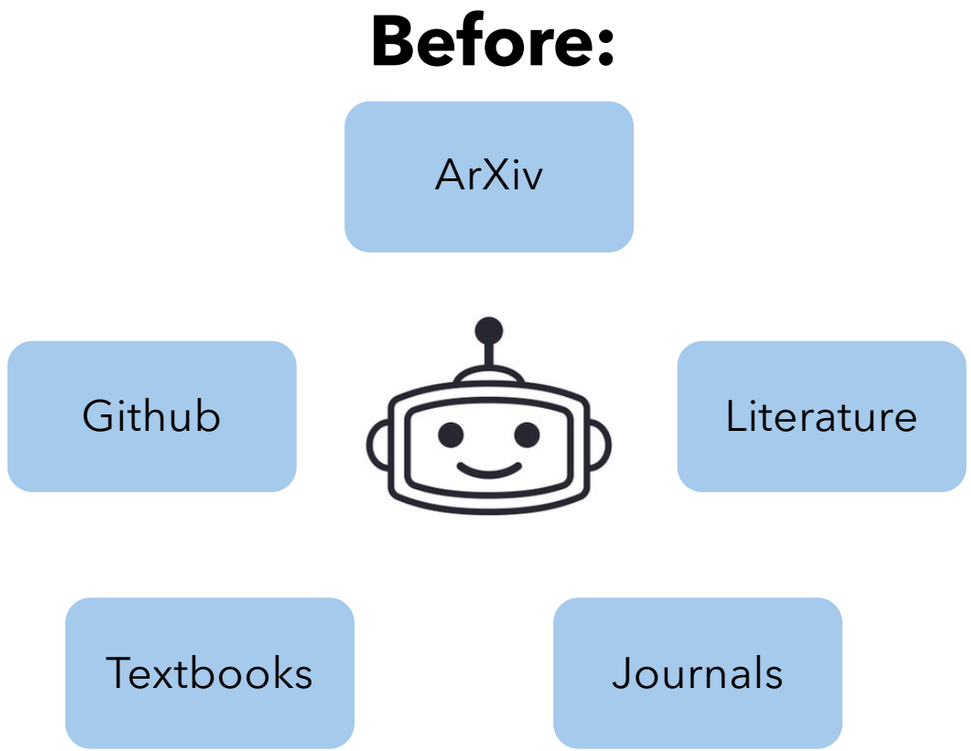
Self-improving at the Frontier of Learnability

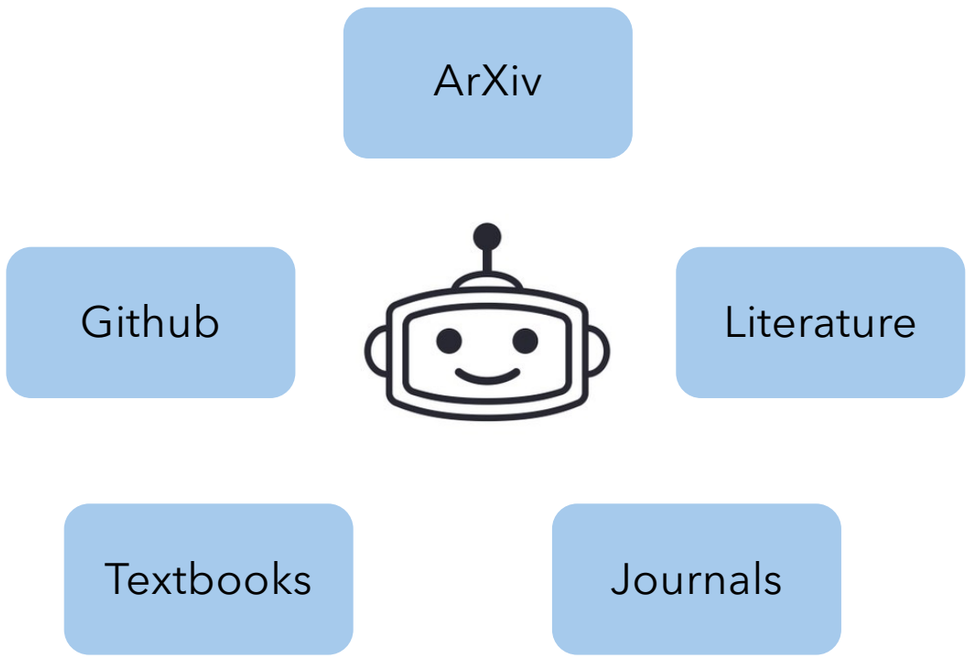
Shobhita Sundaram

How do we scale AI capabilities when human data runs out?

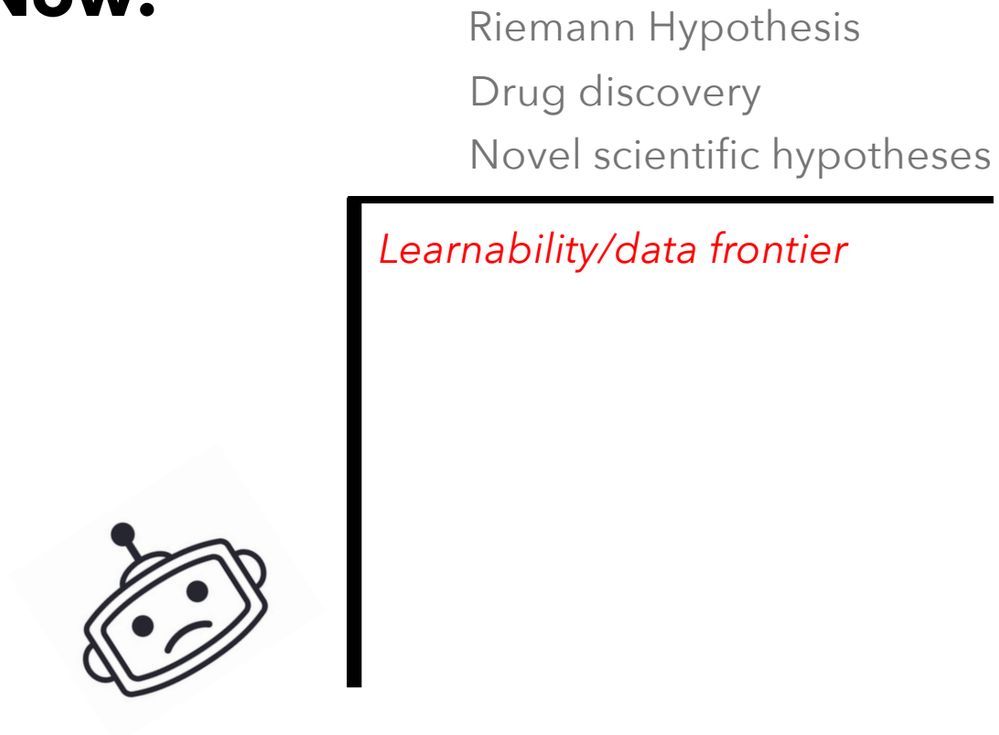


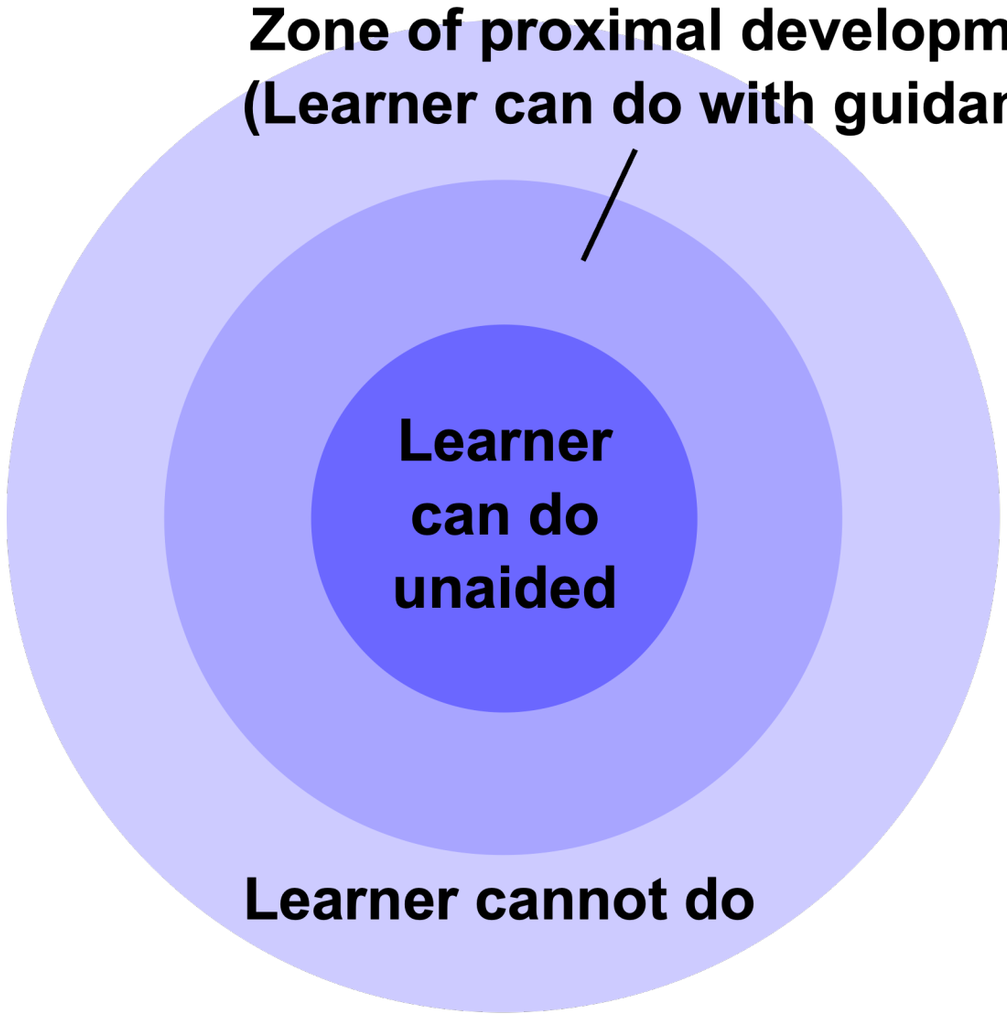
How do we scale AI capabilities when human data runs out?

