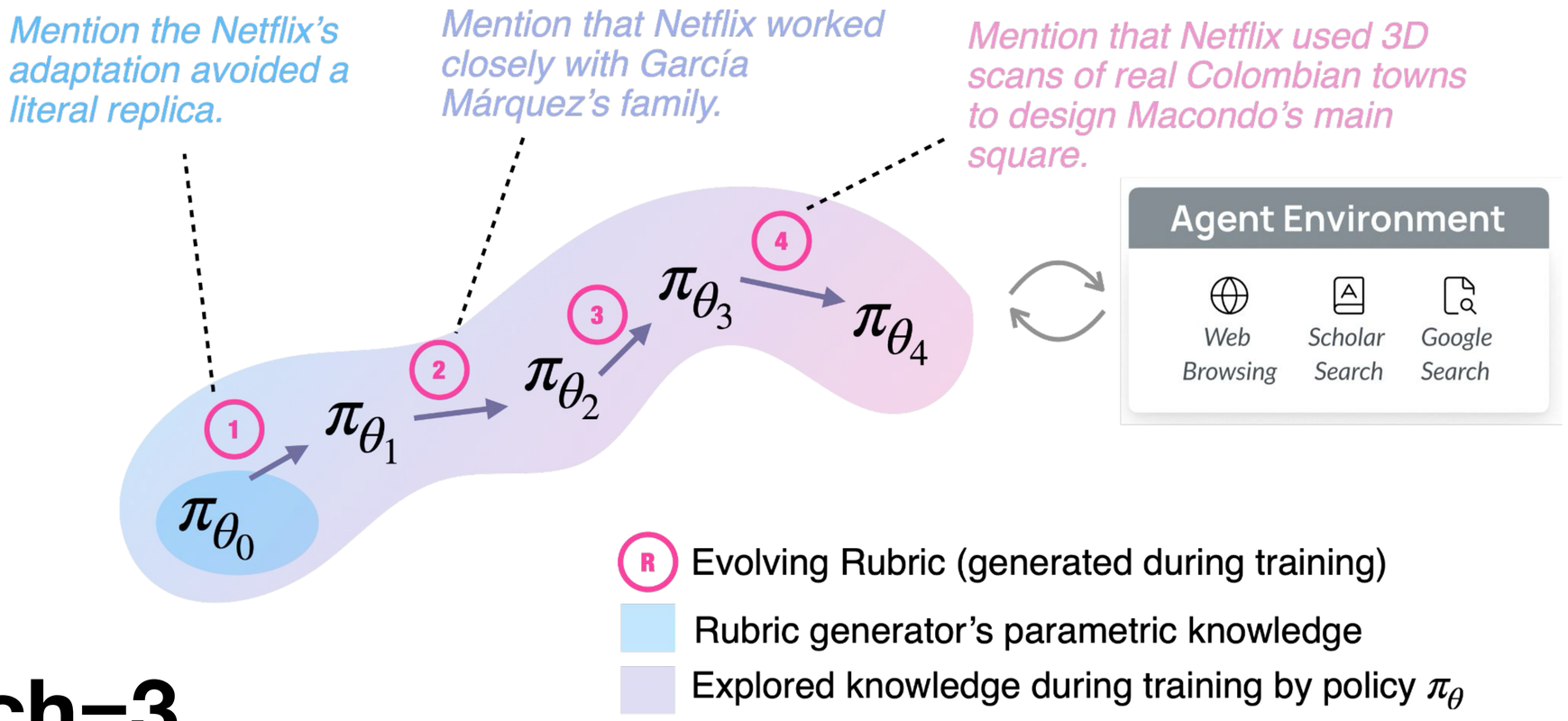
How did Netflix manage to successfully adapt One Hundred Years of Solitude, a notoriously difficult book to bring to the screen?



epoch=1

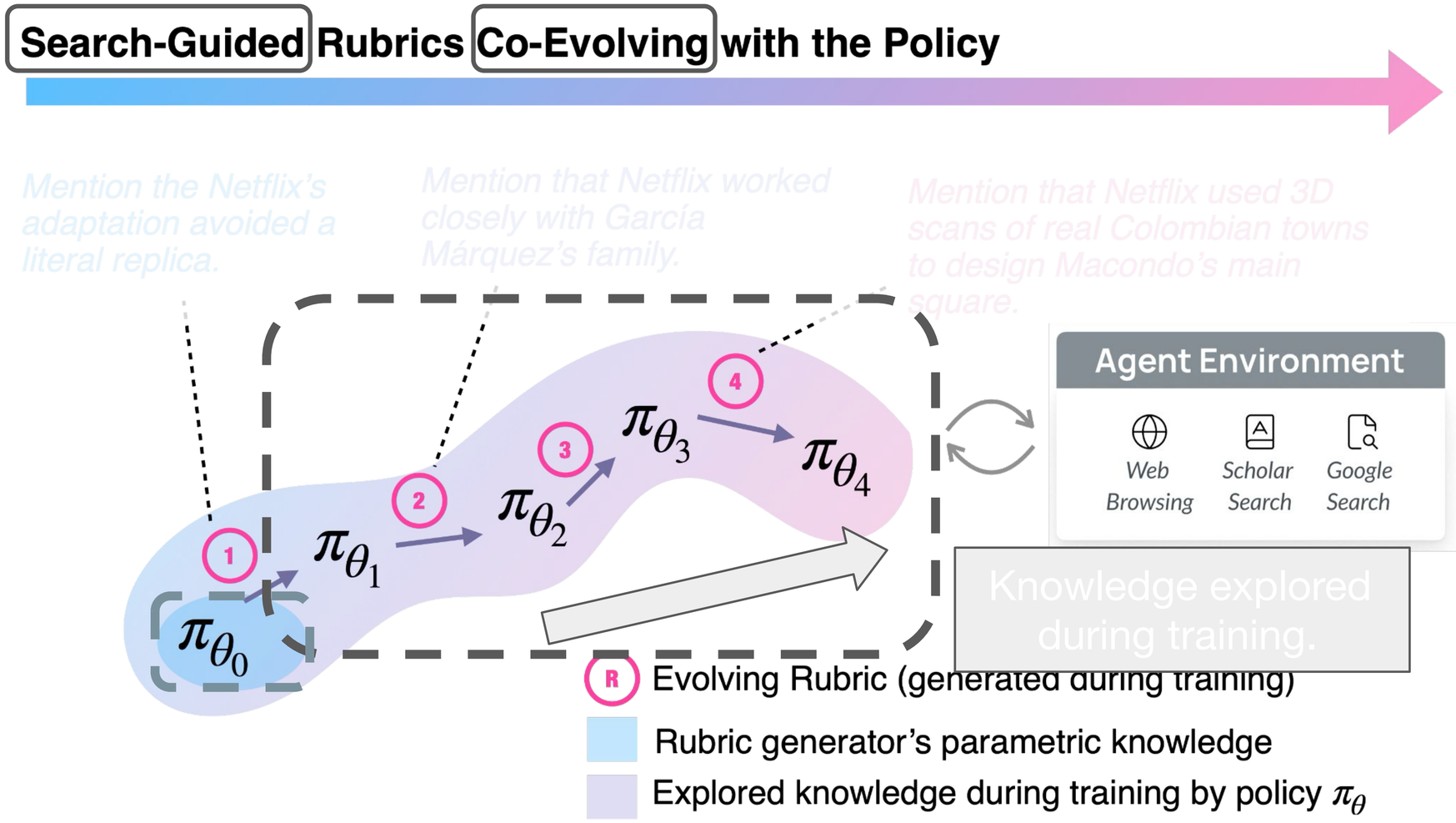
Rubrics co-evolve with the policy

How did Netflix manage to successfully adapt One Hundred Years of Solitude, a notoriously difficult book to bring to the screen?



epoch=3

DR Tulu: RL with Evolving Rubrics (RLER)



DR Tulu: Training

SFT with teacher trajectories

Prompt:

OpenScholar User Qs

SearchArena user Qs

Response:

GPT5

QwQ-32B

DR Tulu-SFT

DR Tulu: Training

github.com/rlresearch/dr-tulu

SFT with teacher trajectories

RLER

Prompt:

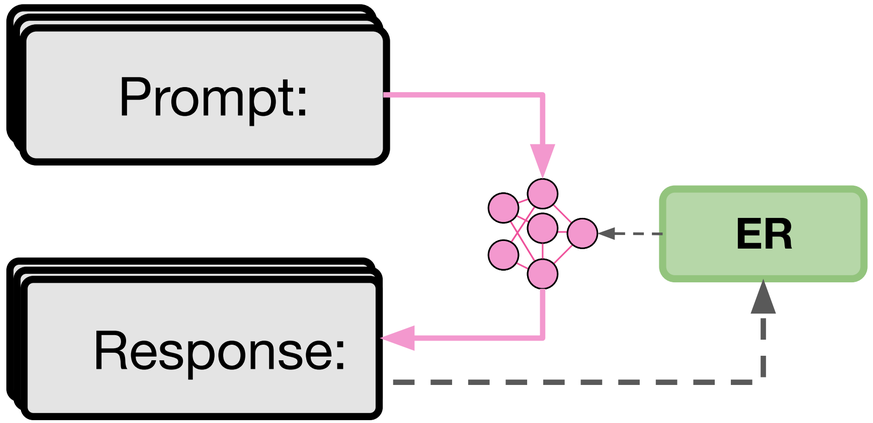
OpenScholar User Qs

SearchArena user Qs

Response:

GPT5

QwQ-32B

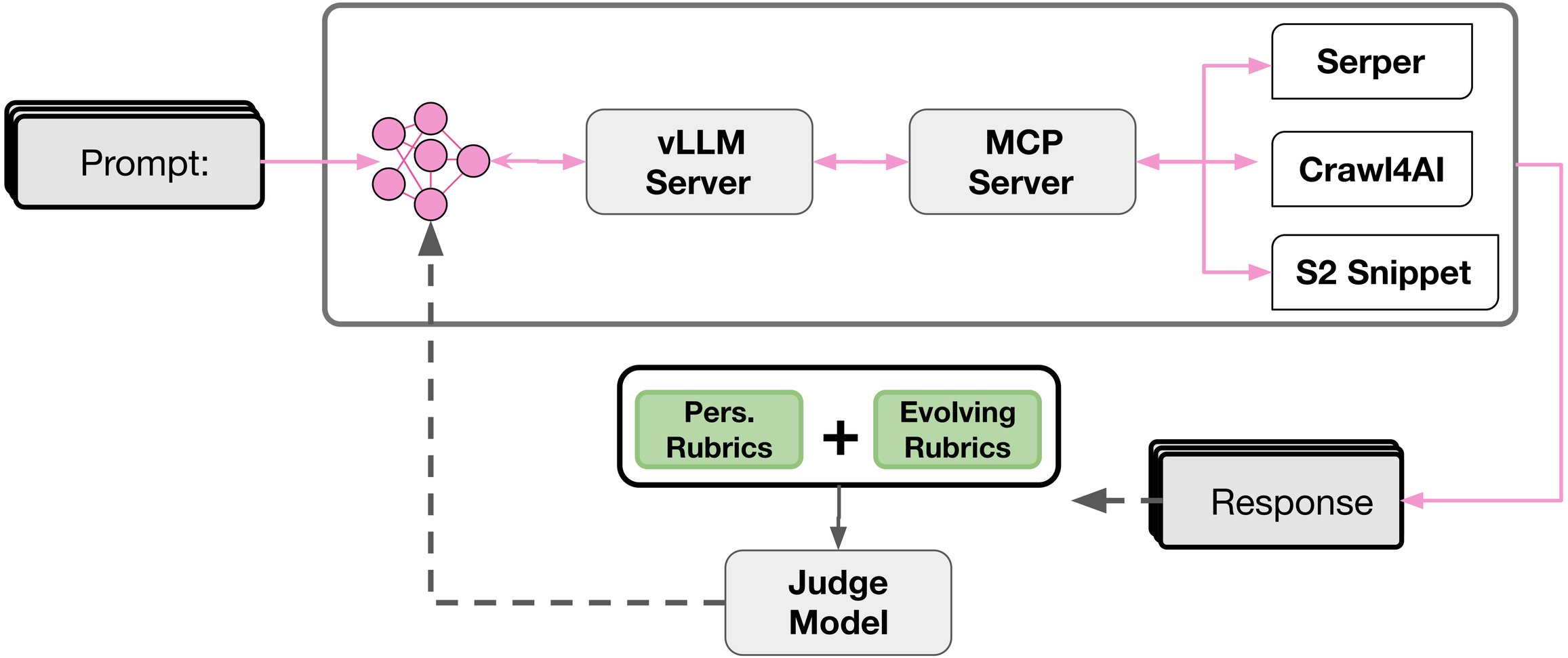


DR Tulu-SFT

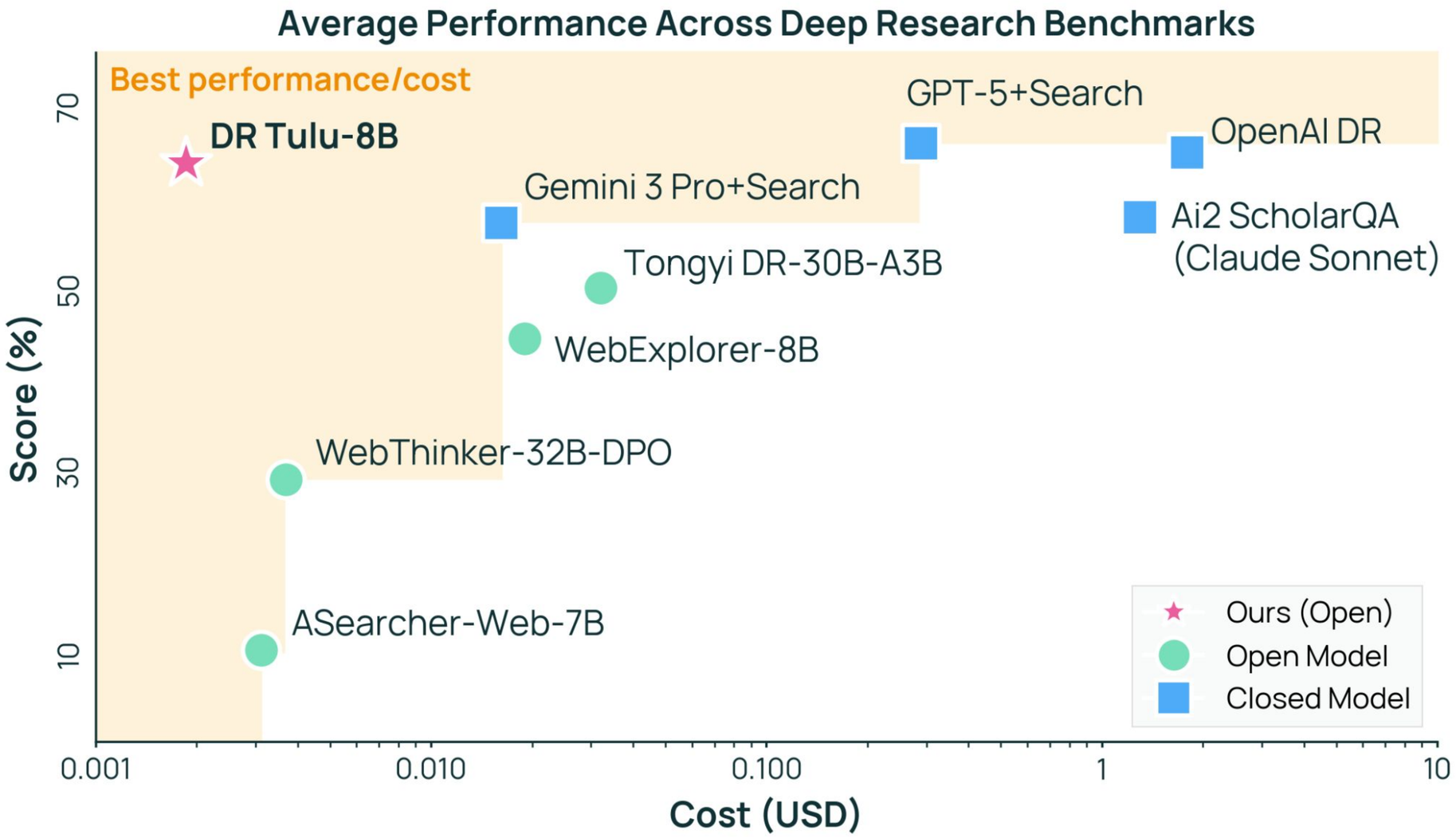
DR Tulu (RL)

DR Tulu: RL Training

github.com/rlresearch/dr-tulu



DR Tulu is at the Pareto frontier!



DR Tulu shows effectiveness on a real expert task

User

What is the pathomechanism of variant NM_001127222.2(CACNA1A):c.4174G>A?

DR Tulu

Long-form Report with Citations

NM_001127222.2(CACNA1A):c.4174G>A corresponds to p.Val1392Met in the CACNA1A α1A subunit and is associated with familial hemiplegic migraine type 1 (FHM1) in clinical databases and registries [1]

Multiple experimental studies show that missense mutations in CACNA1A that cause FHM1 confer gain-of-function on Cav2.1 channels, increasing Ca2+ influx through single channels and/or altering channel behavior at physiological potentials; knock-in mouse models replicate synaptic gain-of-function effects [2]

Reviews similarly categorize Cav2.1 gain-of-function mutations as the cause of FHM, while loss-of-function mutations lead to episodic ataxia [3]

The pathomechanism of NM_001127222.2(CACNA1A):c.4174G>A (p.Val1392Met) is most consistent with a toxic gain-of-function of Cav2.1 channels, aligning with FHM1-associated missense variants. However, current data better support a gain-of-function mechanism for FHM1-associated missense variants and a loss-of-function/null mechanism for EA2-associated variants [3]

Sources

[1] Clinvar, RCV000415108

[2] Insights into migraine mechanisms and CaV2.1 calcium channel function, pmc.ncbi.nlm.nih.gov/articles/PMC2901975/

[3] A Review of the CACNA Gene Family: Its Role in Neurological Disorders https://doi.org/10.3390/diseases12050090

GeneticDiseasesQA (GDQA)

47 questions derived from expert-curated information about 24 disease-causing genetic variants

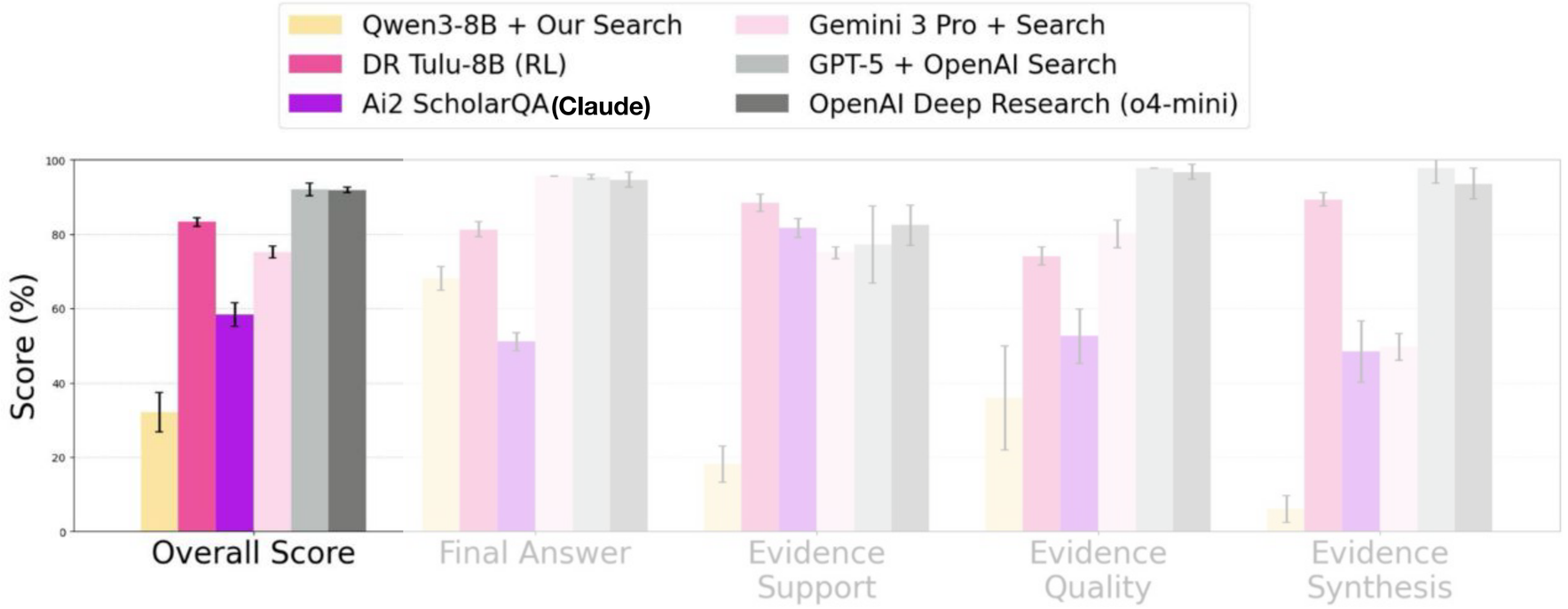
Seattle Children's
HOSPITAL • RESEARCH • FOUNDATION

N=1

Evaluation criterion:

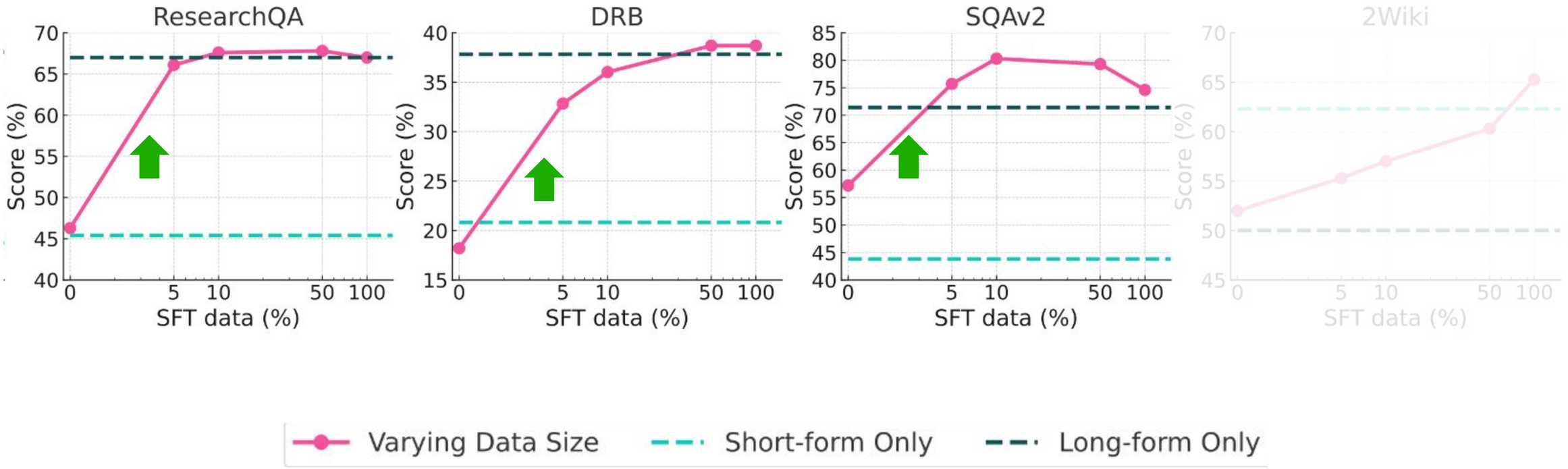
- Final answer
- Evidence Support
- Evidence Correctness
- Evidence Synthesis

DR Tulu shows effectiveness on a real expert task

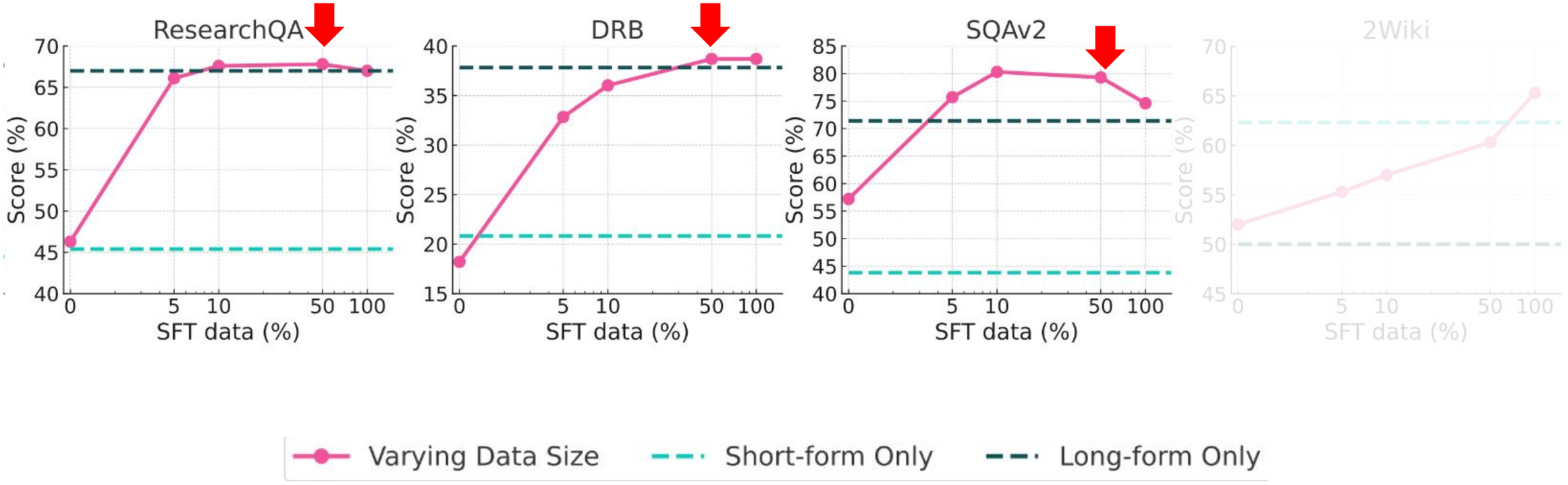


Overall DR Tulu outperformed Ai2 ScholarQA w/ Claude Sonnet & Gemini 3

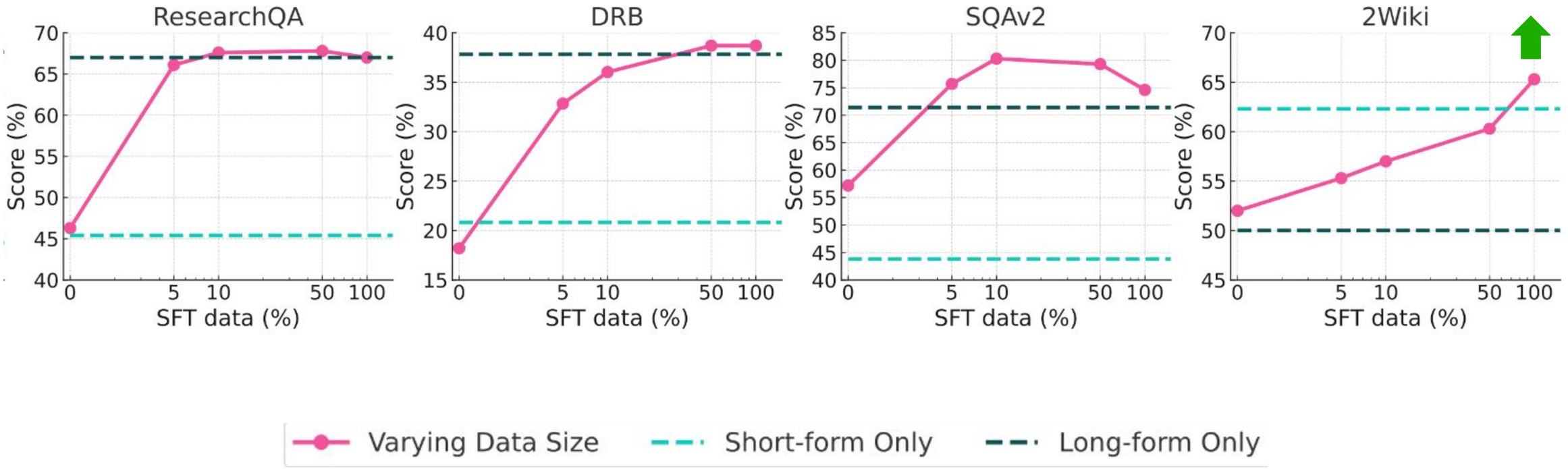
Finding #1: Small-scale SFT improves OE DR a lots



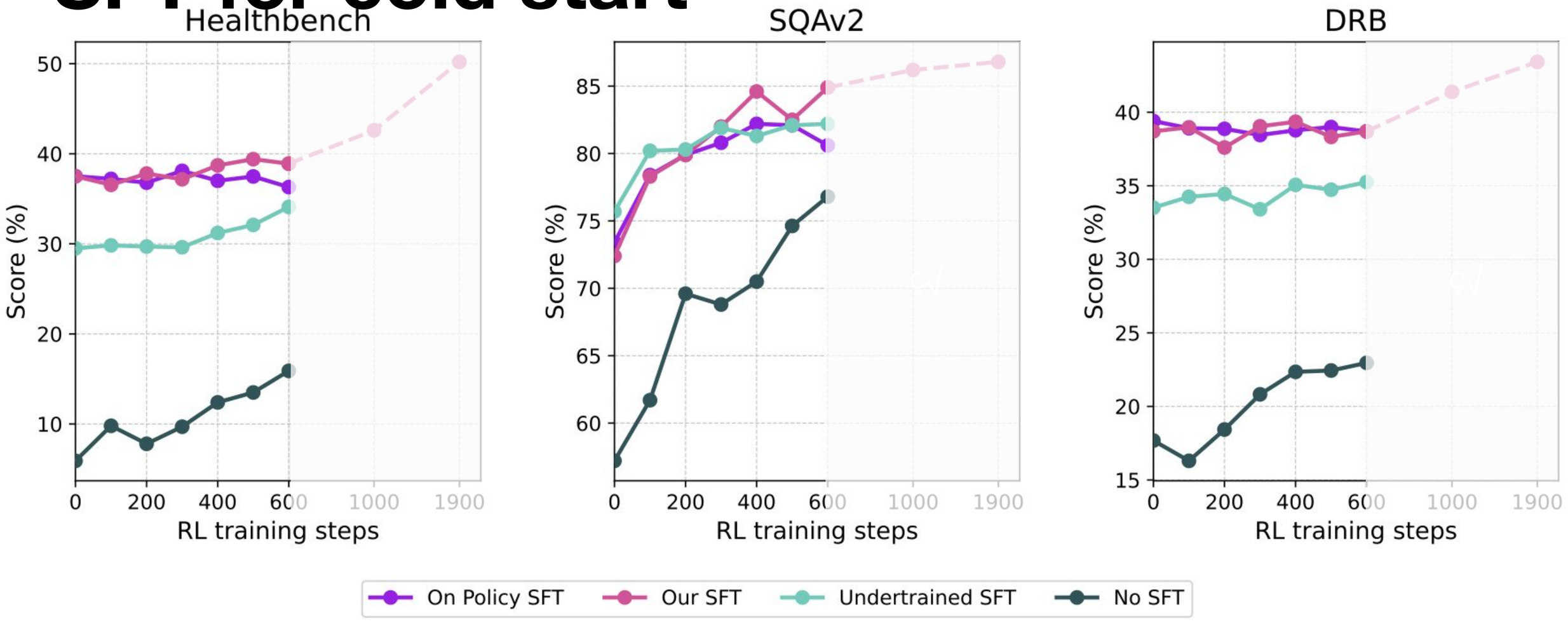
Finding #2: SFT only can’t improve OE DR further



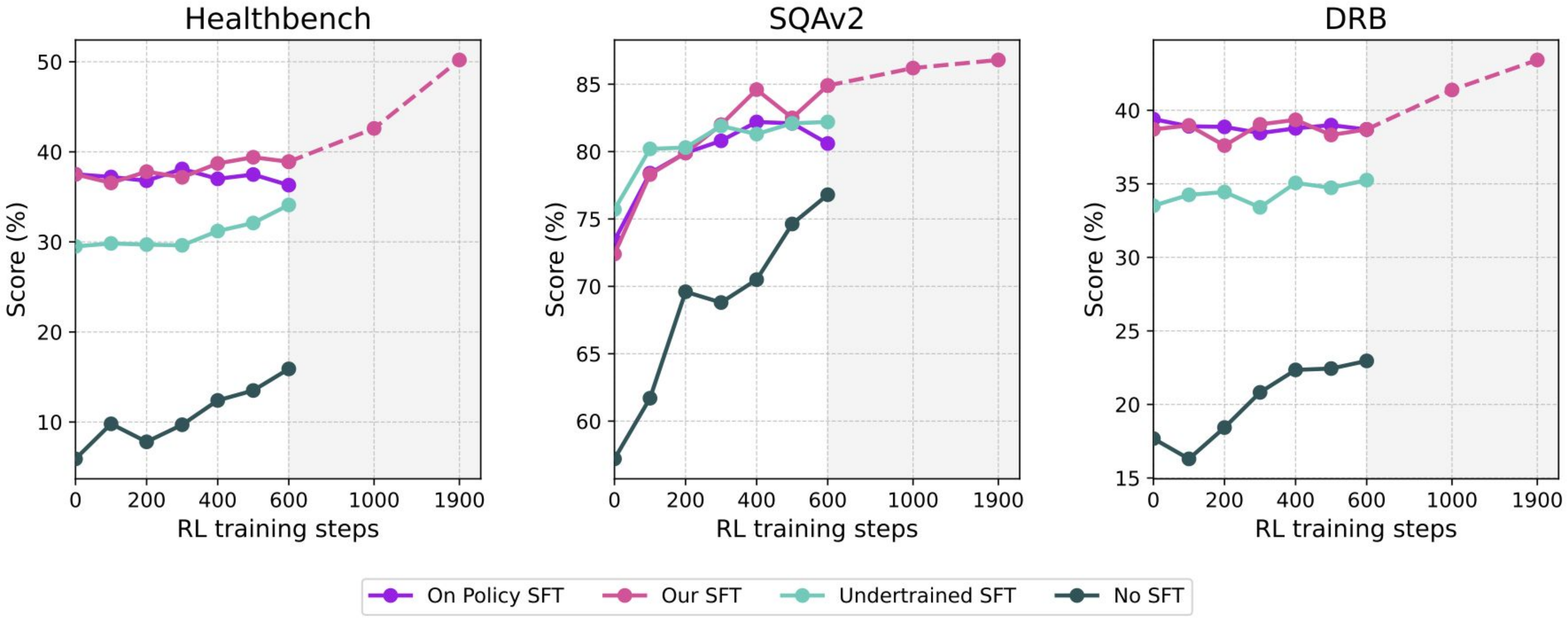
Finding #2: SFT only can’t improve OE DR further