Before:



Now:





Can models generate these curricula for themselves?

Teaching Models to Teach Themselves: Reasoning at the Edge of Learnability

https://ssundaram21.github.io/soar/


Shobhita Sundaram*[1]


John Quan[2]


Ariel Kwiatkowski[2]


Kartik Ahuja[2]

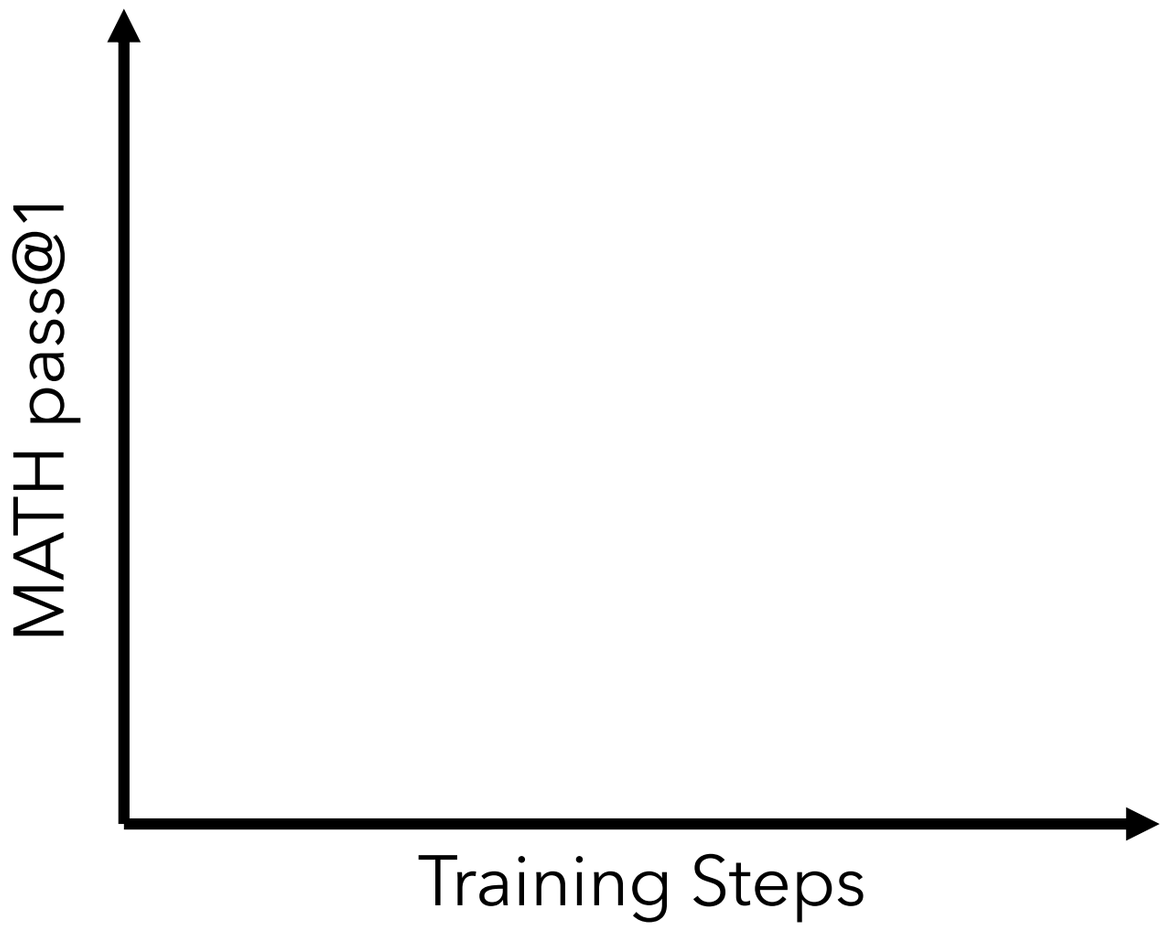

Yann Ollivier[2]

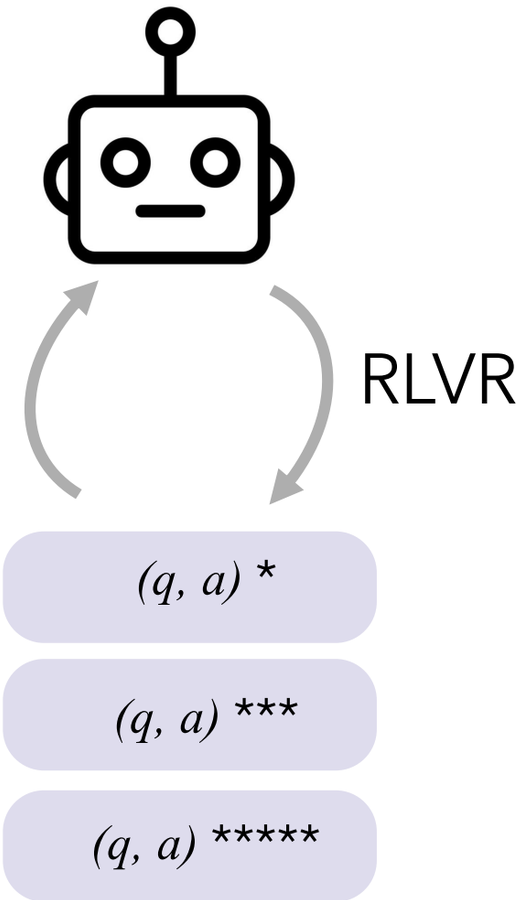

Julia Kempe[2,3]