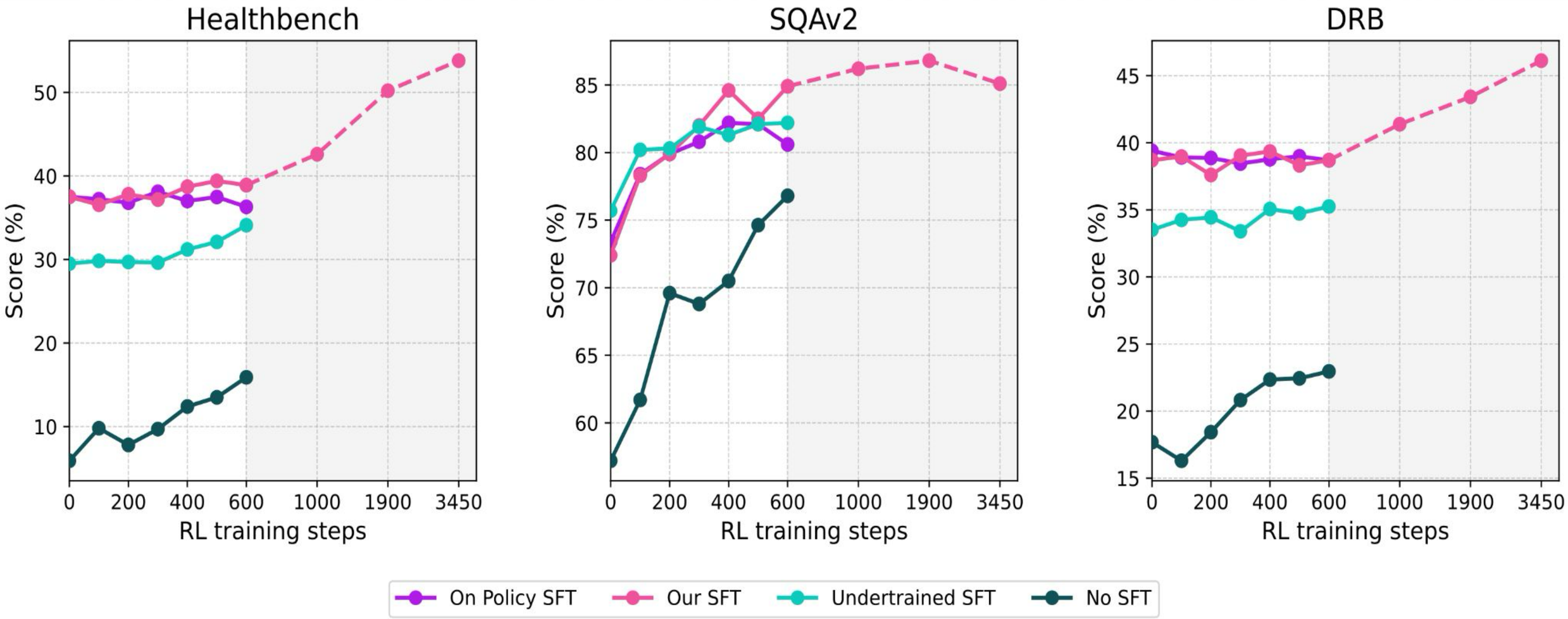
Finding #3: Importance of SFT for cold start



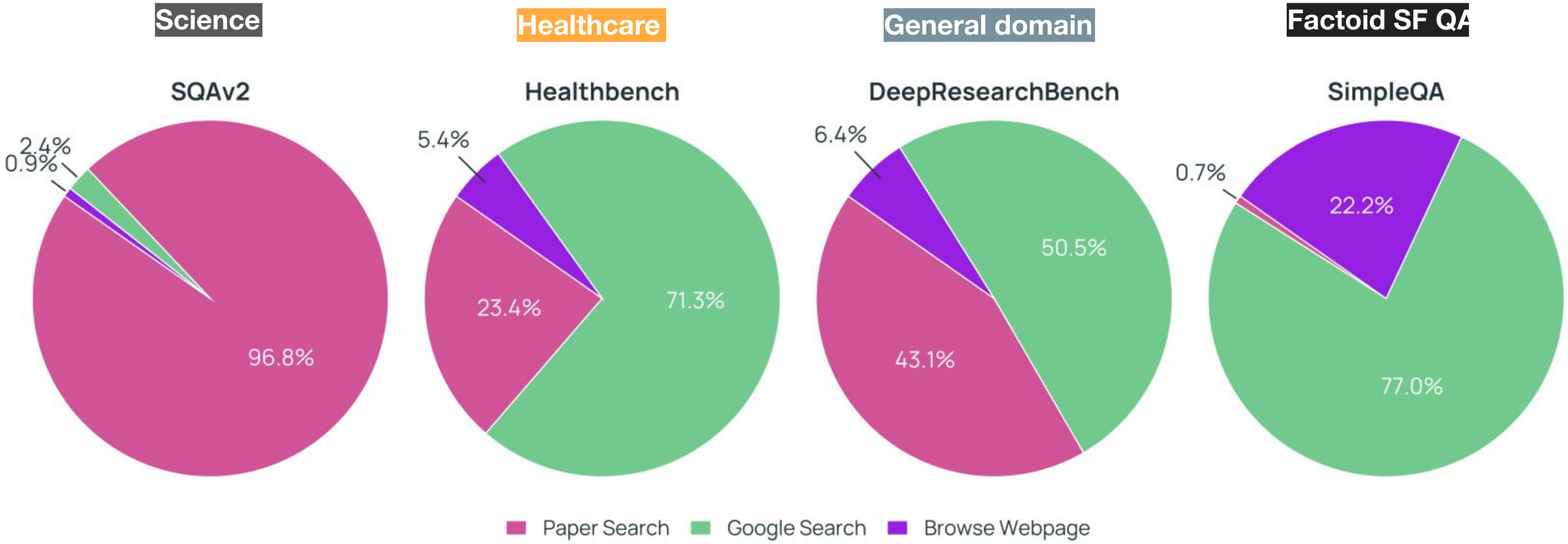
Finding #4: Longer RL was key



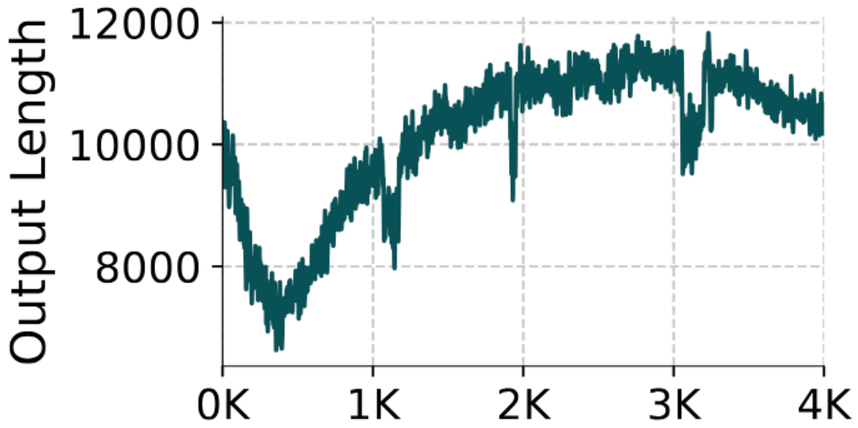
Finding #4: Longer-er RL is key

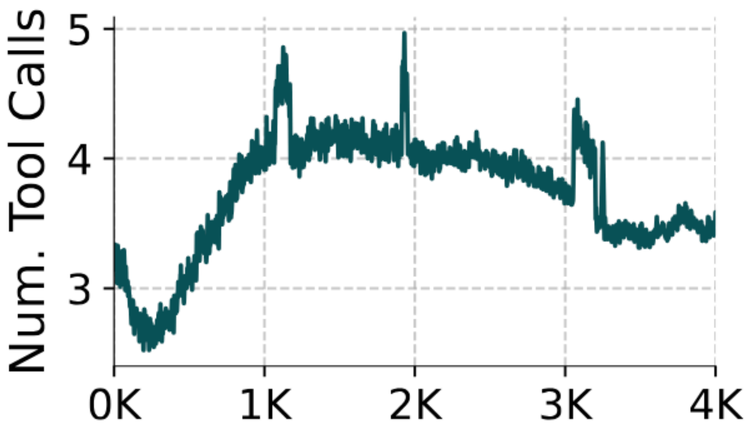


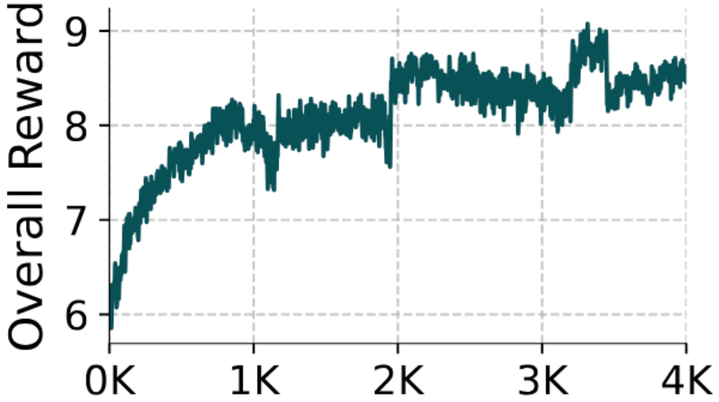
Finding #5: DR Tulu learns to use relevant tools



Finding #6: RL is robust to tool failures!



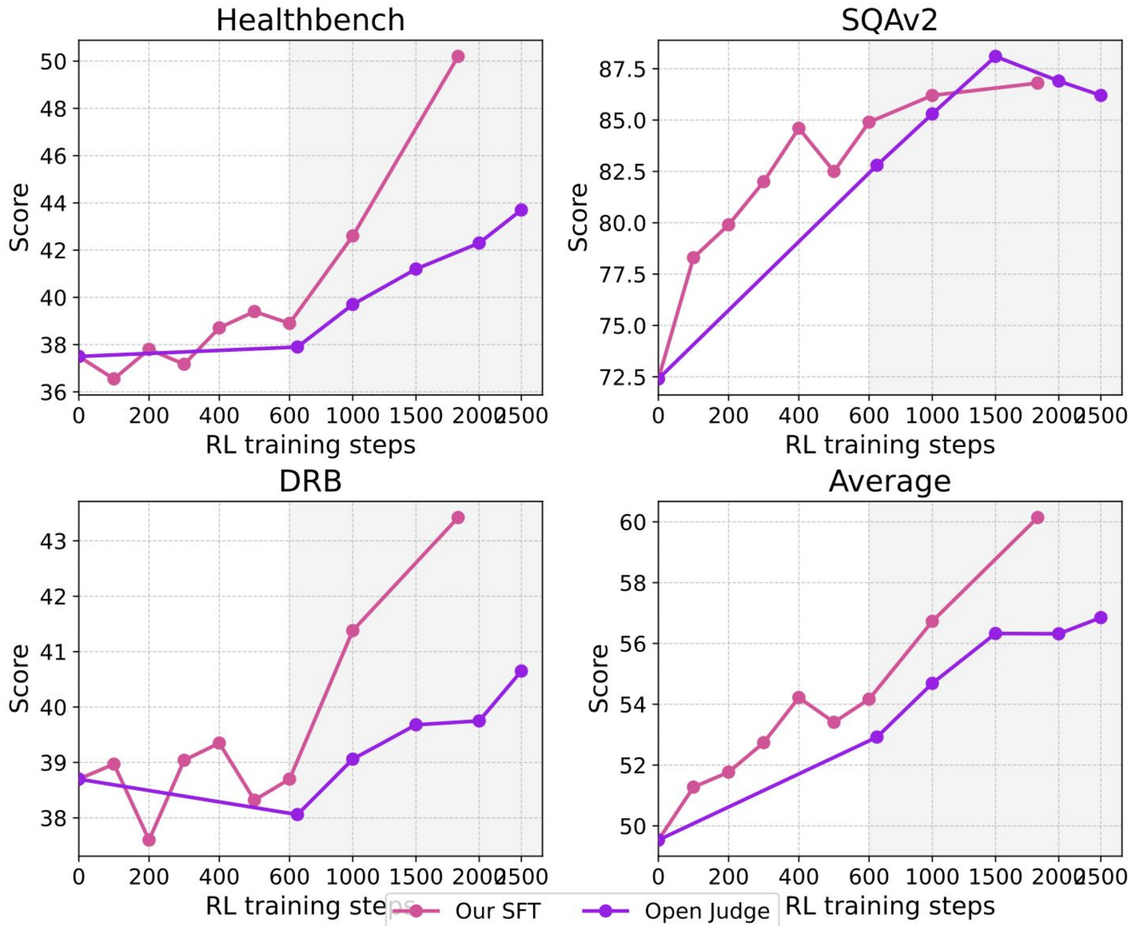




Finding #7: Ablation on the rubric/judge model

Pink: GPT4.1 for rubric generation.

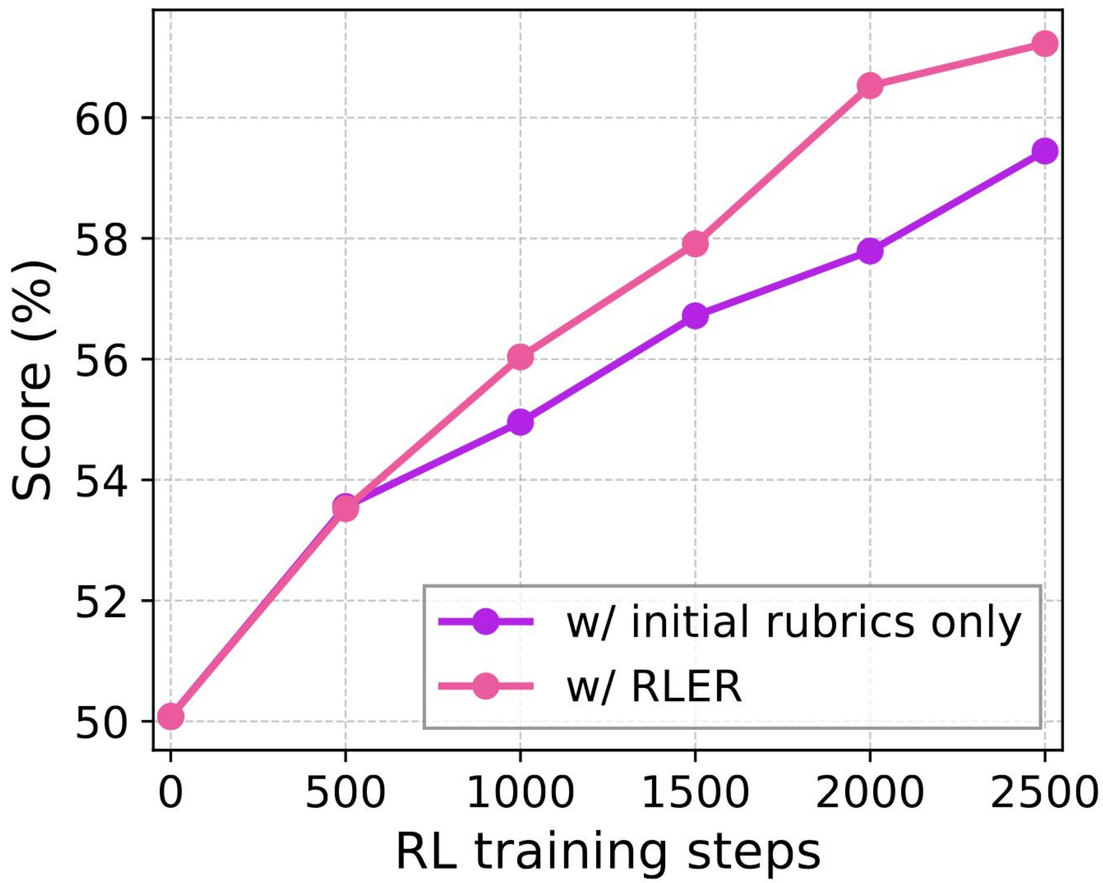
Purple: Qwen3-8B for rubric generation.



Finding #8: Ablation on ER

Pink: search-based persistent rubrics + evolving rubrics

Purple: search-based persistent rubrics only



Finding #9: Ablation on search-based rubrics

- Due to compute constraint, we ran this ablation for only 500 steps.

	SQAv2	Health	Research	DRB	Avg.
DR Tulu (SFTv0.1)	73.4	37.5	68.6	39.4	54.7
General rubrics	80.6	36.0	65.0	37.5	54.8
Closed-book rubrics	83.2	34.8	66.6	37.6	55.6
Initial search-based rubrics	82.8	37.9	66.9	39.3	56.7

Takeaways

- **Evolving Rubrics** allow us to dynamically improve difficulty and quality of rubrics along with the model.
- **Deep Research** poses lots of interesting challenges beyond ‘typical’ RL tasks (‘unverifiable’).
- **RL training can take a long time!**

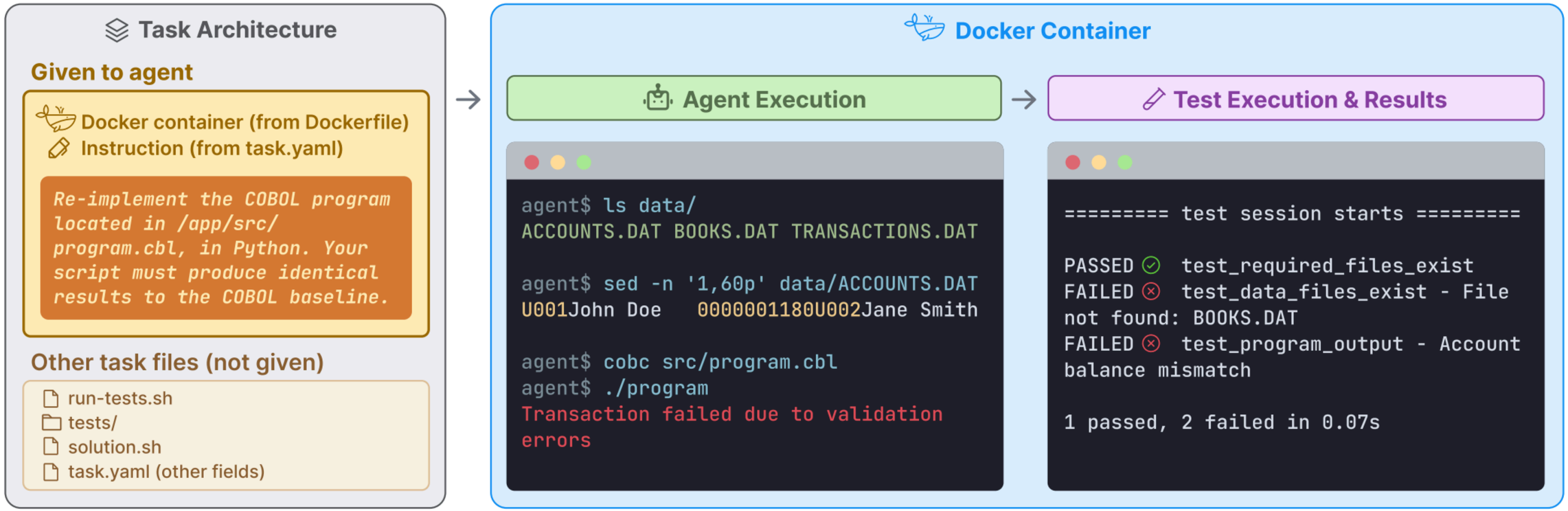
TMAX:
A simple recipe for terminal agents

Hamish Ivison[*,α,ω] **Junjie Oscar Yin**[*,ω]
Rulin Shao[α,ω] **Teng Xiao**[α] **Nathan Lambert**[α] **Hannaneh Hajishirzi**[ω]

[α]Allen Institute for AI [ω]University of Washington
{hamishiv,osey}@cs.washington.edu

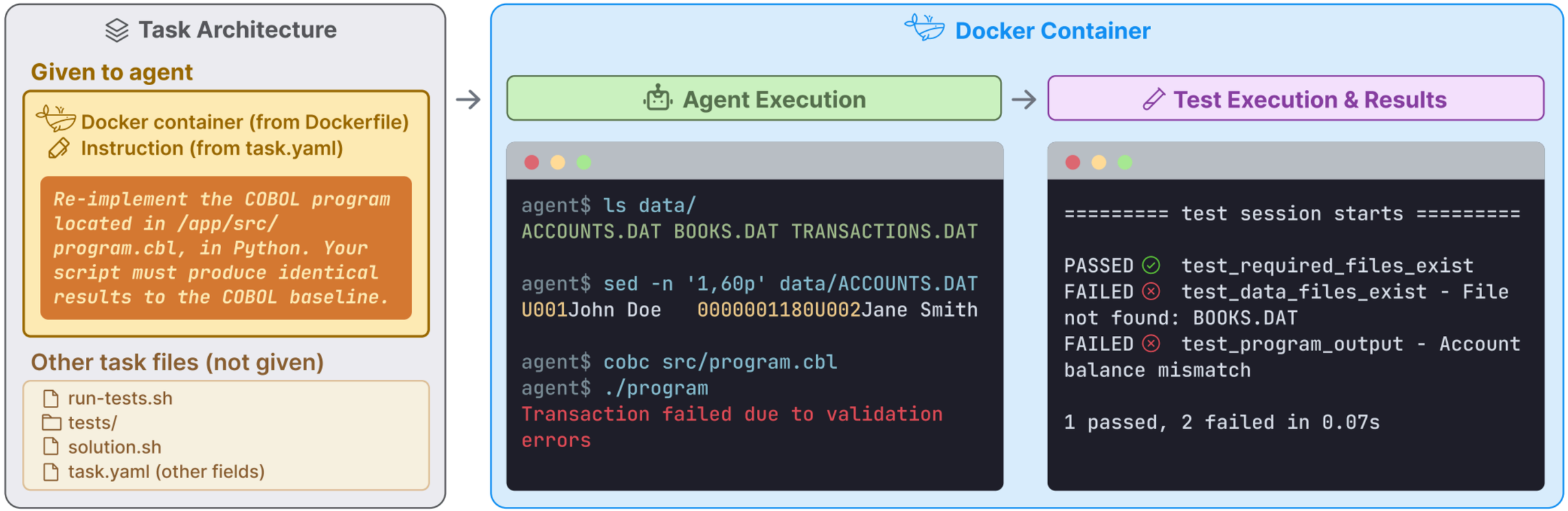


Terminal agent tasks



Merrill, Mike A. et al. “Terminal-Bench: Benchmarking Agents on Hard, Realistic Tasks in Command Line Interfaces.” *arXiv abs*/2601.11868 (2026).

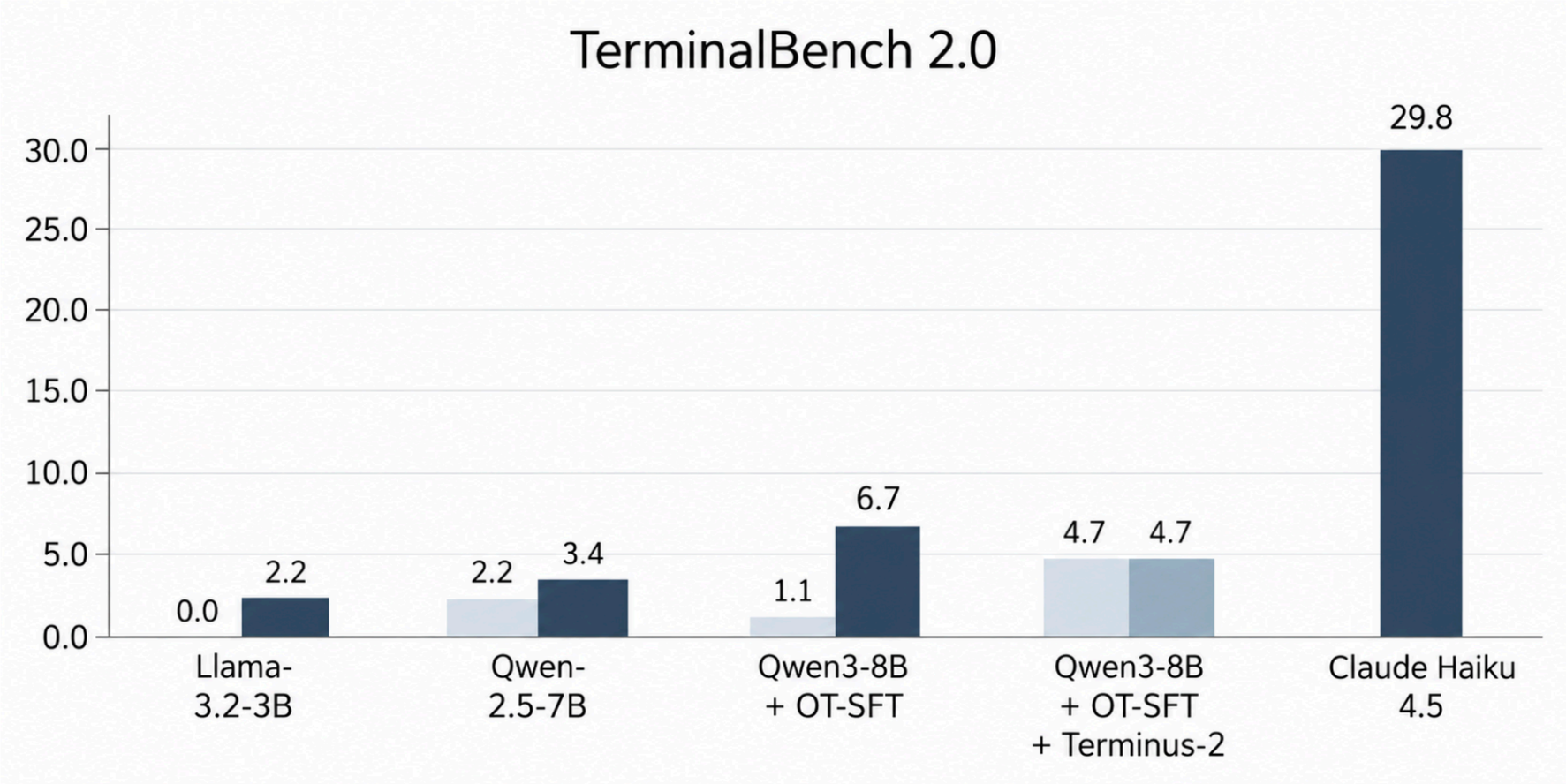
Terminal agent tasks



Merrill, Mike A. et al. “Terminal-Bench: Benchmarking Agents on Hard, Realistic Tasks in Command Line Interfaces.” *arXiv abs*/2601.11868 (2026).

Require *interaction* over many turns!

Open terminal agents are lacking



Gandhi, Kanishk et al. “Endless Terminals: Scaling RL Environments for Terminal Agents.” *arXiv abs*/2601.16443 (2026).

Open terminal agents are lacking



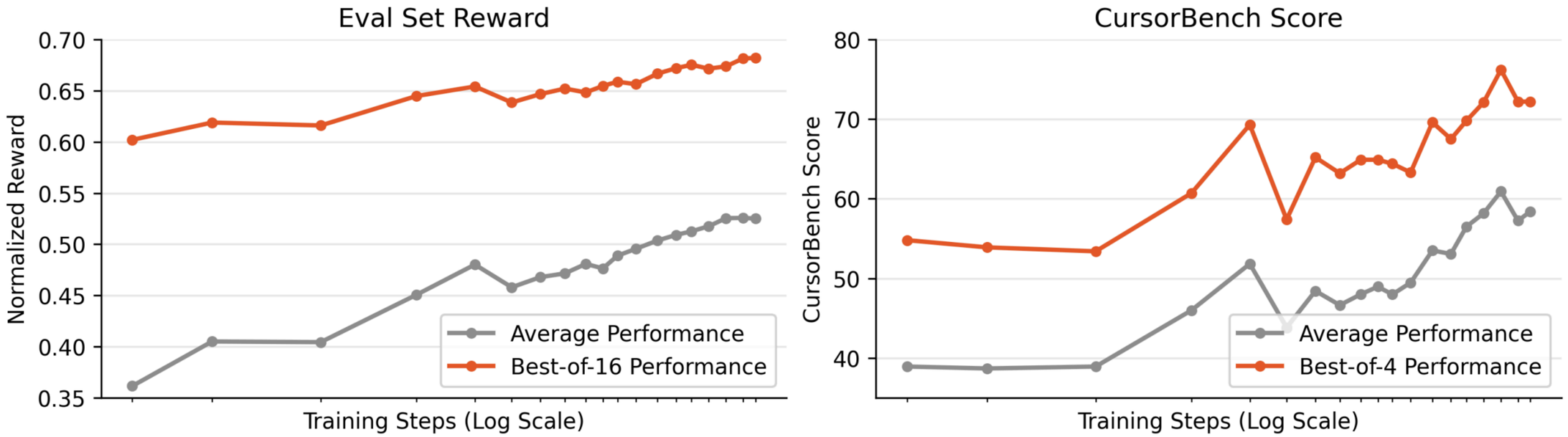
OpenThoughts Agent

Reinforcement Learning

RL is a key part of the next phase of the OpenThoughts-Agent project. We have built an RL training stack by integrating SkyRL with Harbor and stress-tested it with a large number of RL runs on TerminalBench-style tasks. Conducting RL on our SFT-only model using our RL data, OpenThoughts-Agent-v1-RL, we get a small improvement on our development set of around ~2% and an improvement of 1% on SWE-Bench verified.

OpenThoughts-Agent Team. "OpenThoughts-Agent." OpenThoughts (2025).

We know RL is crucial to performance!



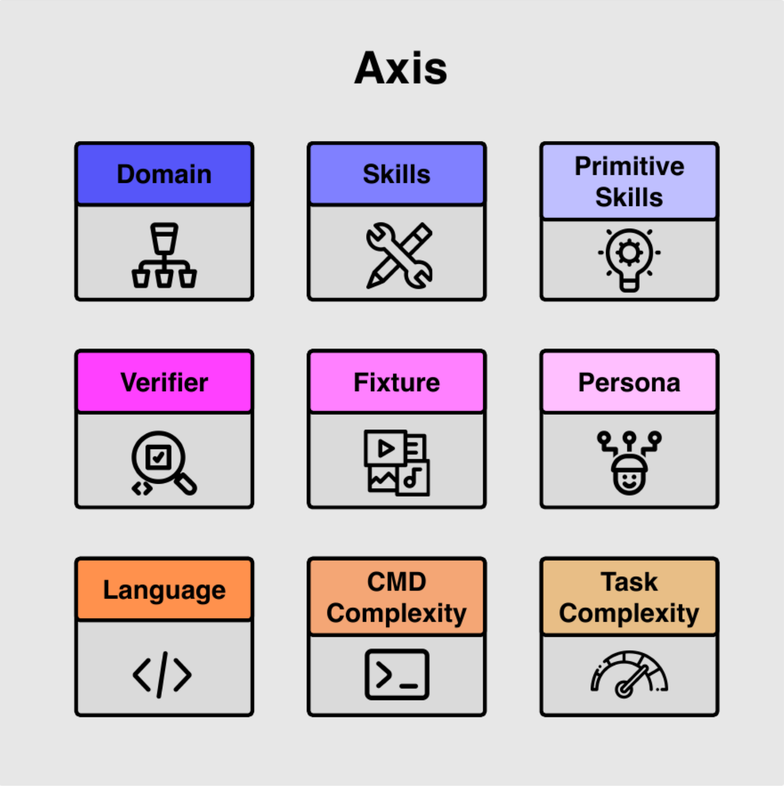
Chan, Aaron et al. “Composer 2 Technical Report.” *arXiv abs*/2603.24477 (2026).

Fixing this: TMax

1. Open data and recipe.
2. Significant improvements from the RL process alone.
3. Smaller-scale models (accessible on an academic budget)

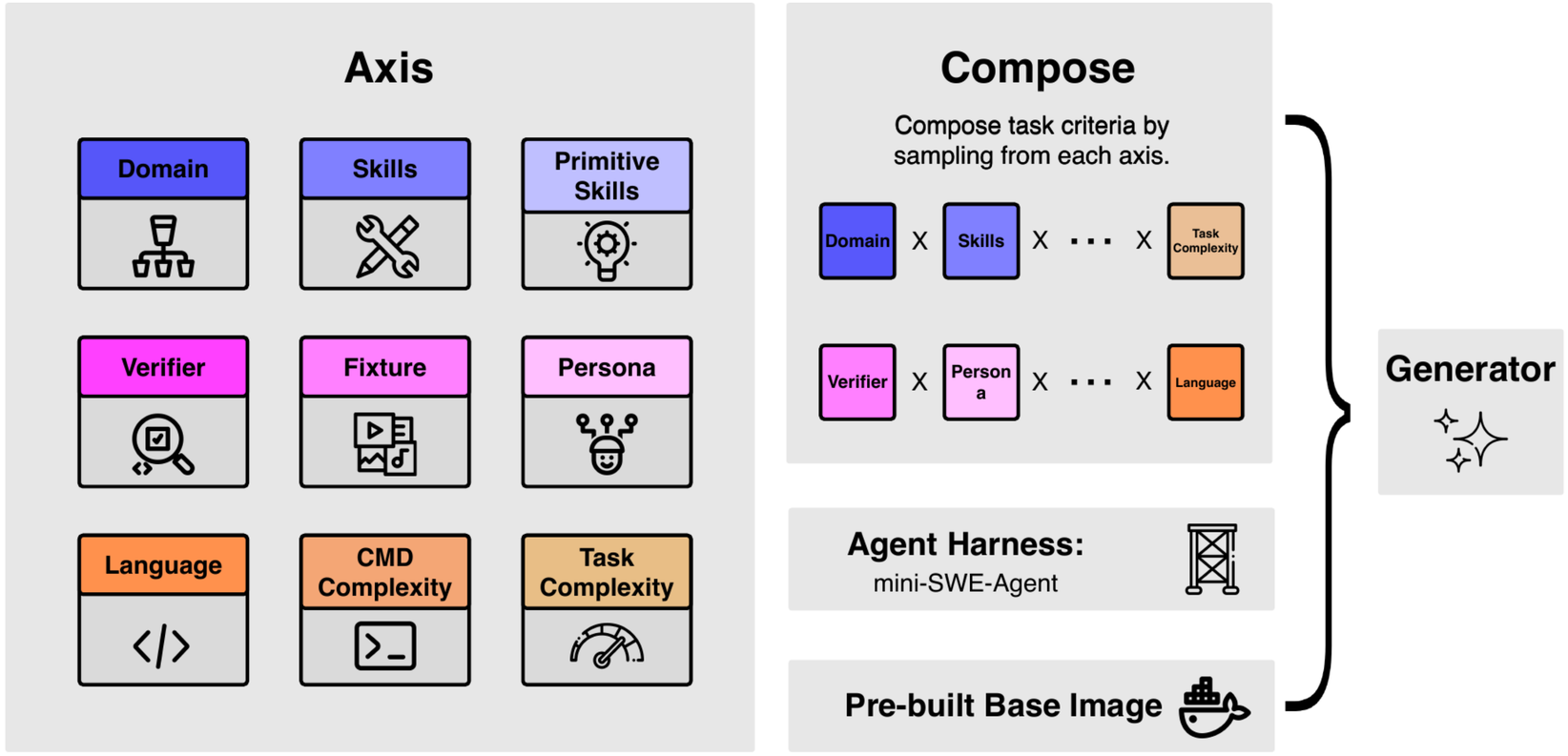


TMax: data generation

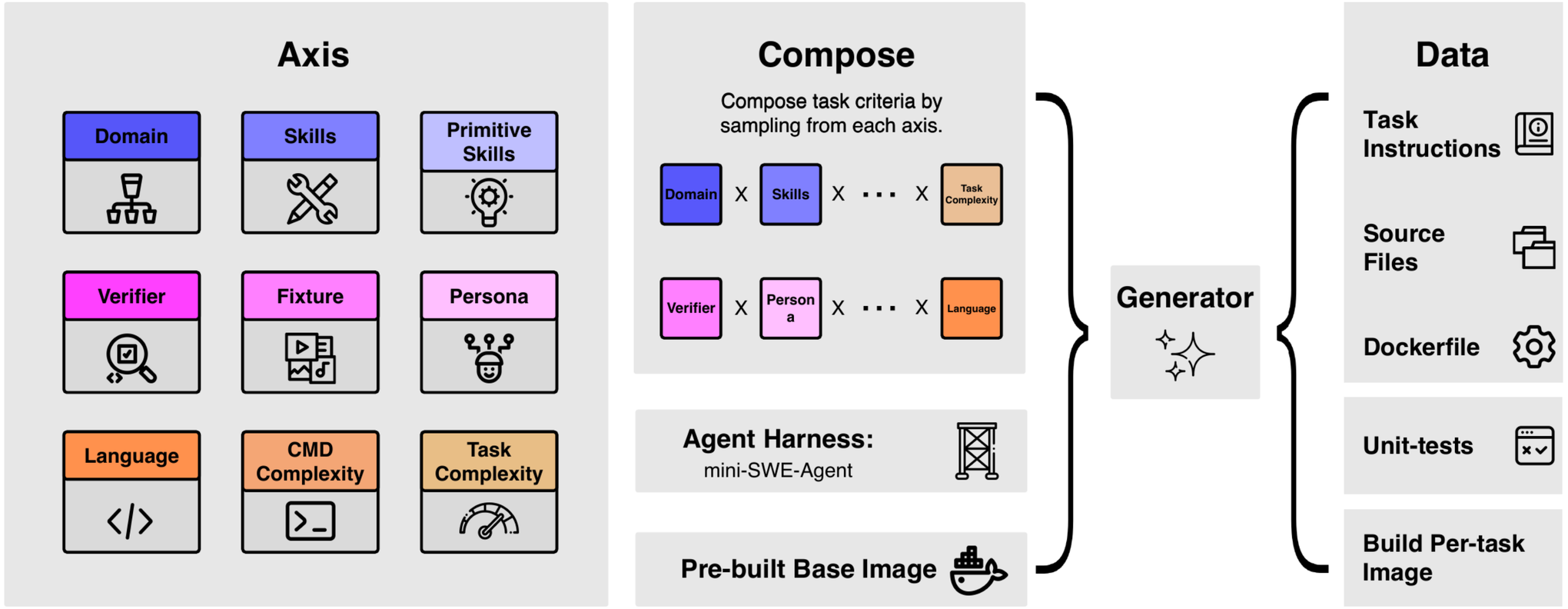


Axis	Source	Cardinality	Values
Domain	Pi et al. (2026)	9	security, software_engineering, file_operations, data_querying, data_science, debugging, scientific_computing, data_processing, system_administration
Skill type	Pi et al. (2026)	4–7 / domain	e.g. Algorithmic, Systems, Data Processing, Web Security, Testing, Mathematical, Multi-Language
		...	
Task complexity	Ours	4	short, moderate, complex, intricate (30–60 commands)
Command complexity	Ours	3	bash-only; bash + code; bash + code + system services
Fixture	Ours	7	text_only, image, audio, video, stripped_binary, vendored_package, multi_service_compose
Verifier	Ours	5	exact_text, metric_threshold, adversarial_corpus, fuzz_equivalence, multi_protocol