*Work done during an internship at Meta

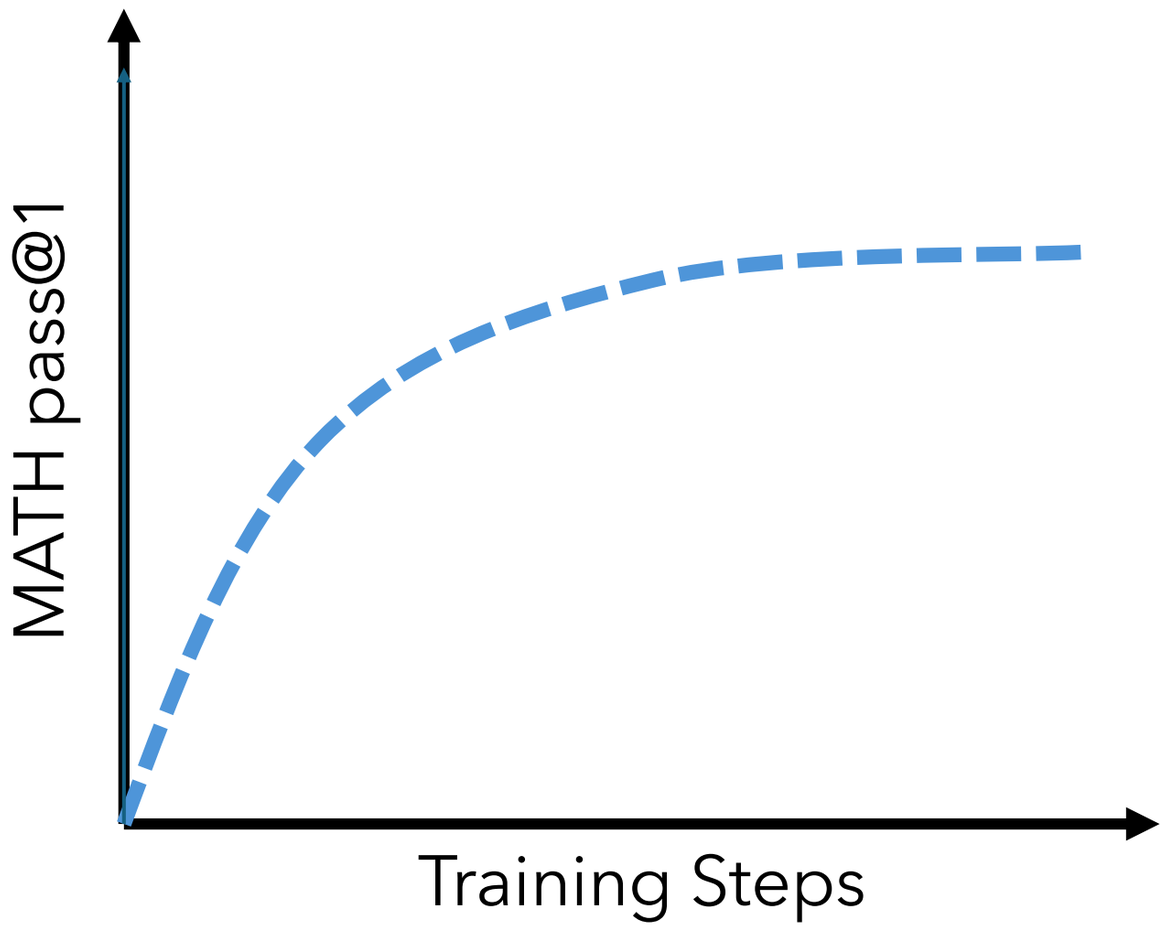
[1] MIT [2] Meta [3] NYU

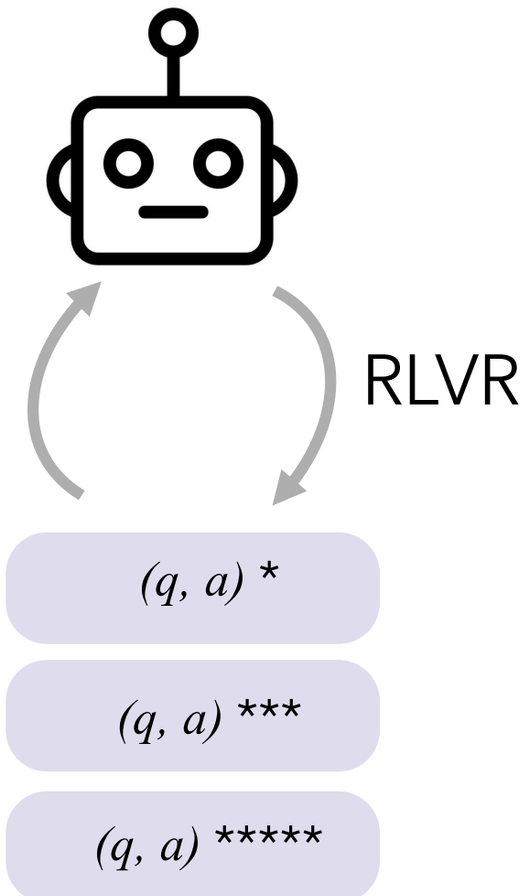
The sparse reward plateau



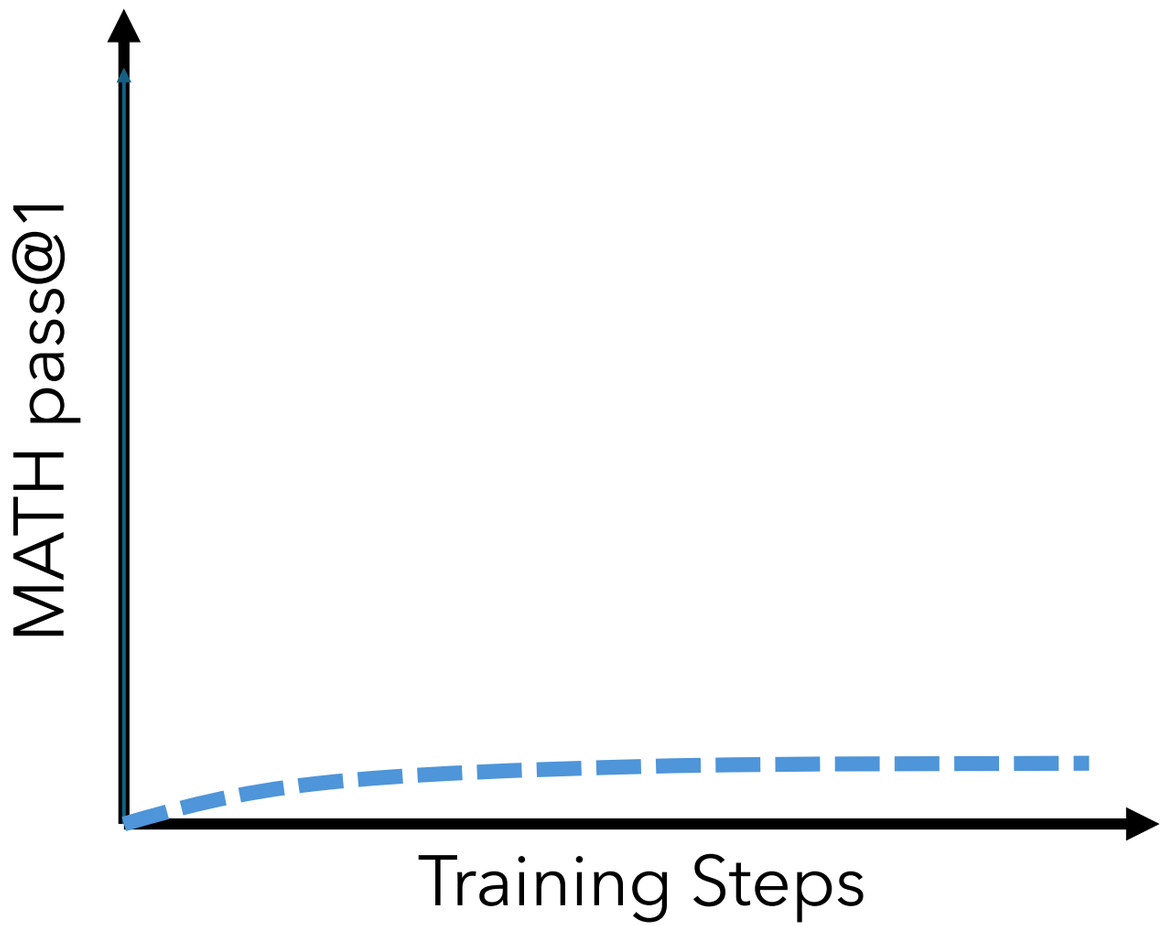


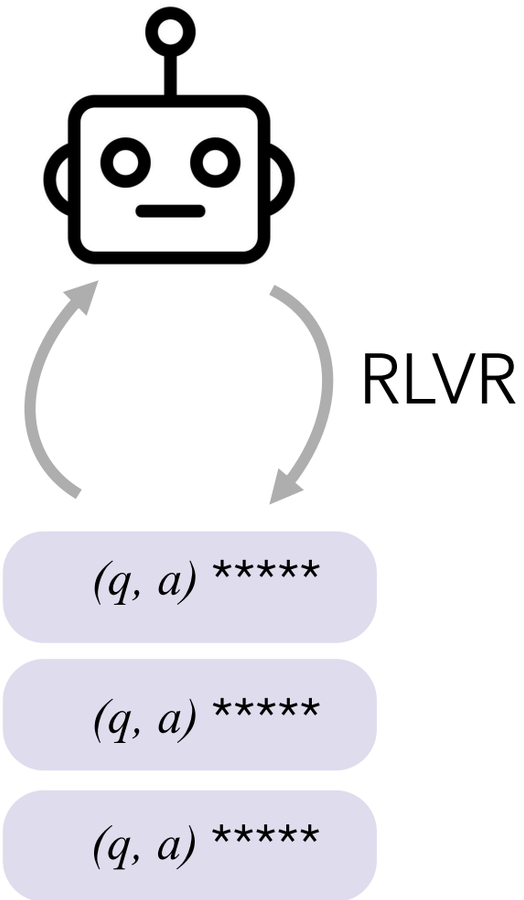
The sparse reward plateau





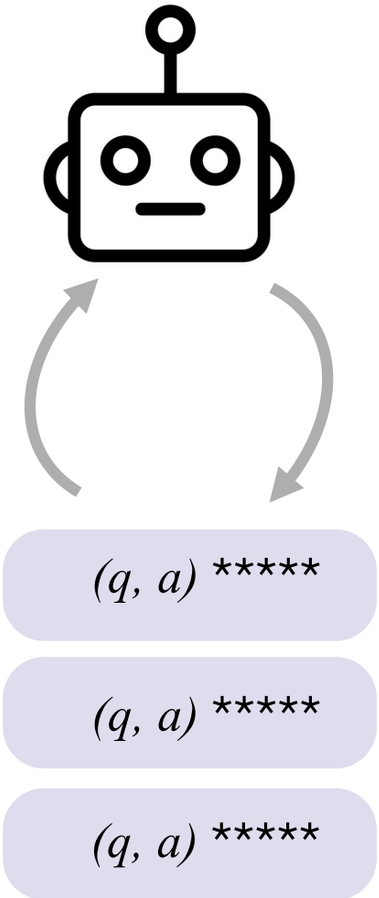
The sparse reward plateau



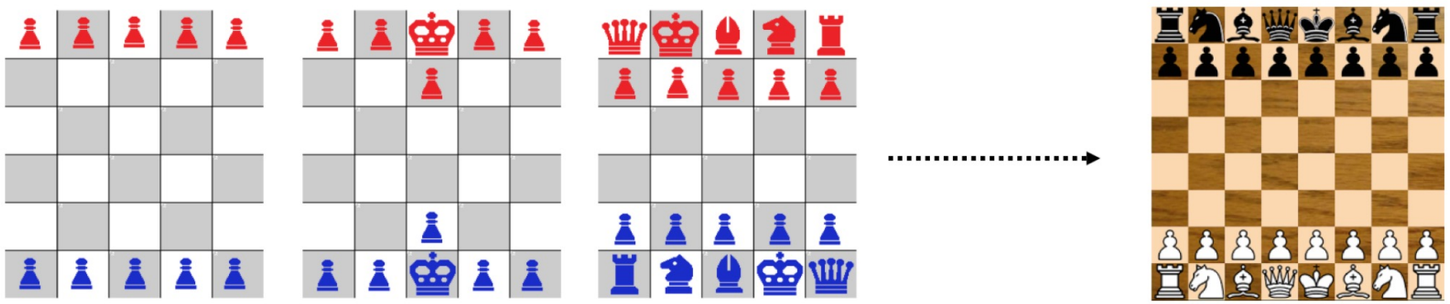


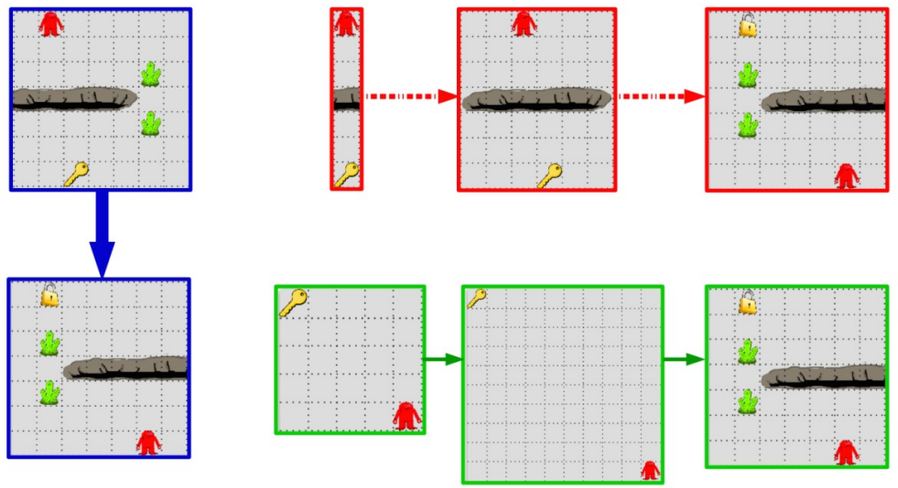
The sparse reward plateau

What do we do
when RL is stuck?