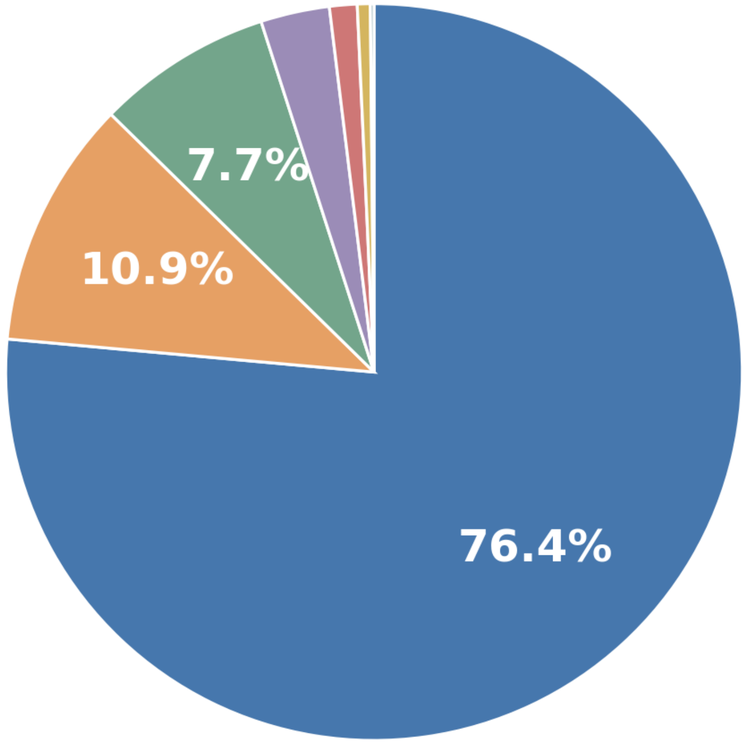
TMax: data generation



TMax: data generation

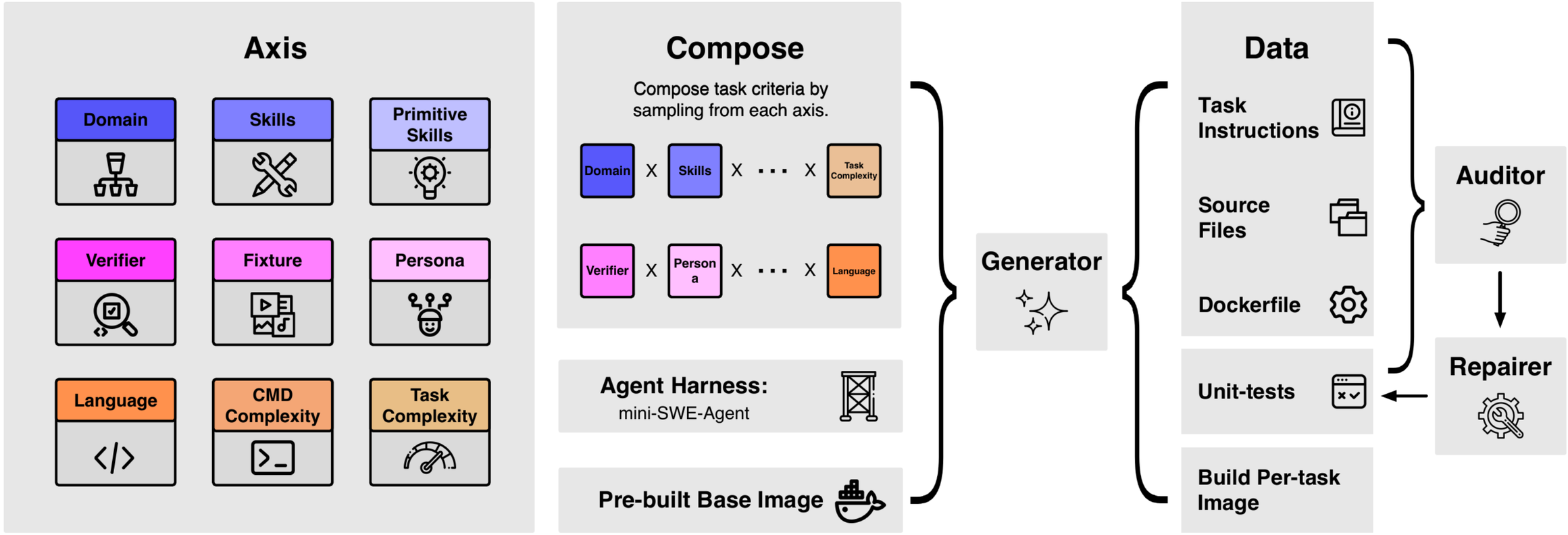


Single-pass generation: many failures!

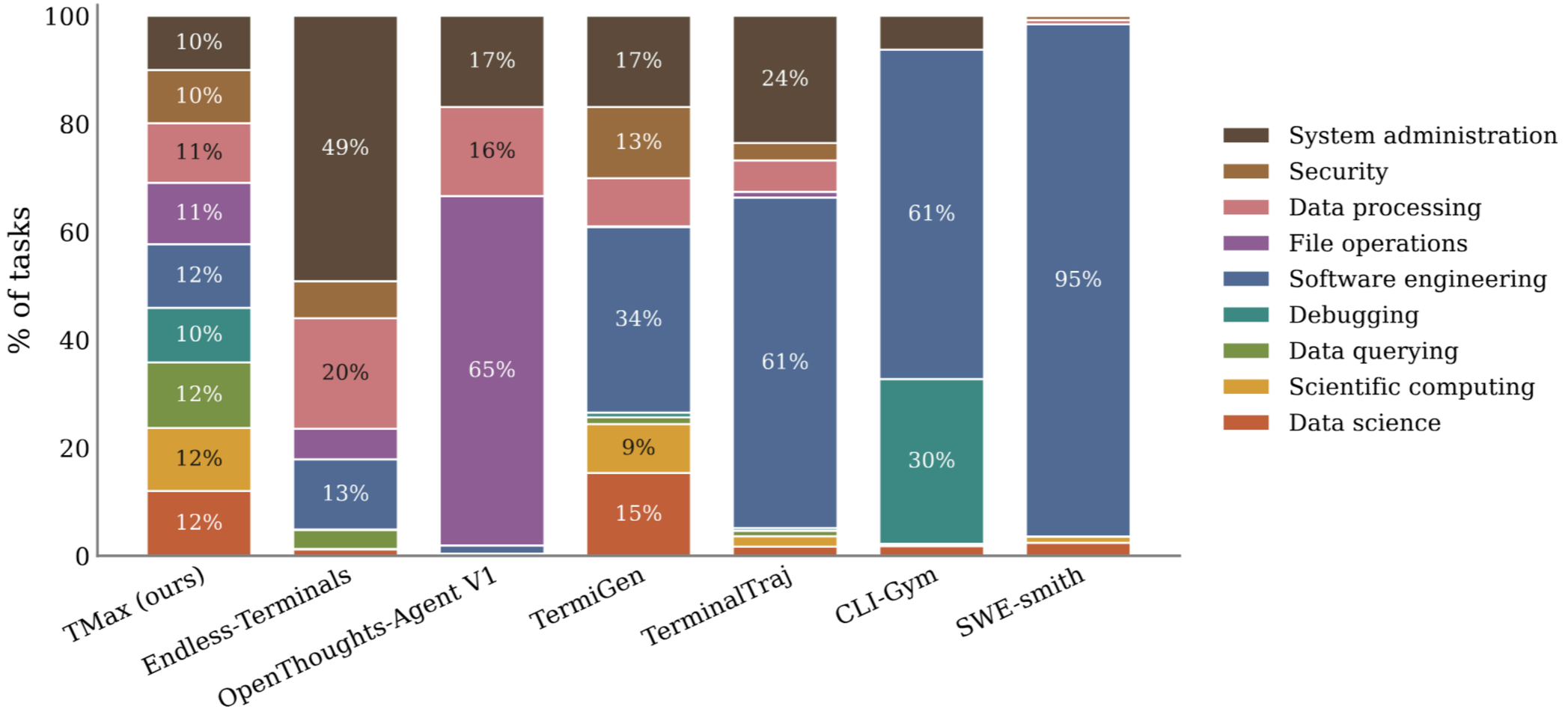


LABEL	SHARE
VERIFIER_TOO_WEAK	76.4%
ANSWER_LEAK	10.9%
CLEAN	7.7%
INSTR_VERIFIER_MISMATCH	3.0%
INSTR_ENV_MISMATCH	1.2%
OTHER	0.6%
INSTR_AMBIGUOUS	0.2%
TASK_TRIVIAL	0.0%

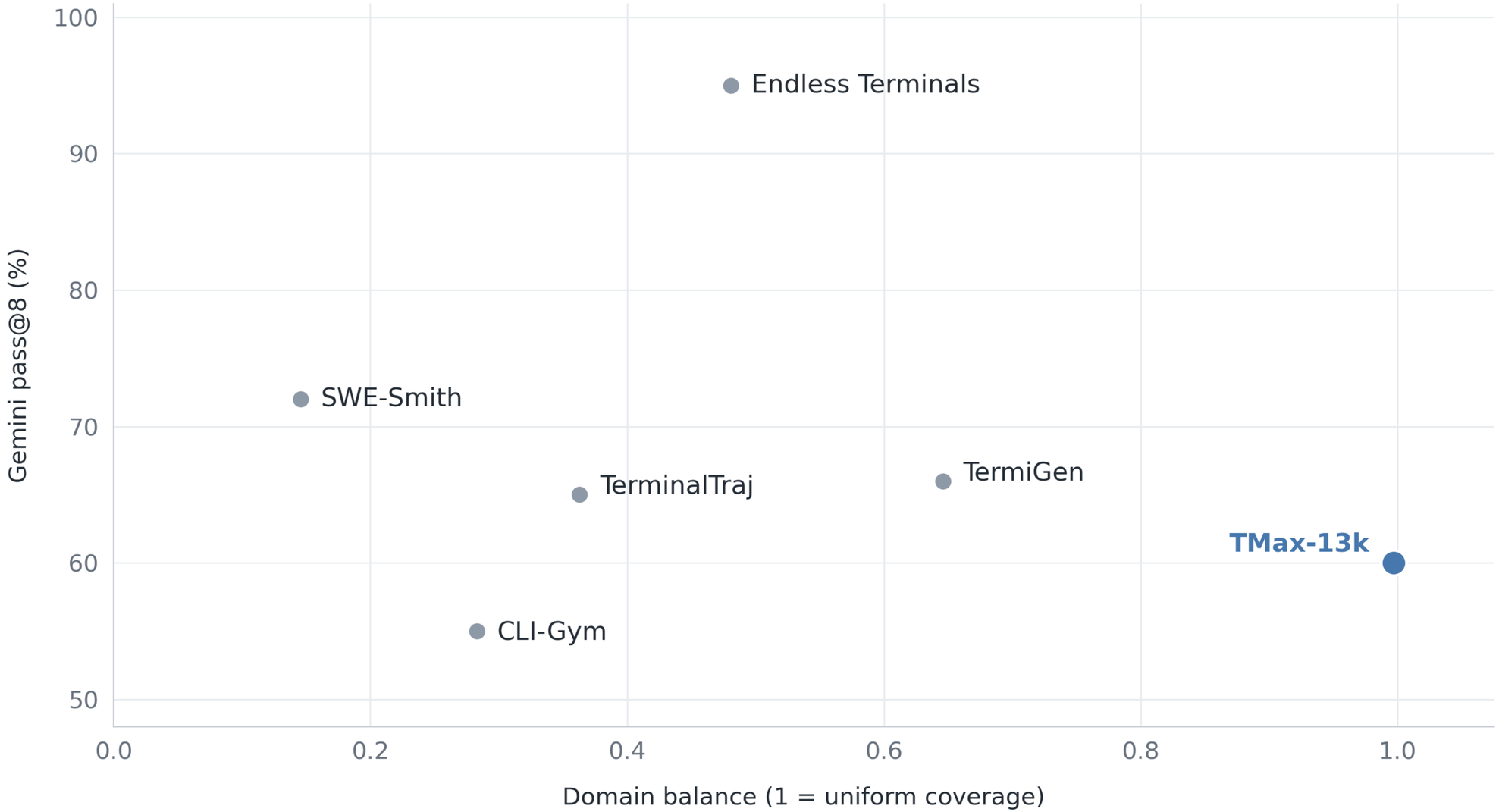
TMax: data generation



TMax: data balance



TMax: data difficulty



TMax: experimental setup

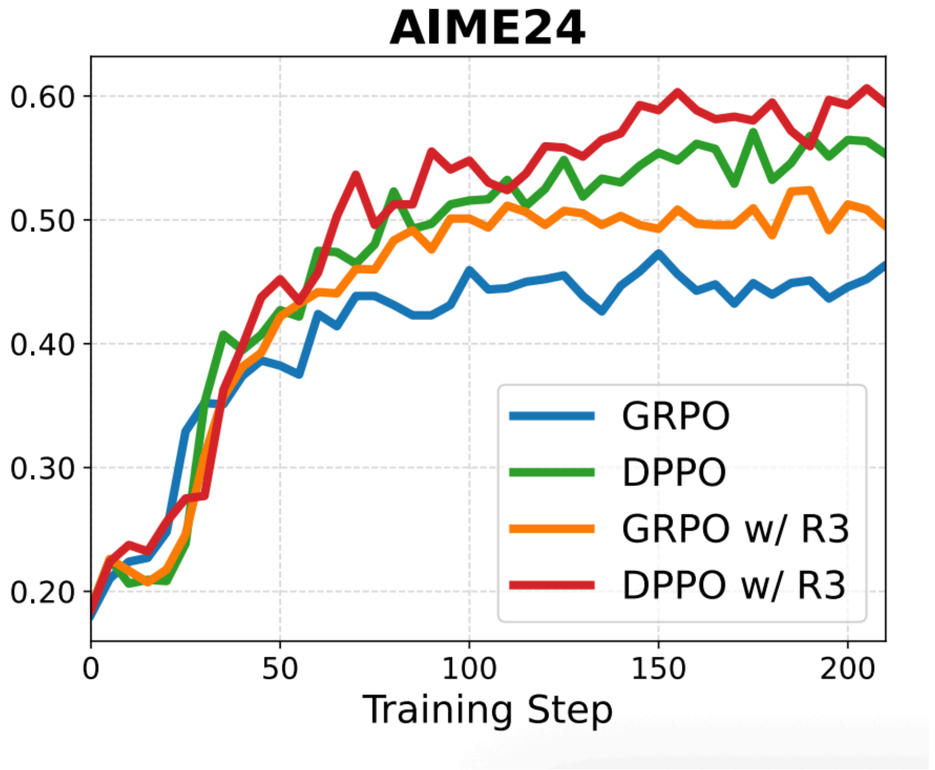
- Qwen 3.5 9B (instruct) with a custom mini-swe-agent harness
- 500 steps
- Use DPPO (better stability) with $\delta = 0.1$

DPPO?

$$L_\mu(\pi) = \mathbb{E}_{y\sim\mu}\left[\sum_{t=1}^{|y|} M_t \cdot \frac{\pi(y_t|s_t)}{\mu(y_t|s_t)} \cdot \hat{A}_t\right],$$

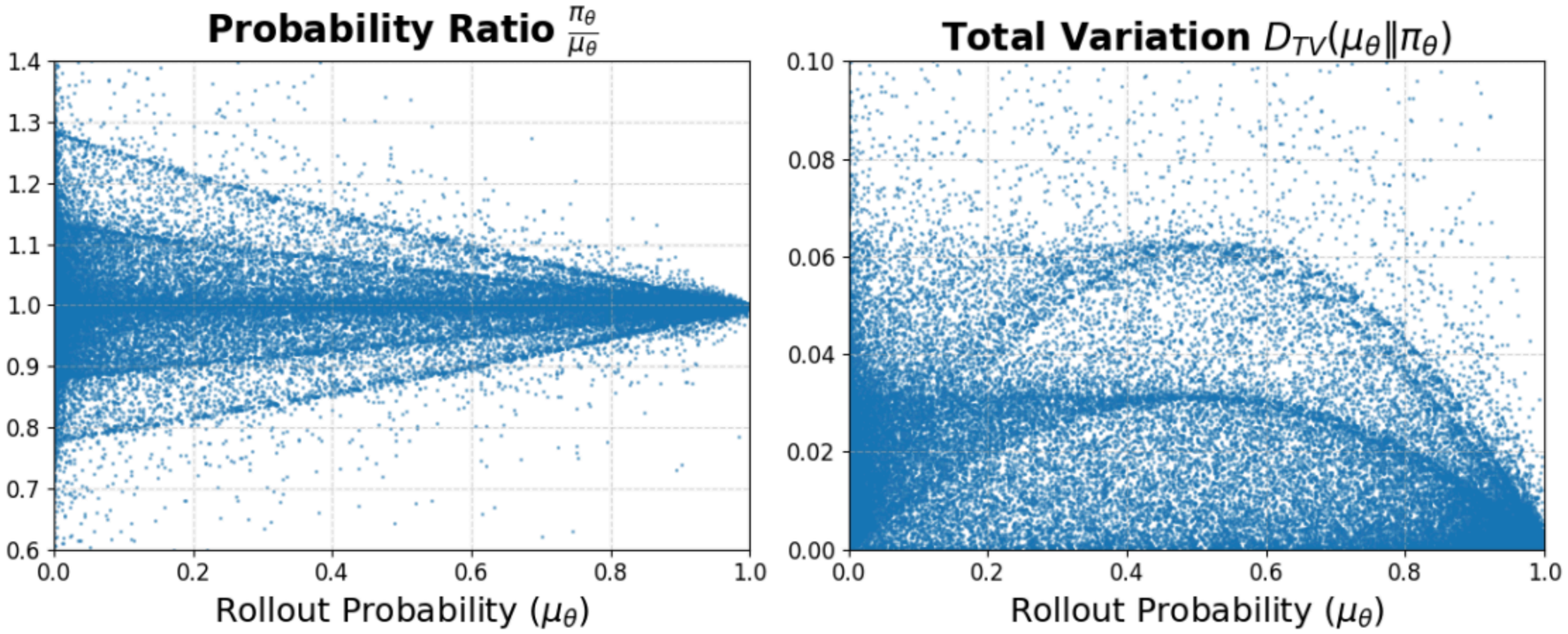
$$M_t^{\text{PPO}} = \begin{cases} 0, & \text{if } \hat{A}_t > 0 \text{ and } \frac{\pi(y_t|s_t)}{\mu(y_t|s_t)} > 1 + \epsilon_{\text{high}}, \\ 0, & \text{if } \hat{A}_t < 0 \text{ and } \frac{\pi(y_t|s_t)}{\mu(y_t|s_t)} < 1 - \epsilon_{\text{low}}, \\ 1, & \text{otherwise,} \end{cases}$$

$$M_t^{\text{DPPO}} = \begin{cases} 0, & \text{if } \hat{A}_t > 0 \text{ and } \pi(y_t|s_t) - \mu(y_t|s_t) > \delta, \\ 0, & \text{if } \hat{A}_t < 0 \text{ and } \mu(y_t|s_t) - \pi(y_t|s_t) > \delta, \\ 1, & \text{otherwise.} \end{cases}$$



https://arxiv.org/abs/2602.04879

DPPO?