Curriculum learning – a potential solution?





Narvekar et al, JMLR 2020.

Not so easy, though…

Revisiting Generalization Across Difficulty Levels: It's Not So Easy

Yeganeh Kordi[♣] **Nihal V. Nayak**[♢] **Max Zuo**[♣] **Ilana Nguyen**[♣] **Stephen H. Bach**[♣]
[♣]Brown University [♢]Harvard University
{kordi, stephen_bach}@brown.edu

Paper	Core Claim	Difficult for whom?	Training Method
Hase et al. (2024)	Training on easy data performs almost as well on the hard test set as training on hard data.	LLM + Human	SFT, ICL, Linear Probing
Sun et al. (2024)	Training only on easy tasks can outperform training on all tasks.	Human	RL
Yang et al. (2024)	Hard data improves the model's consistency on similar questions more effectively than easy data.	Human	SFT, ICL
Pikus et al. (2025)	Training on the hardest data performs best.	LLM	RL
Ding et al. (2024)	Training provides generalization to similar difficulties, but this generalization reduces as training difficulty increases.	LLM + Human	SFT
Our Analysis	Training on only hard or easy data fails to generalize to other difficulty levels. Human-centric difficulty metrics are not well-suited for studying LLMs.	LLM	SFT

Not so easy, though…

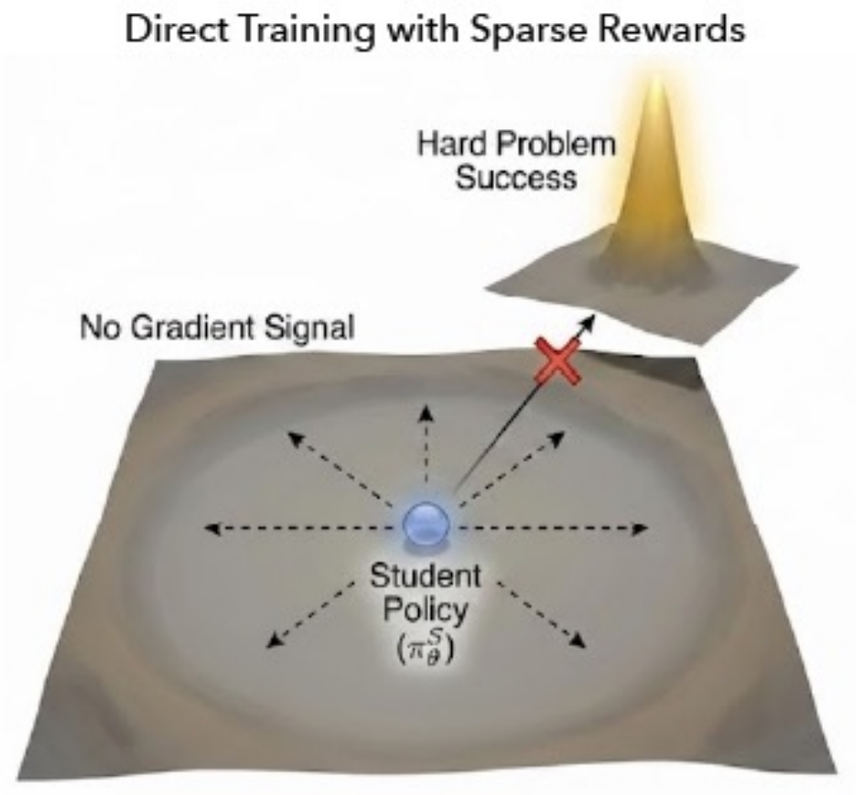
- Requires an existing pool of curated, known intermediate data
- Curricula can be fragile; the best learnable problems might be unavailable or unknown!

Can a model break its own reasoning plateau
by **self-generating** a curriculum?

Hypothesis: Latent pedagogical ability

Hypothesis: Pretrained LLMs can be finetuned to generate automated curricula that *make hard problems learnable, without being able to solve those hard problems.*

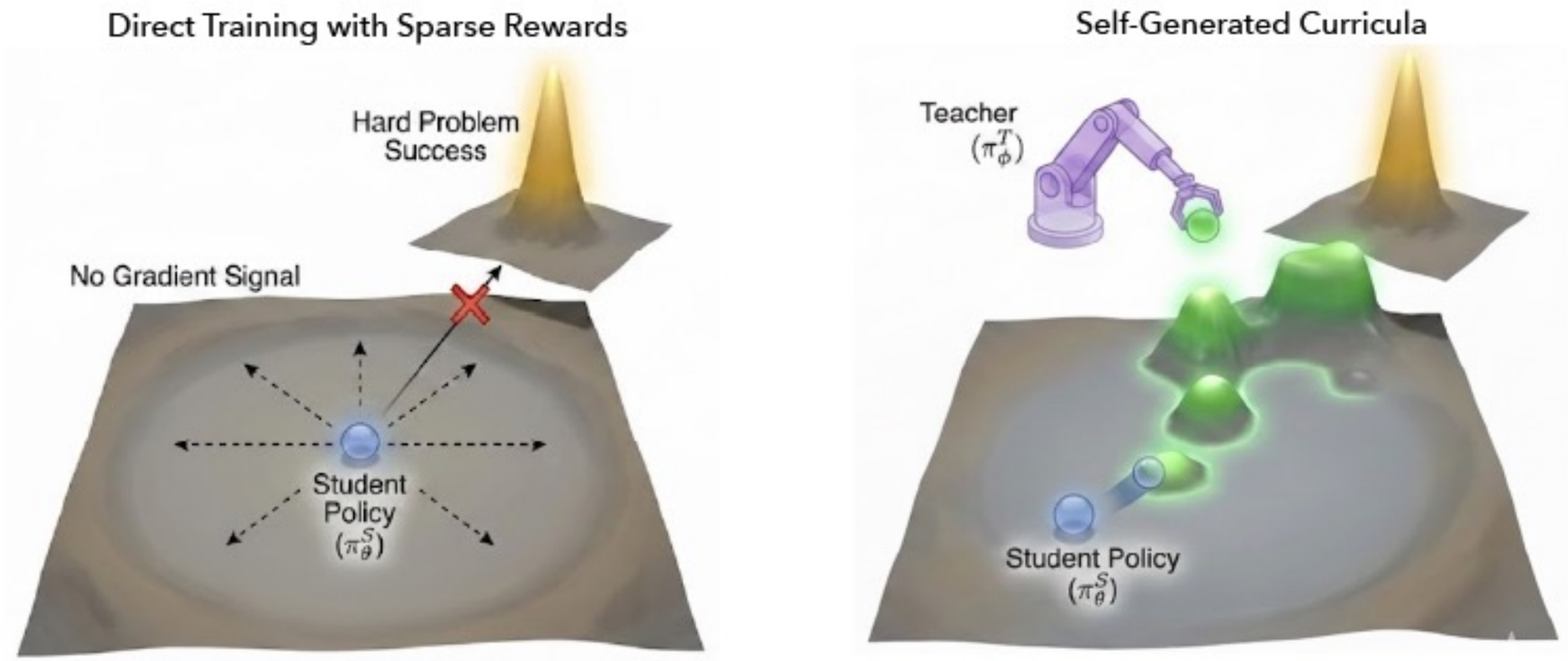
Hypothesis: Latent pedagogical ability



Hypothesis: Pretrained LLMs can be finetuned to generate automated curricula that *make hard problems learnable, without being able to solve those hard problems.*

Figure from Gemini

Hypothesis: Latent pedagogical ability



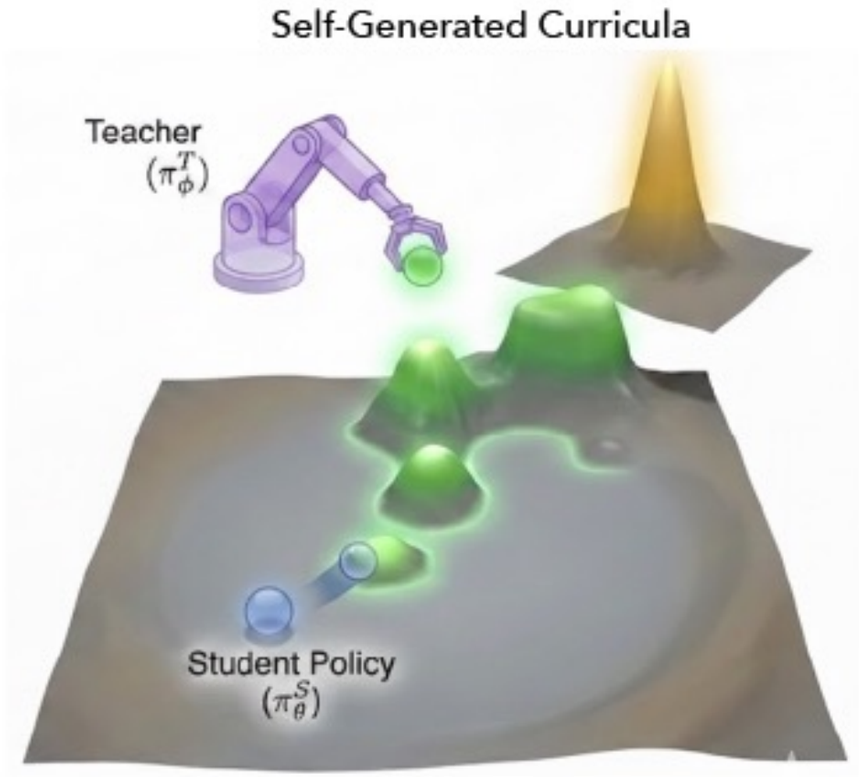
Hypothesis: Pretrained LLMs can be finetuned to generate automated curricula that *make hard problems learnable, without being able to solve those hard problems.*

Figure from Gemini

Hypothesis: Latent pedagogical ability

1. **Useful gradient signal:** Reinforces useful reasoning traces

2. **Learnability frontier:** Solving subtasks and getting non-zero rewards pushes the student policy into a more learnable reward region ("digging out" capabilities that can't be accessed with direct sampling/training)



Hypothesis: Pretrained LLMs can be finetuned to generate automated curricula that *make hard problems learnable, without being able to solve those hard problems.*

Figure from Gemini

Key Questions

1. Is this latent knowledge **present** and **extractable** without human curation?

2. Can we achieve this in domains with **sparse, binary** rewards **without automated question-answer verification**?

We introduce a framework to study these questions!

Formal setup

- Pretrained language model

$$\pi_\theta$$

- "Hard" problems (model produces 0/128 correct generations)

$$\mathscr{D} = \{(q_i, a_i)\}_{i=1}^{|\mathscr{D}|}$$

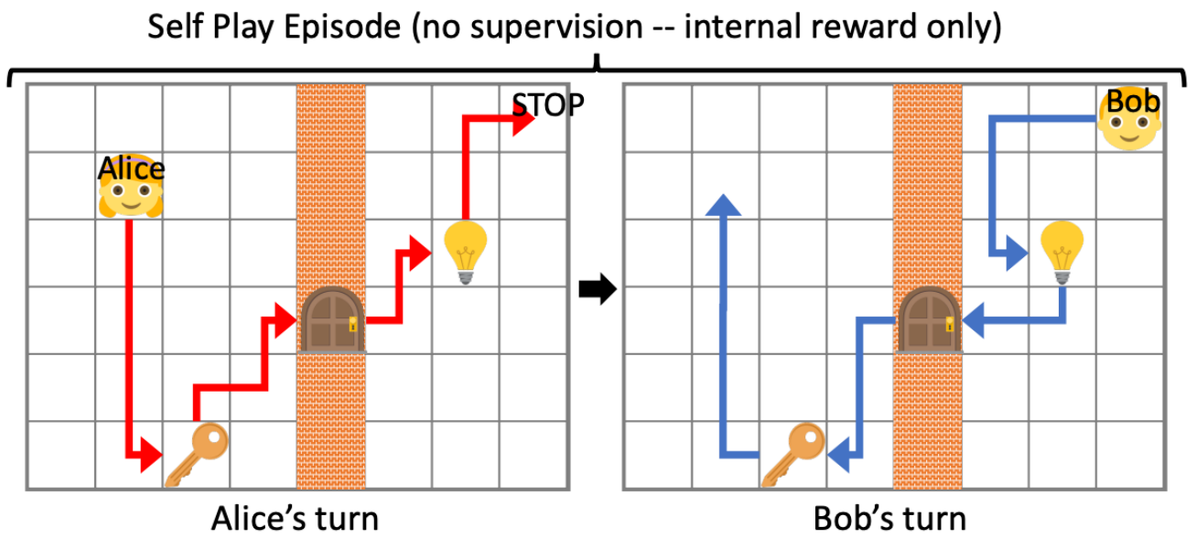
$$\mathscr{D} \rightarrow \{\mathscr{D}_{train}, \mathscr{D}_{test}\}$$

- Natural approach: train with RL directly on $\mathscr{D}_{train}$

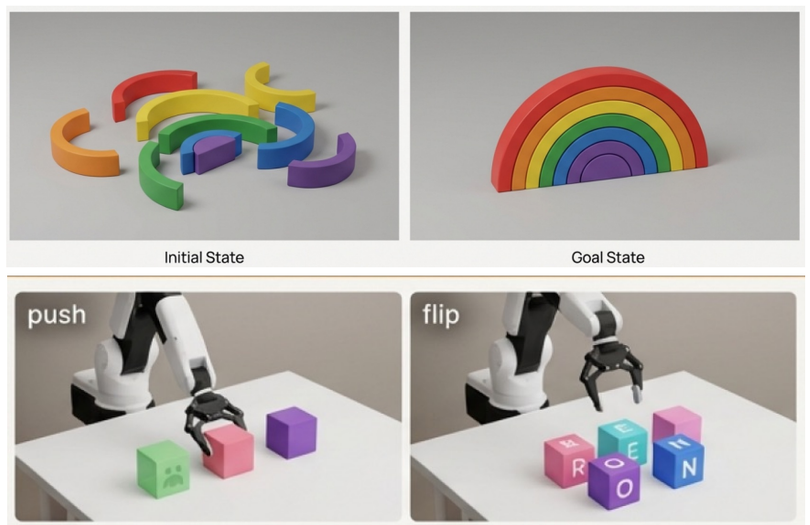
Doesn't work!

Asymmetric Self-Play

Powerful engine for **skill-discovery** and **exploration**



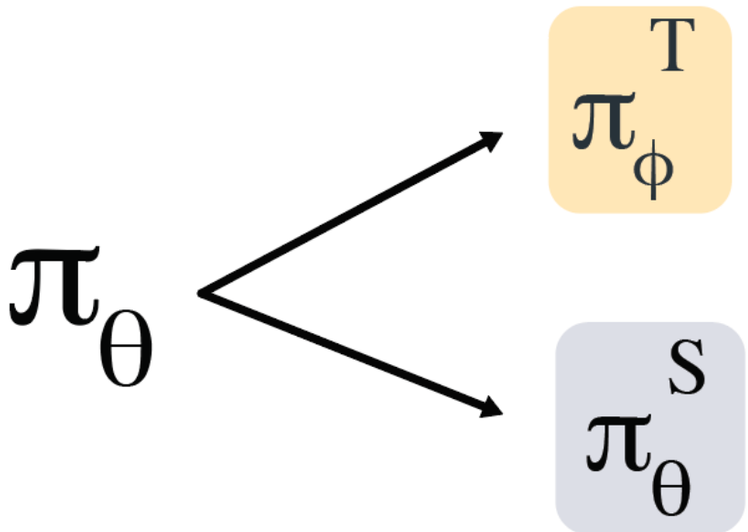
Sukhbaatar et al. Intrinsic Motivation and Automatic Curricula via Asymmetric Self-Play. 2017



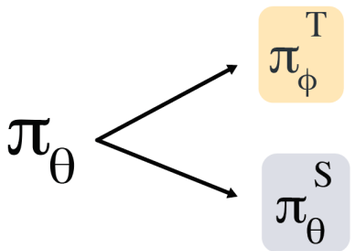
OpenAI. Asymmetric Self-Play for Automatic Goal Discovery… 2021.

Useful way to explore self-generated curricula!

Bilevel Optimization



Bilevel Optimization



Teacher Goal:

Generate a small synthetic dataset $\mathcal{X} = \{(q_i, a_i)\}_{i=1}^n$ s.t. training π_θ^S on $\mathcal{X}$ with RL improves performance on the target domain.

Bilevel Optimization

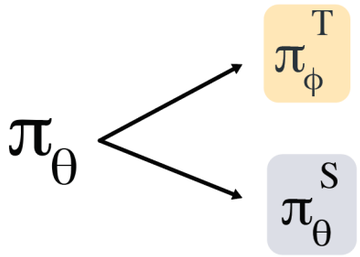
Updated student performance

$$\max_{\phi} \quad \mathbb{E}_{\mathcal{X} \sim \pi_{\phi}^{T}} \left[R\left(\pi^{S}_{\theta'(\mathcal{X})}, \mathcal{D}_{train} \right) \right]$$

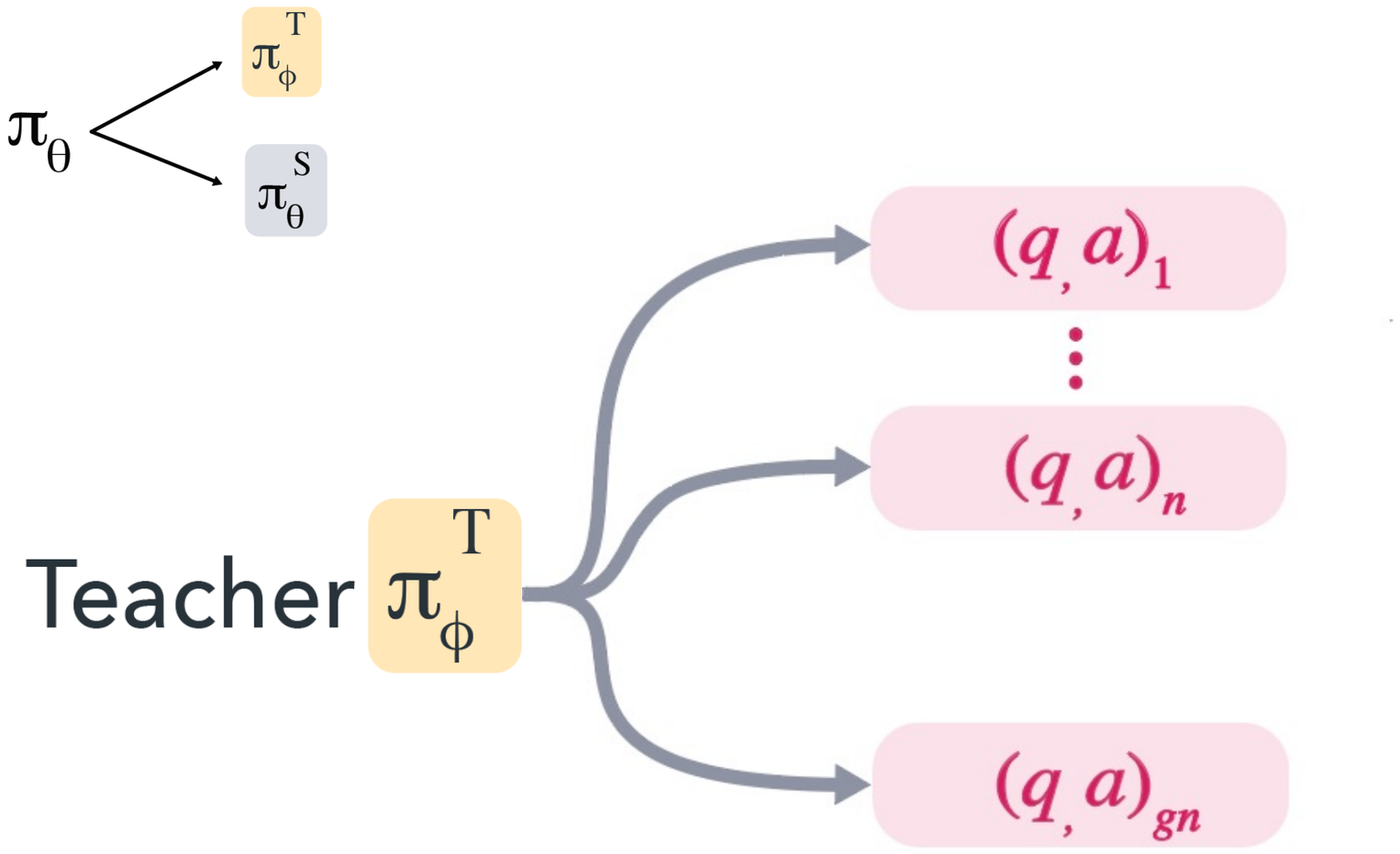
$$\text{subject to} \quad \theta'(\mathcal{X}) = \text{RL-UPDATE}(\theta, \mathcal{X}),$$