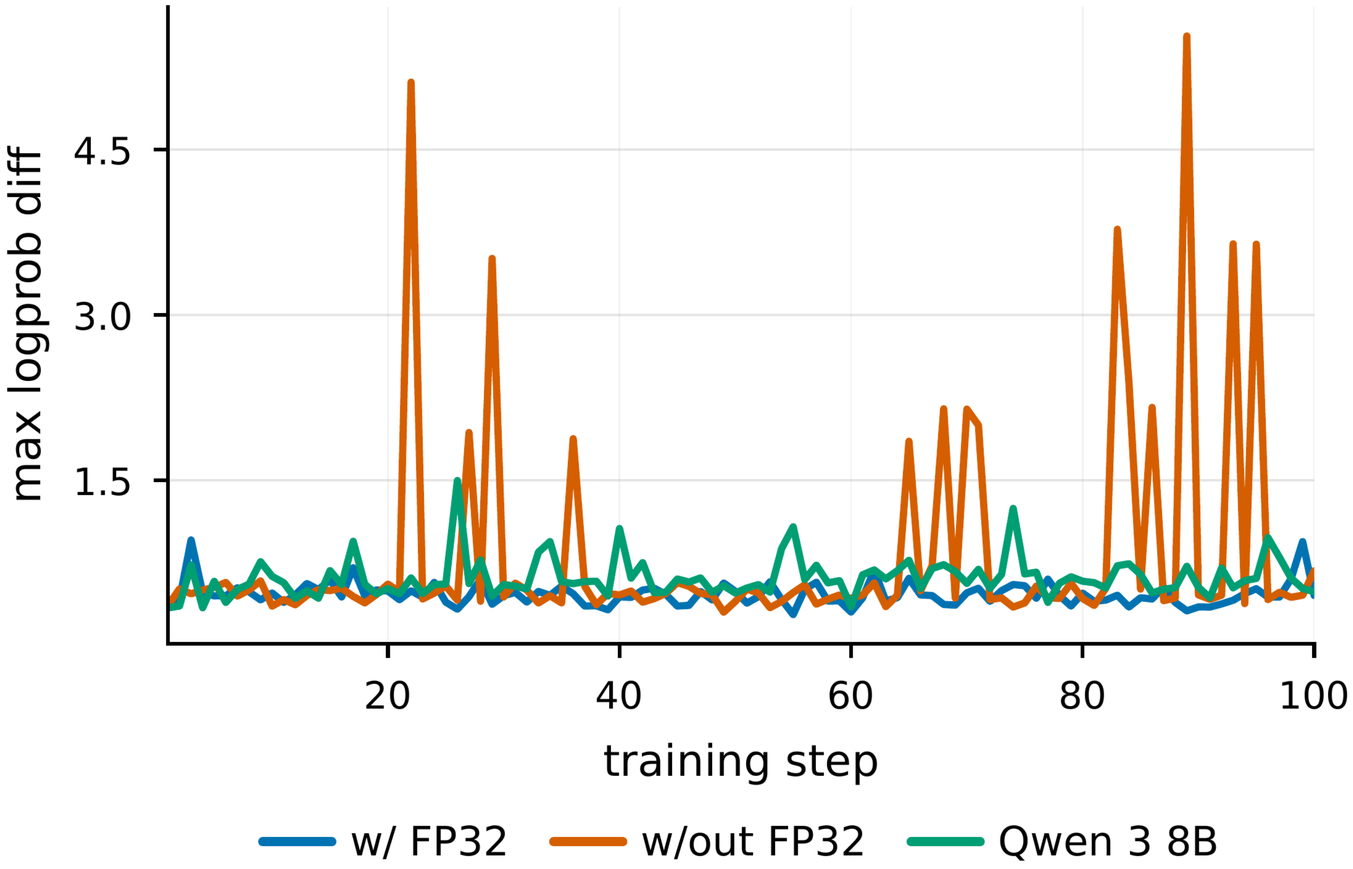


https://arxiv.org/abs/2602.04879

TMax: RL training

Three important tricks for stability:

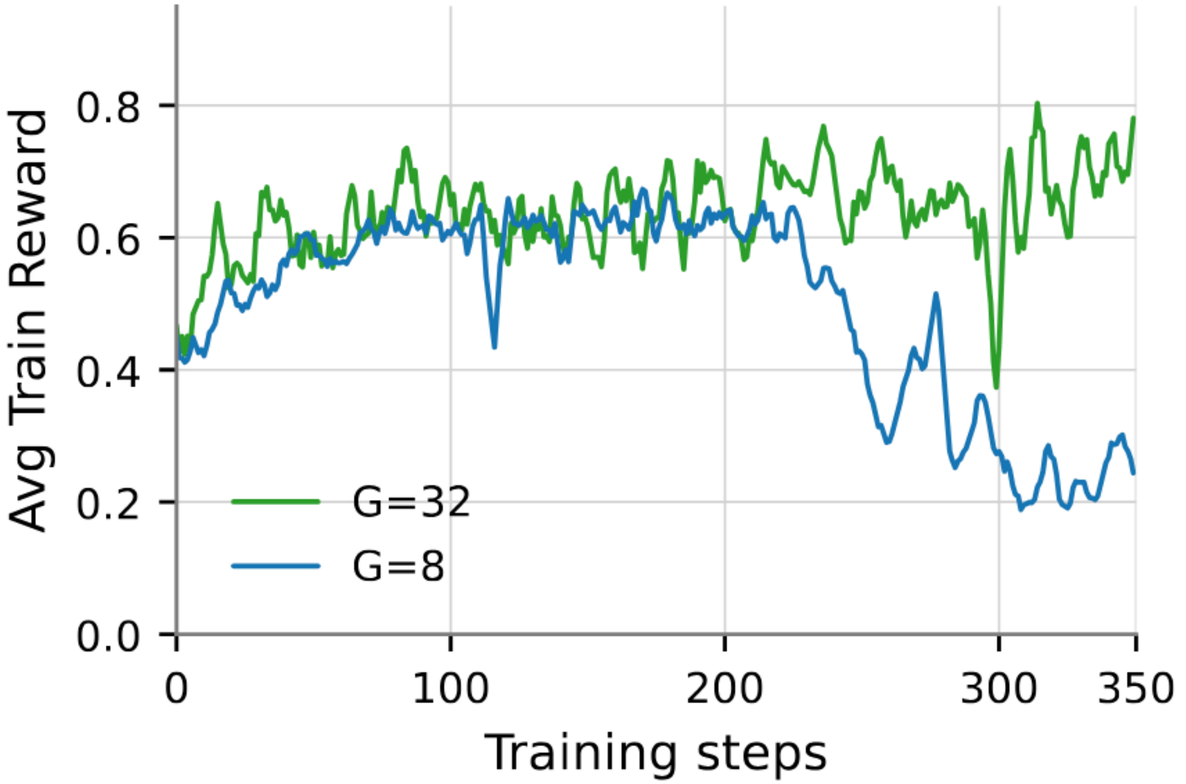
1. FP32 LM head



TMax: RL training

Three important tricks for stability:

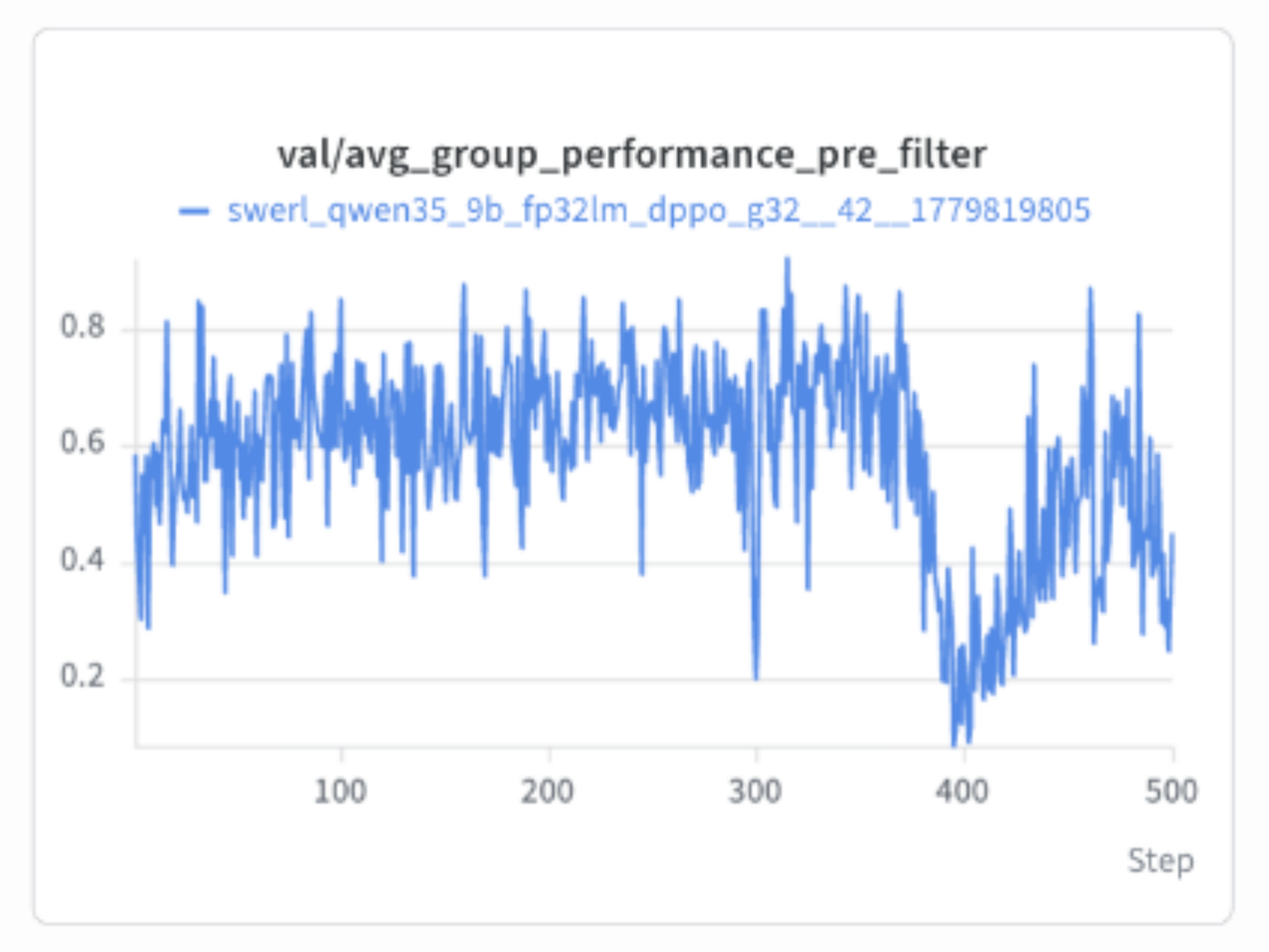
1. FP32 LM head
2. Large group size



TMax: RL training

Three important tricks for stability:

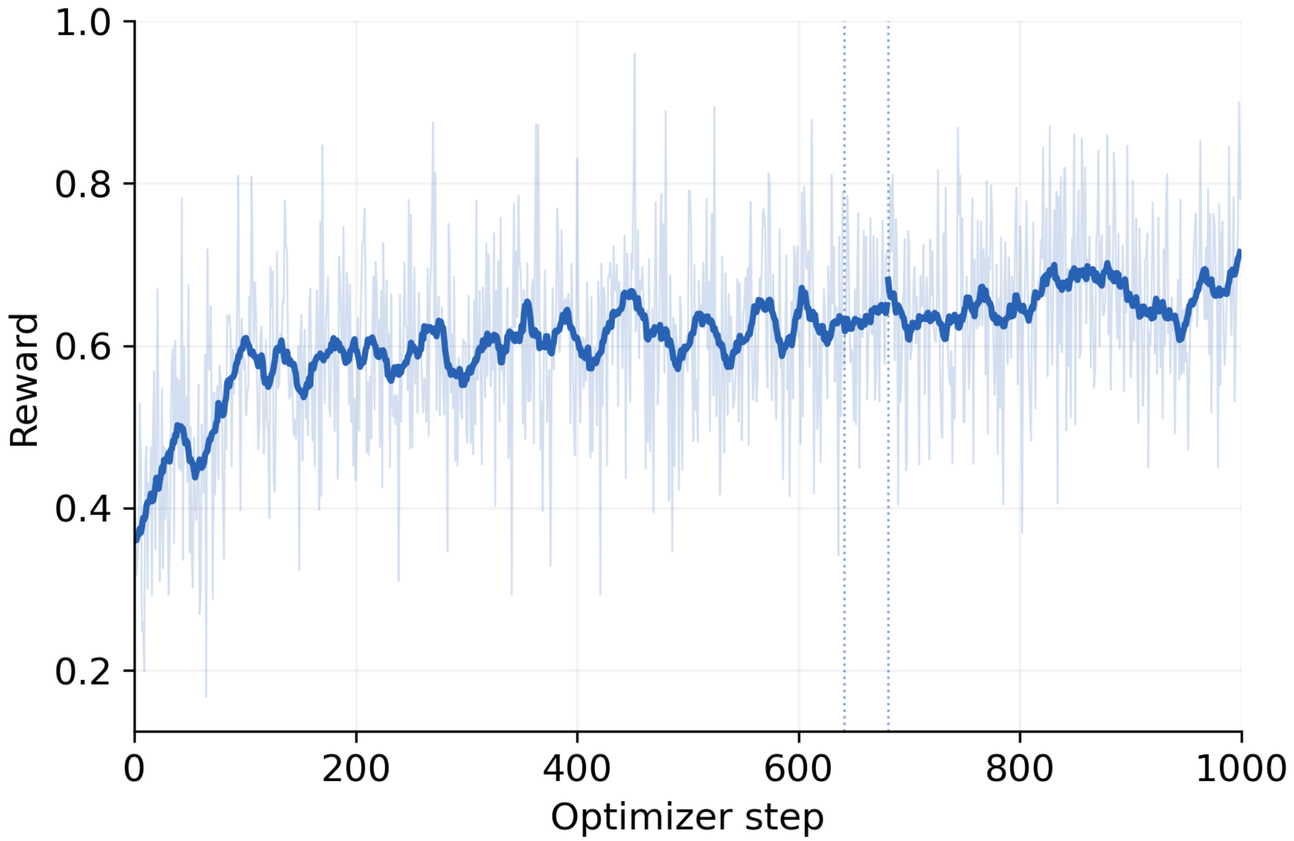
1. FP32 LM head
2. Large group size
3. Stable infrastructure



TMax: RL training

Three important tricks for stability:

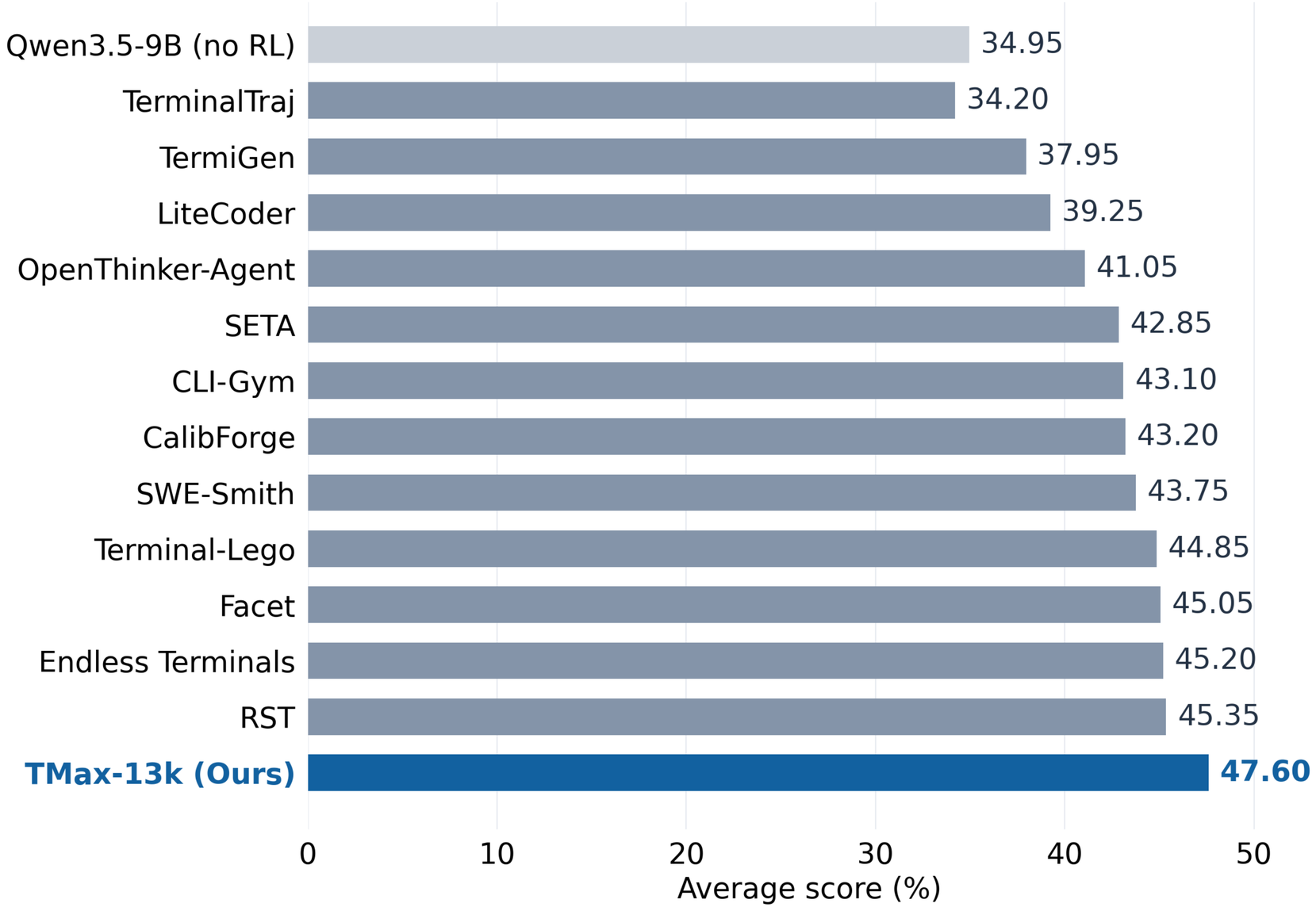
1. FP32 LM head
2. Large group size
3. Stable infrastructure



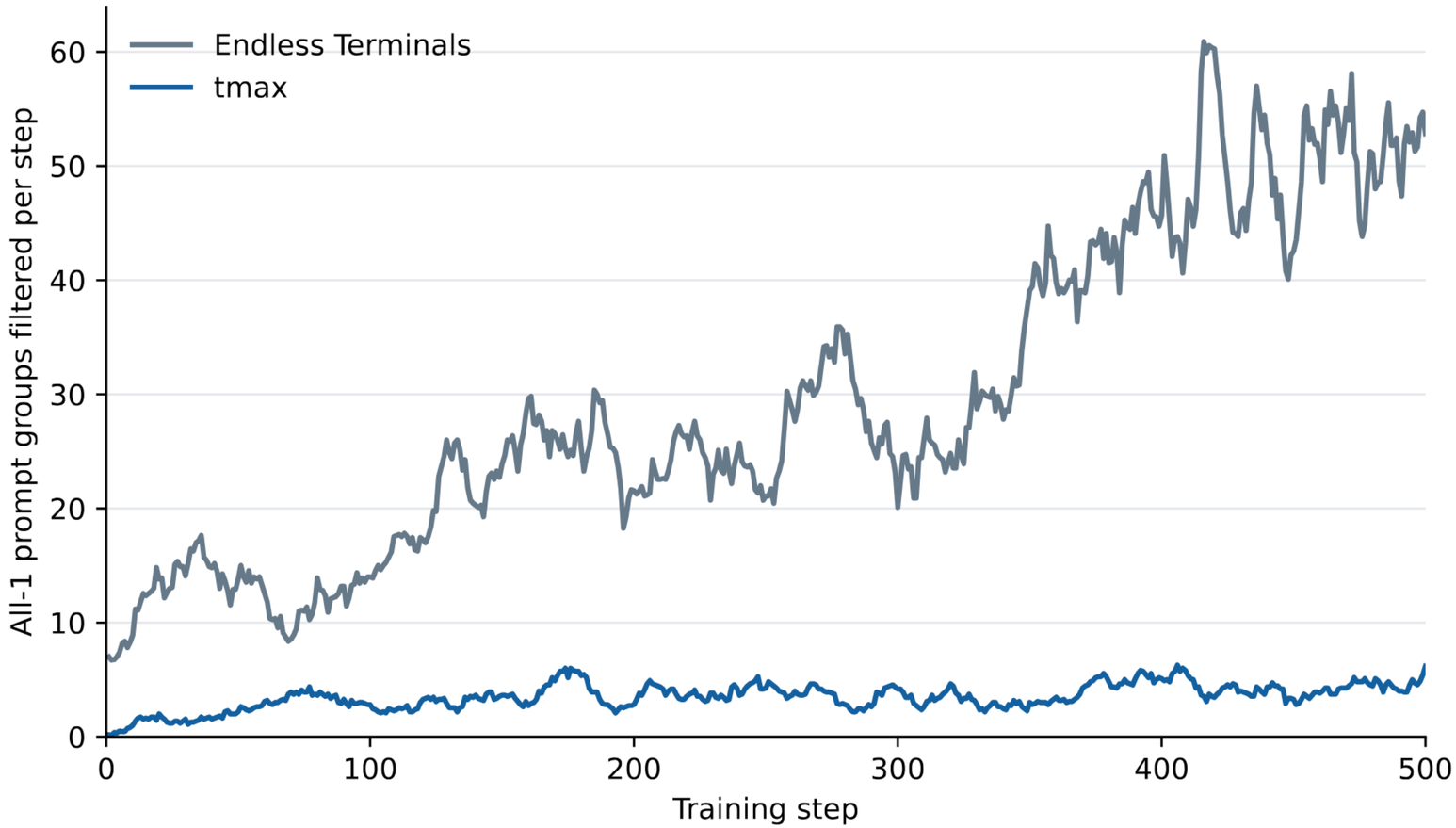
TMax: experimental setup

- Qwen 3.5 9B (instruct) with a custom mini-swe-agent harness
- 500 steps, 8 prompts × 32 groups
- Use DPPO (better stability) with $\delta = 0.1$
- Evaluate on Terminal-Bench Lite and Terminal-Bench 2.1
  - Evaluate every 100 steps and select the best TB Lite checkpoint
  - Use the same harness and settings across all evaluations

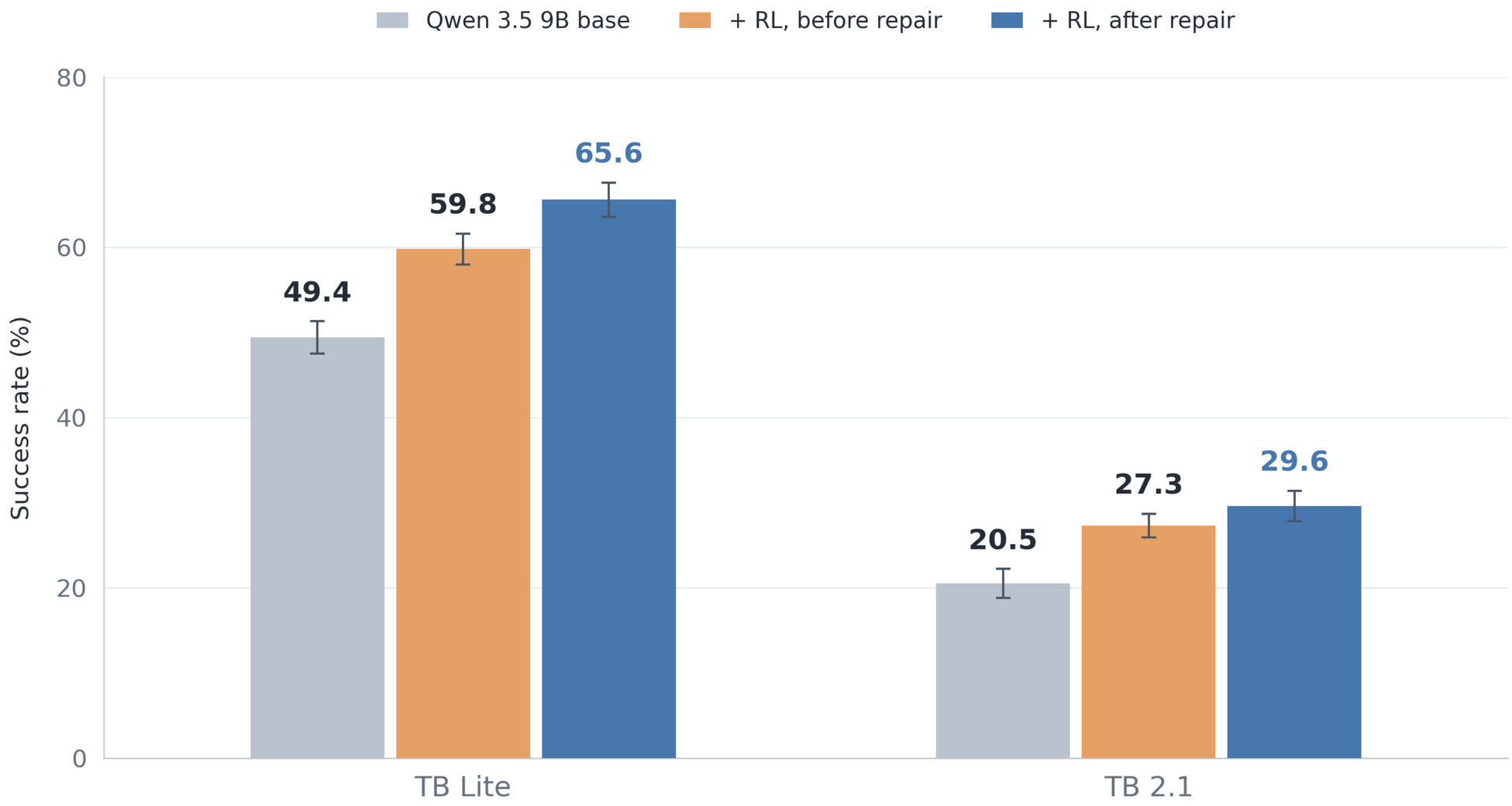
TMax: results



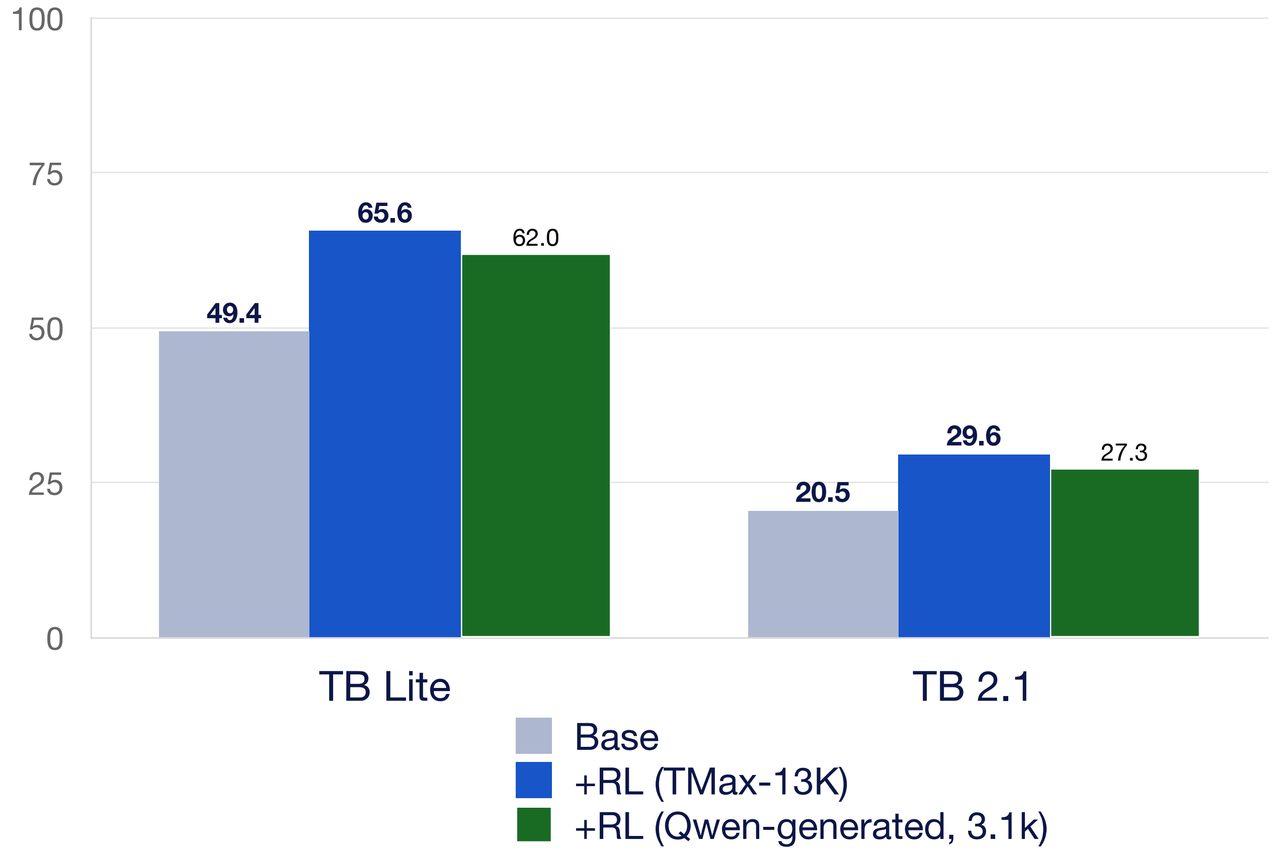
TMax: difficulty during training



TMax: importance of repair

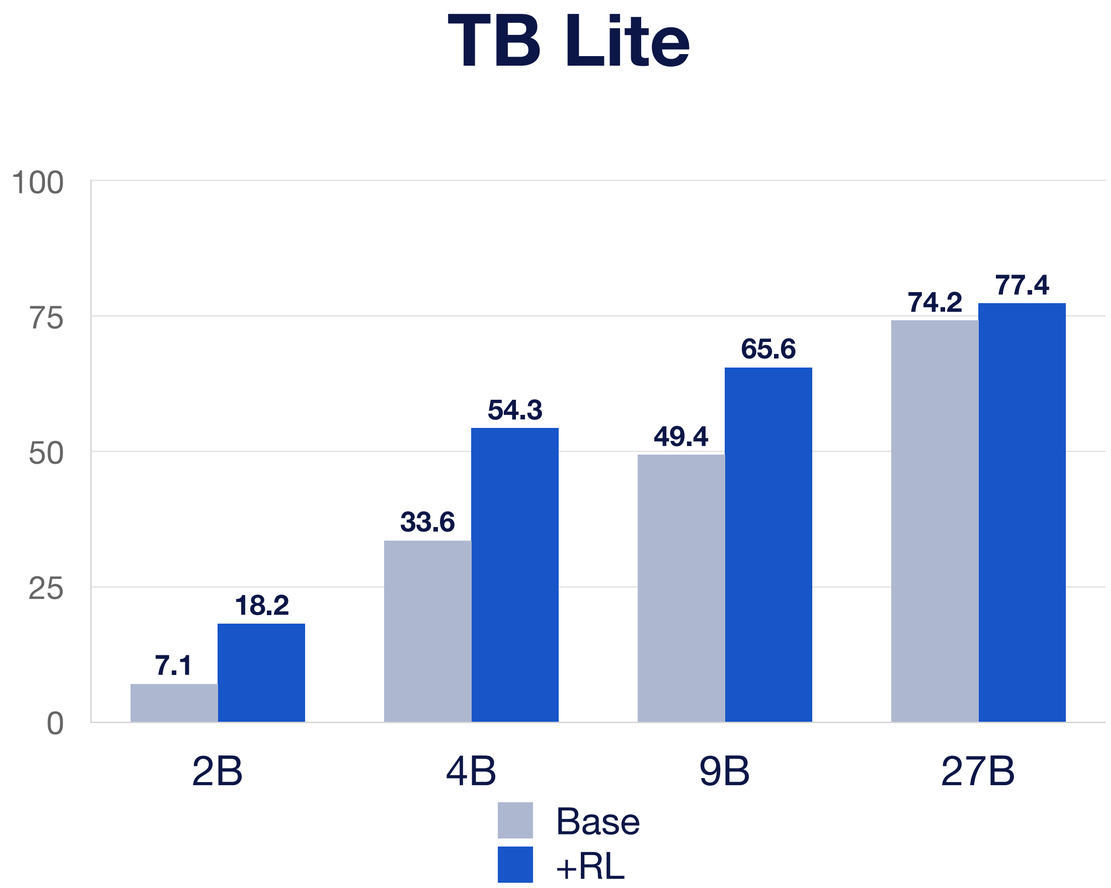


TMax does not rely on a strong teacher



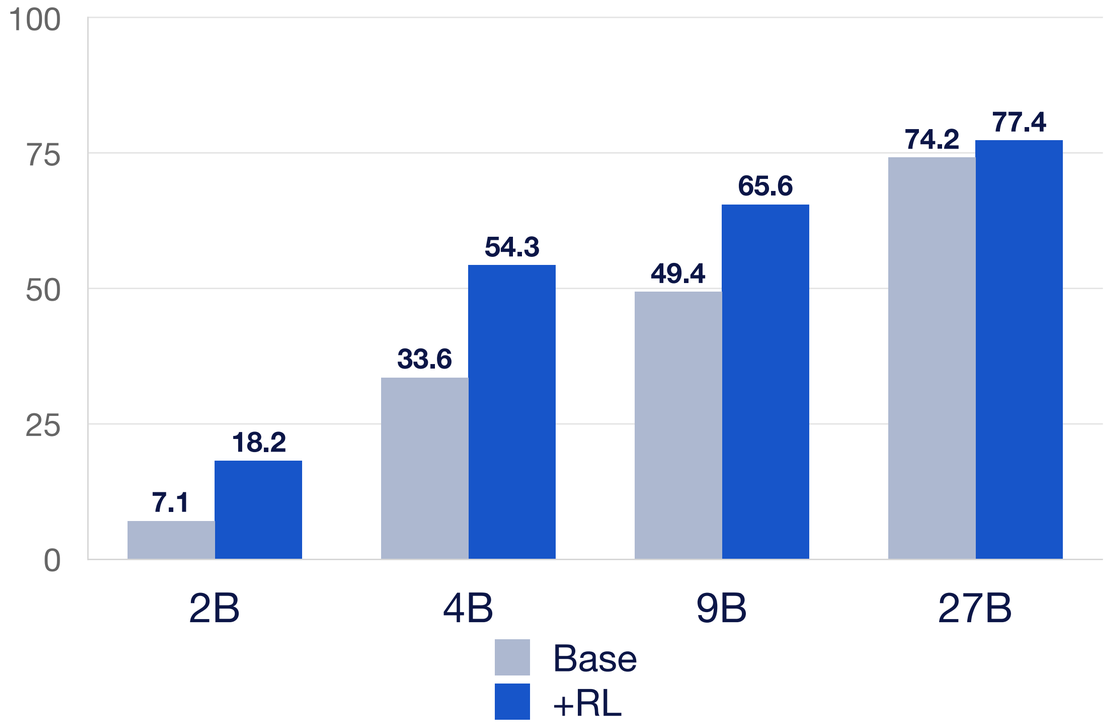
How well does TMax transfer?