RL training of student on X

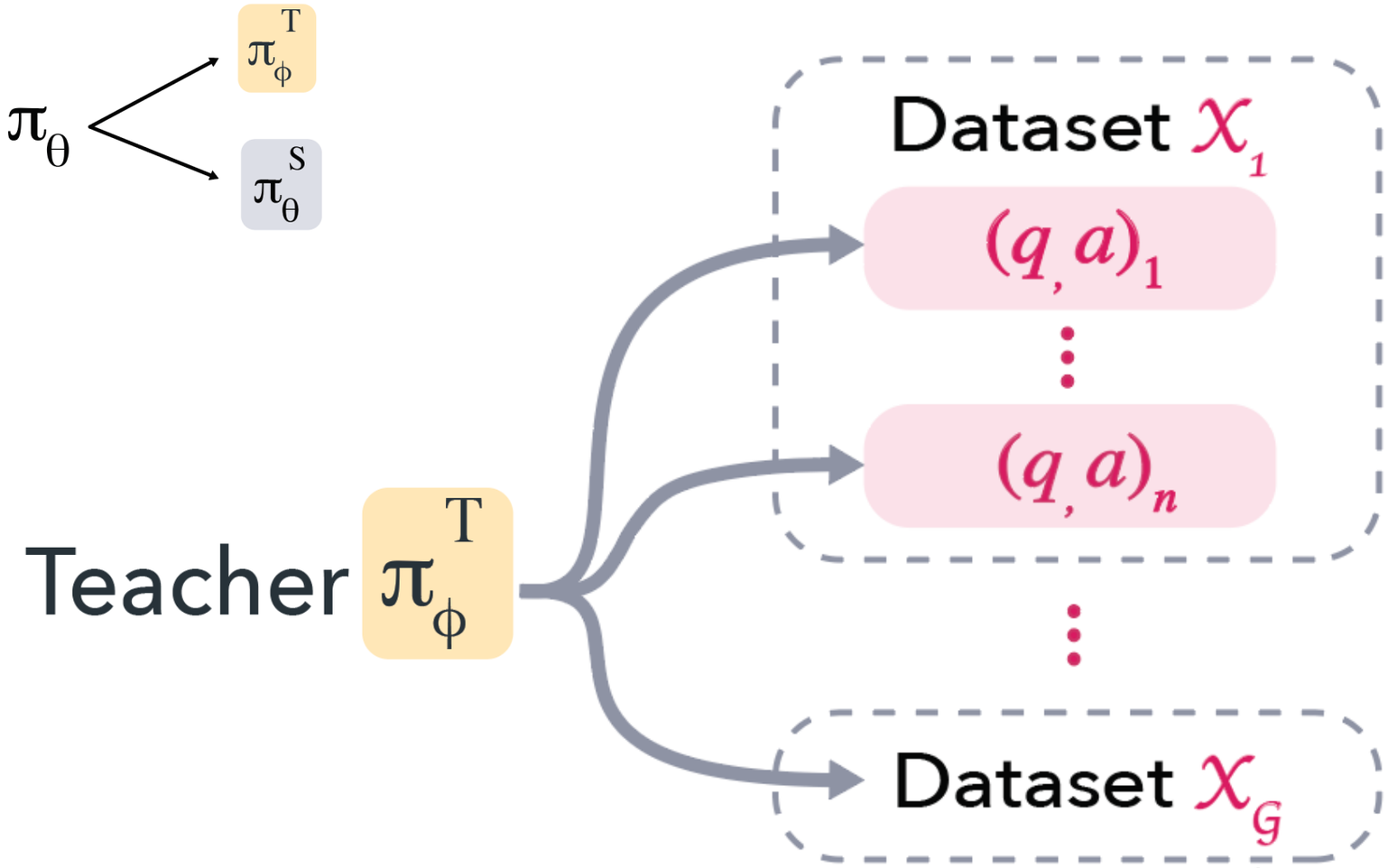
The teacher generates synthetic (question, answer) pairs such that training the student on them improves its performance on the difficult domain