TMax is effective across model sizes

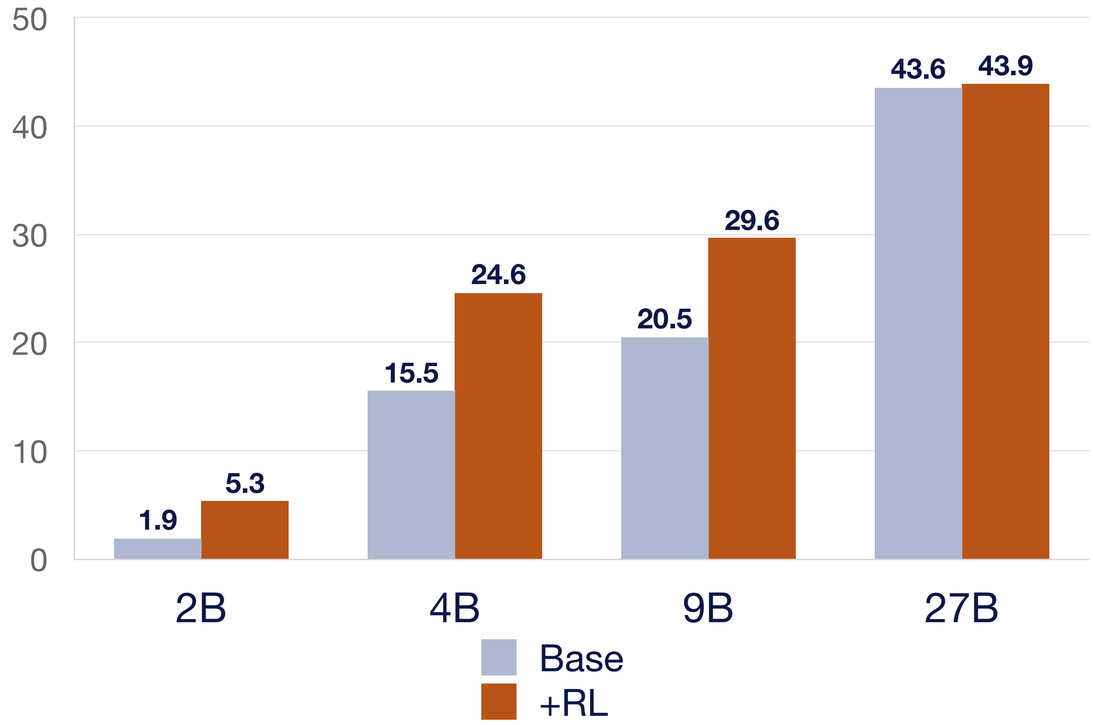


TMax is effective across model sizes

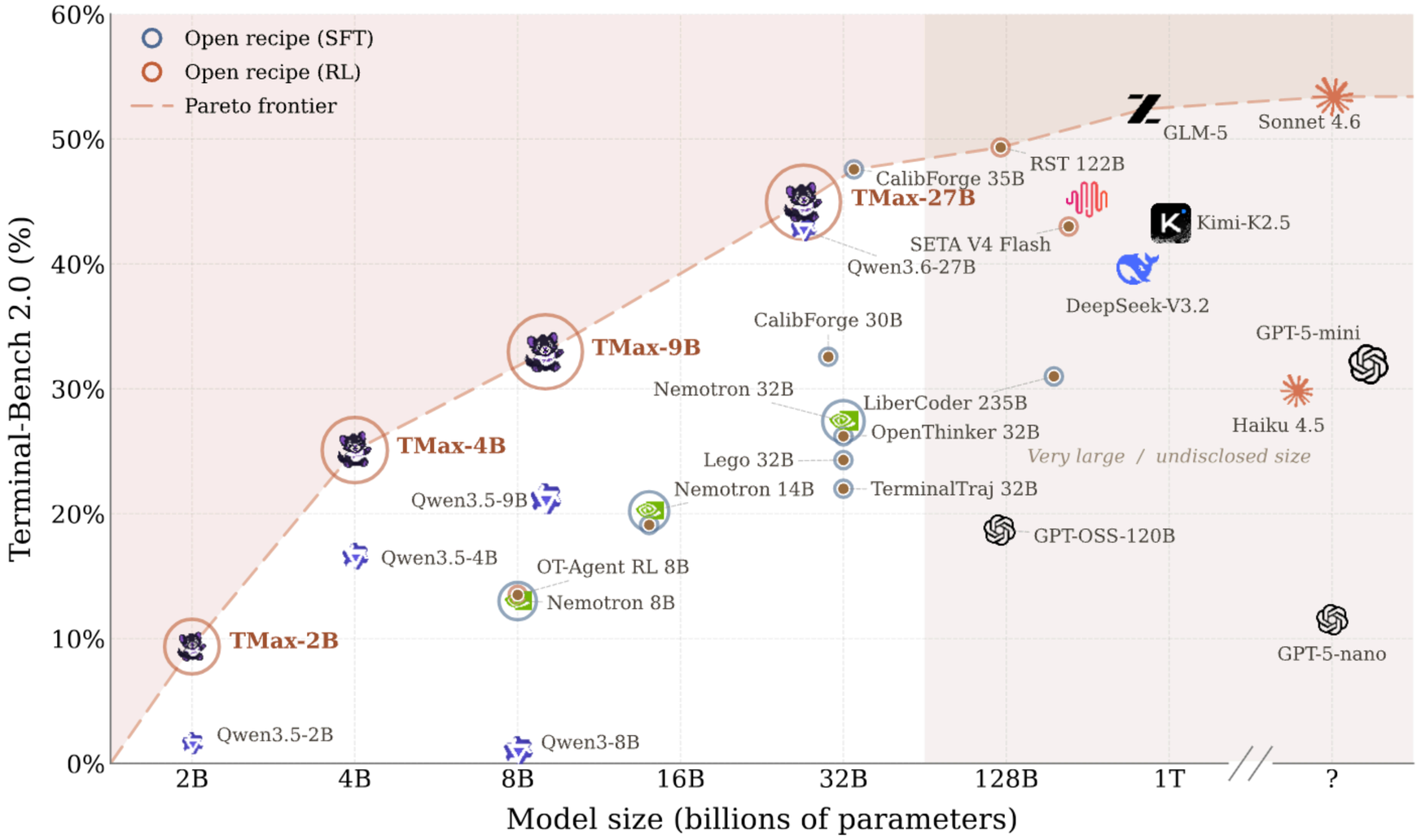
TB Lite



TB 2.1

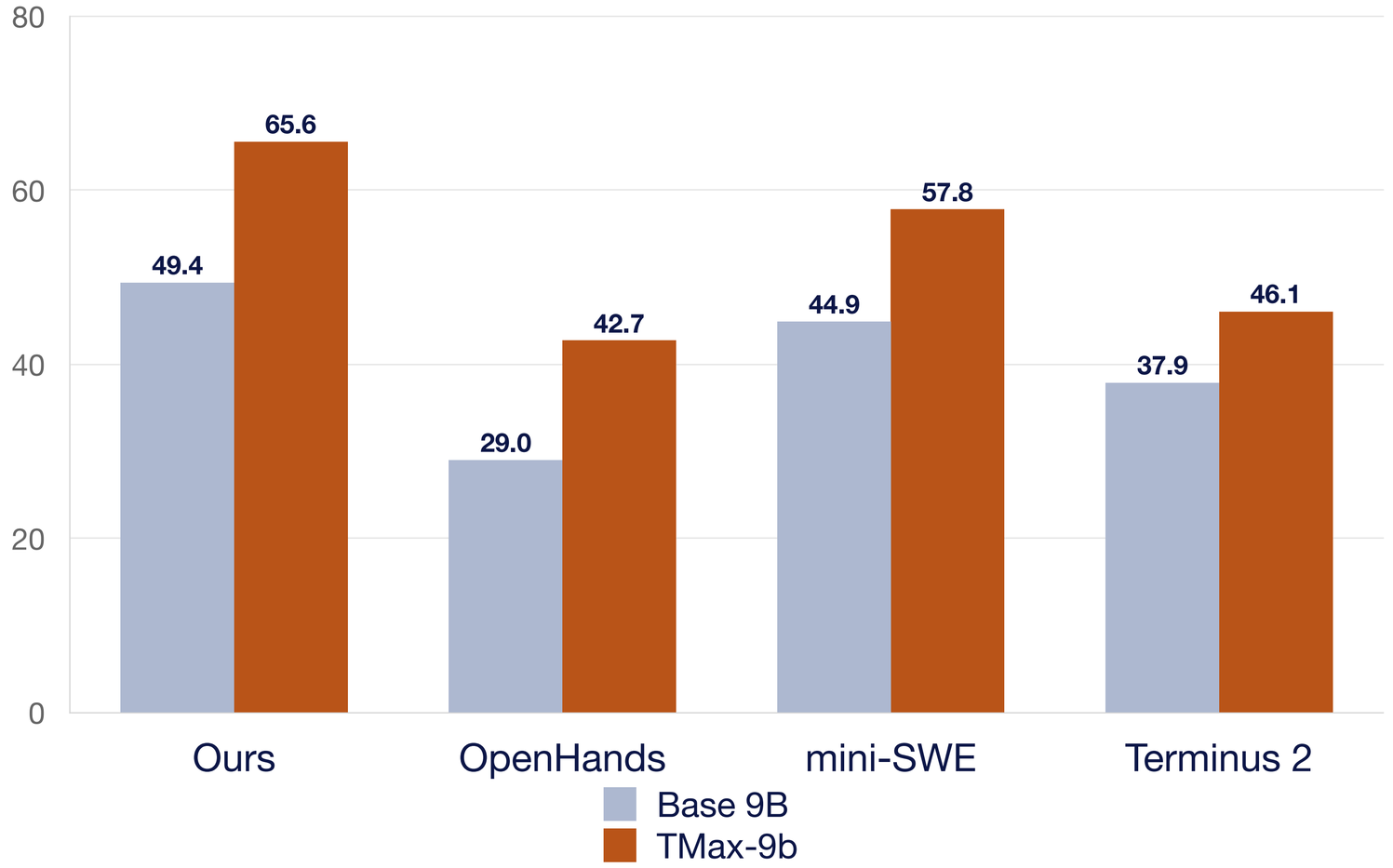


TMax sets the Pareto frontier!

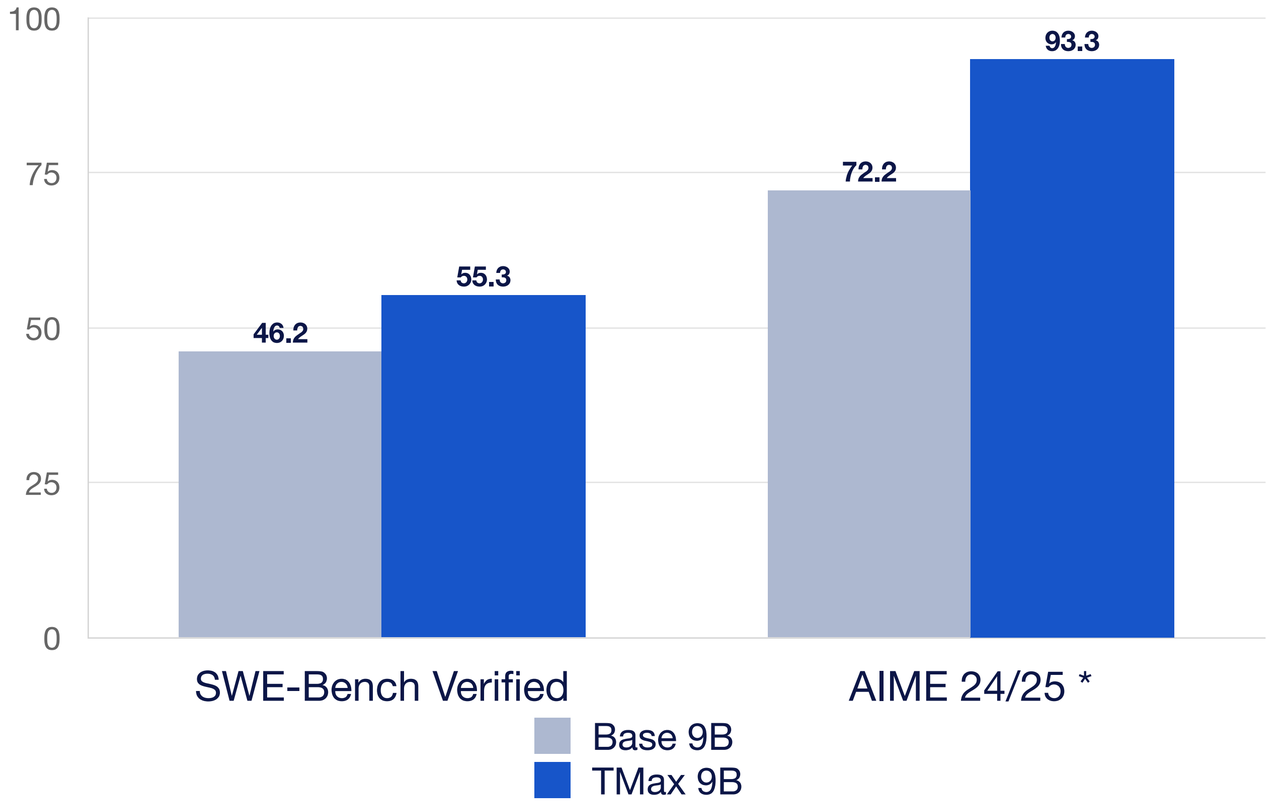


TMax is effective across harness types

TMax is effective across harness types



TMax is effective across tasks



*evaluation performed in sandbox+harness setup

Conclusion

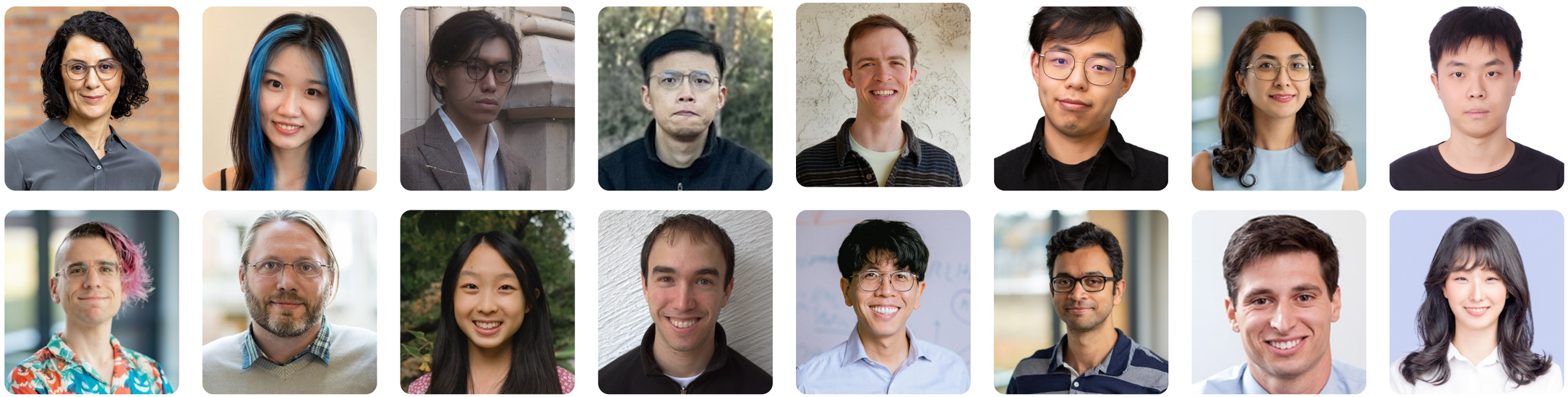
- Two different approaches for generating hard data: **adding rubrics on-the-fly** vs **fully synthetic generation upfront**

- **Careful engineering choices** required for stable and efficient training!

- Training **SOTA models** on **niche domains** does not require **super-large models!**

  - I would argue that these days it is easier than ever to pick up a Qwen, Muse, etc. model and train on your specific task to beat frontier models.

Thanks to my co-authors!







Resources





Code

GitHub

Paper

arXiv

Models + data

HF collection

Code

GitHub

Paper

arXiv

Model + data

HF collection

Let’s Chat!

H https://ivison.id.au/

Happy to answer further questions or generally chat:

 hamishiv@cs.washington.edu

 @hamishivi (and most other things)

Bonus Slides

TMax: entropy