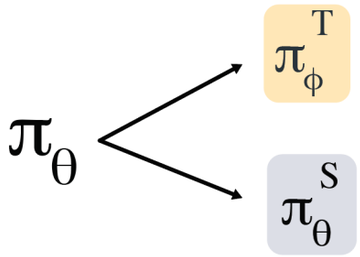
Teacher π_ϕ^T

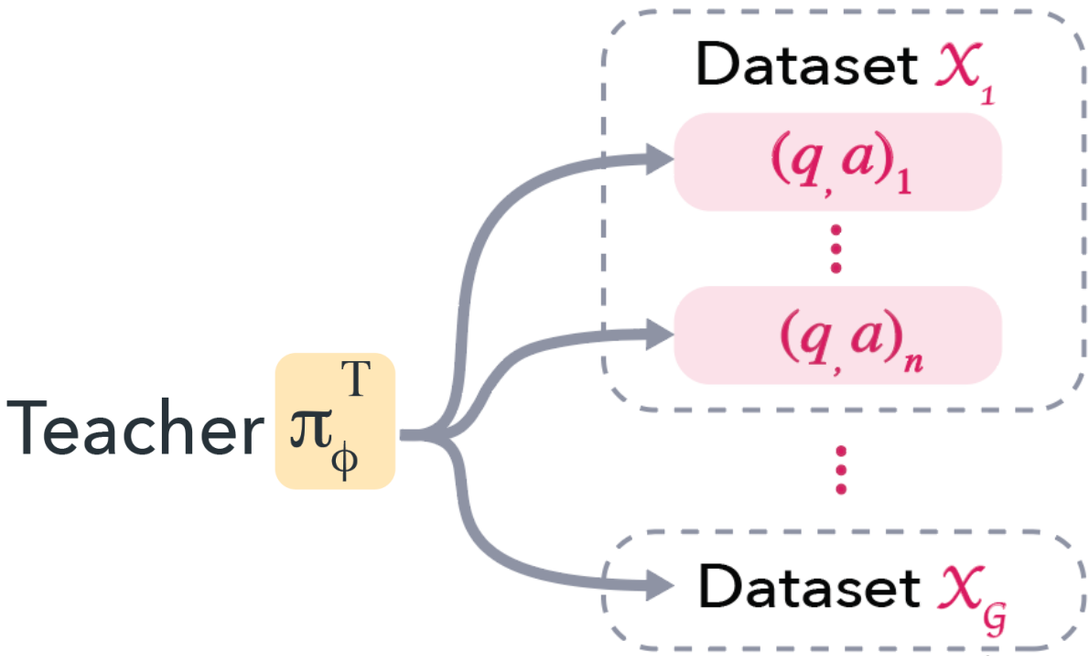


Outer loop – Teacher generates candidate (q, a) pairs for student

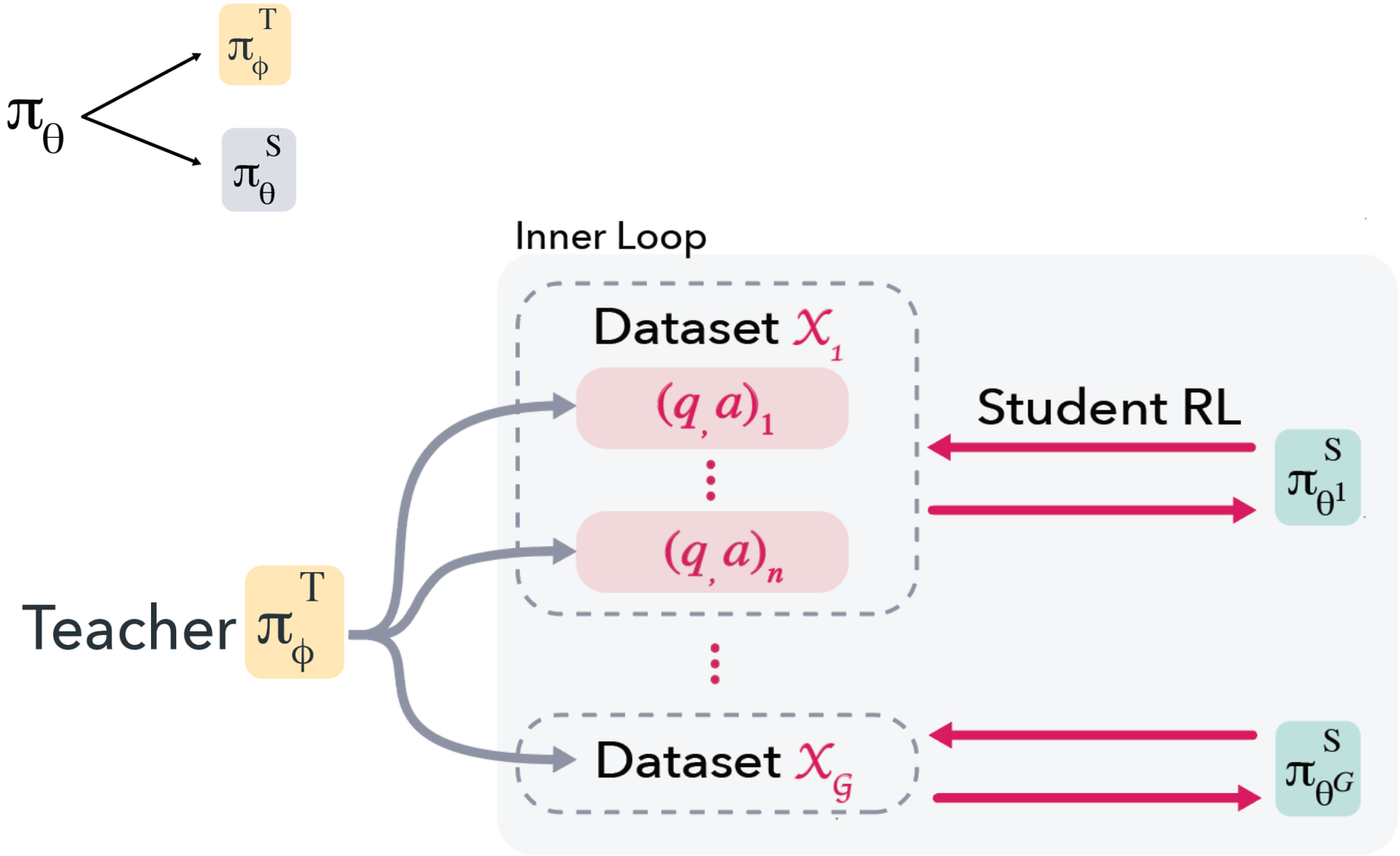


Outer loop – Teacher generates candidate (q, a) pairs for student



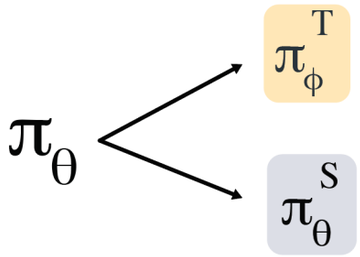


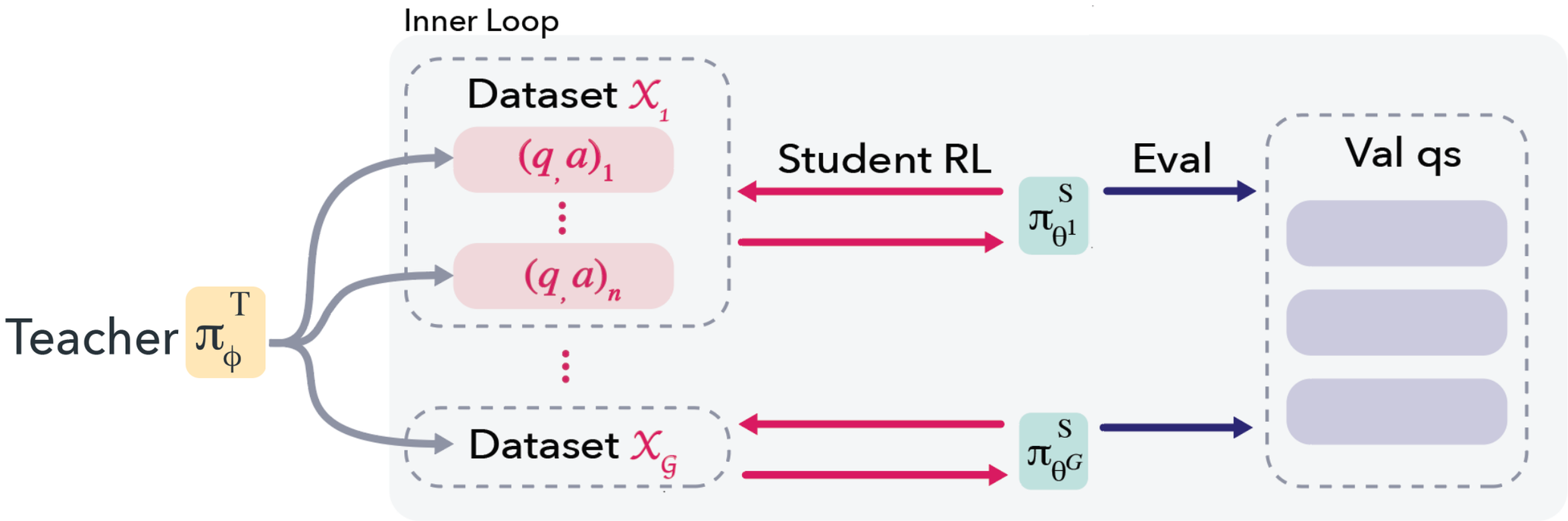
Outer loop – Teacher generates candidate (q, a) pairs for student



Outer loop – Teacher generates candidate (q, a) pairs for student

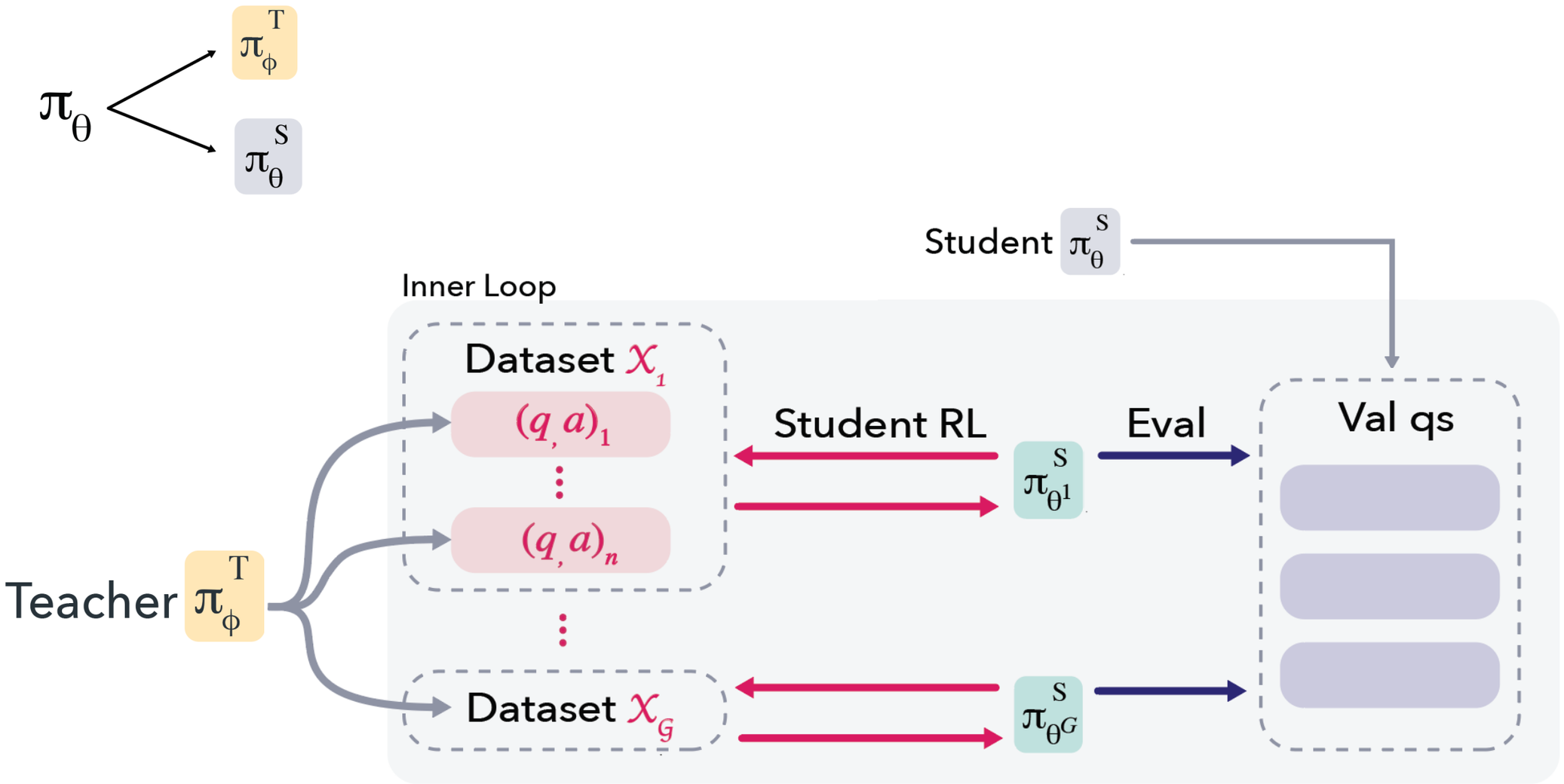
Inner loop – Student trained w/ REINFORCE-style alg for a few steps, then evaluates





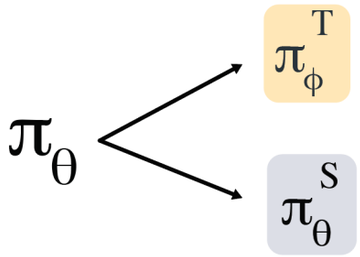
Outer loop – Teacher generates candidate (q, a) pairs for student

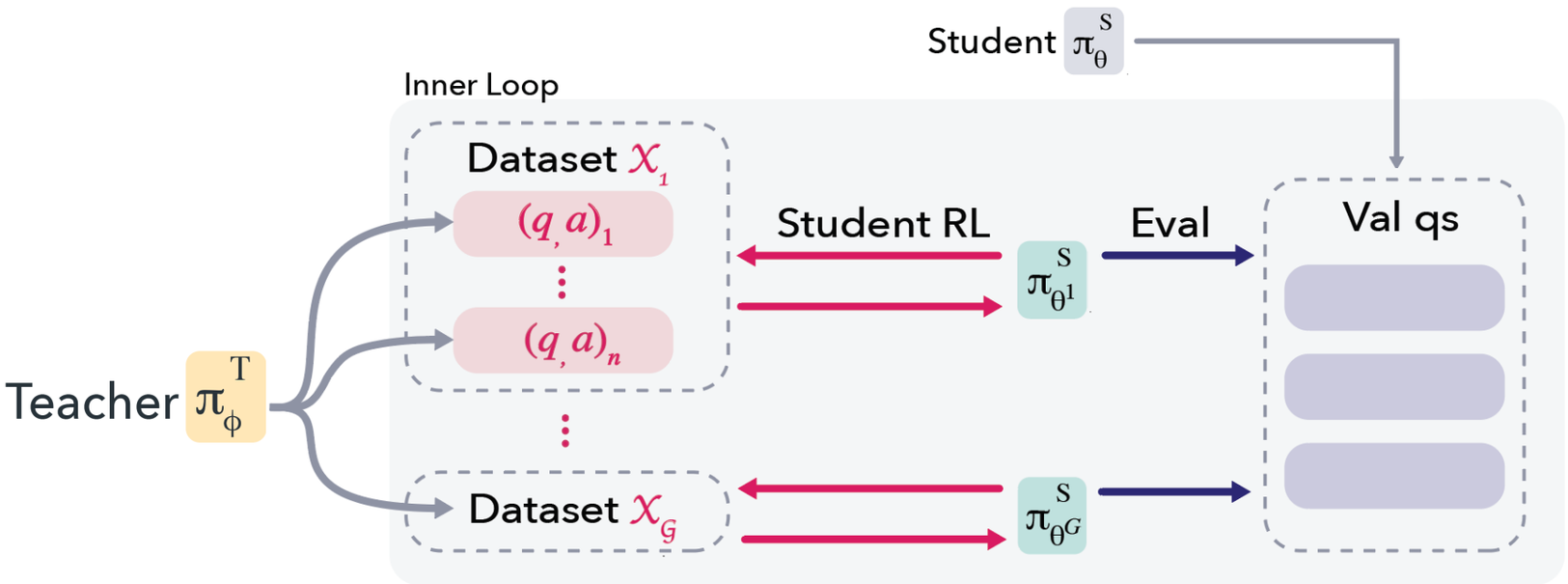
Inner loop – Student trained w/ REINFORCE-style alg for a few steps, then evaluates



Outer loop – Teacher generates candidate (q, a) pairs for student

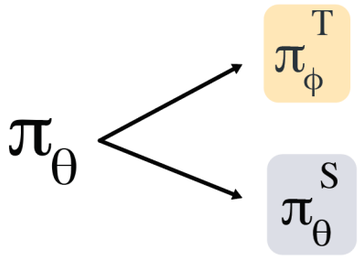
Inner loop – Student trained w/ REINFORCE-style alg for a few steps, then evaluates



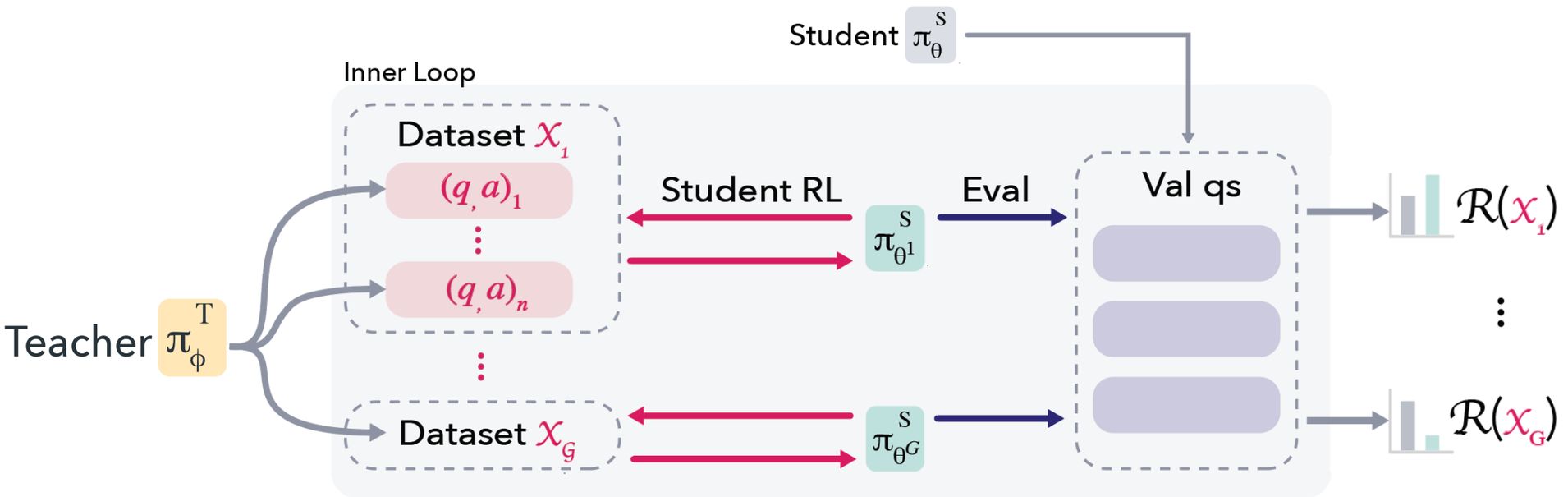


Outer loop – Teacher generates candidate (q, a) pairs for student

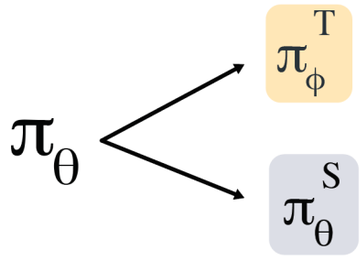
Inner loop – Student trained w/ REINFORCE-style alg for a few steps, then evaluates



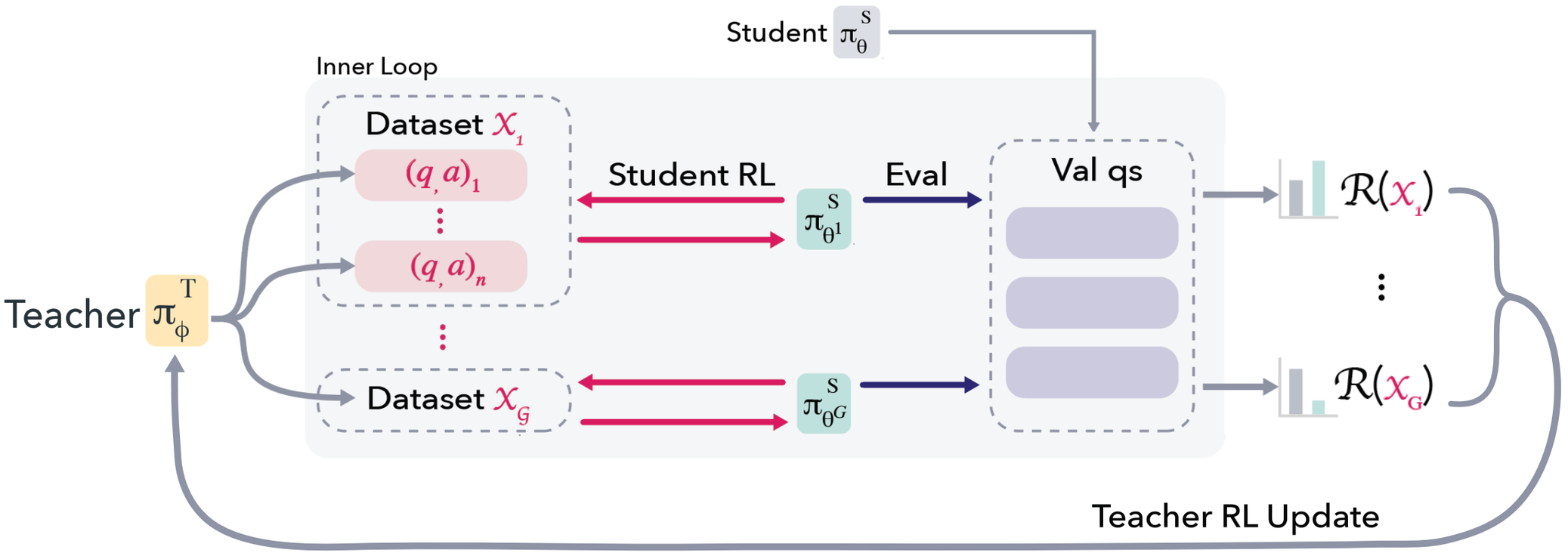
* Teacher gets rewarded only if the student gets better at the *real* task

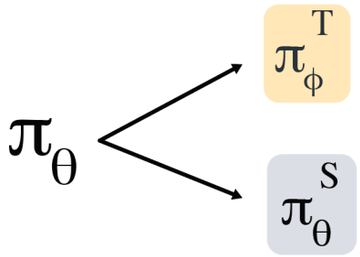


$$\mathcal{R}(\mathcal{X}_k) = \mathrm{Acc}(\pi^S_{\theta'_k}(\mathcal{Q}_R)) - \mathrm{Acc}(\pi^S_\theta(\mathcal{Q}_R))$$



* Teacher gets rewarded only if the student gets better at the *real* task





Can the teacher adapt to an improving student?

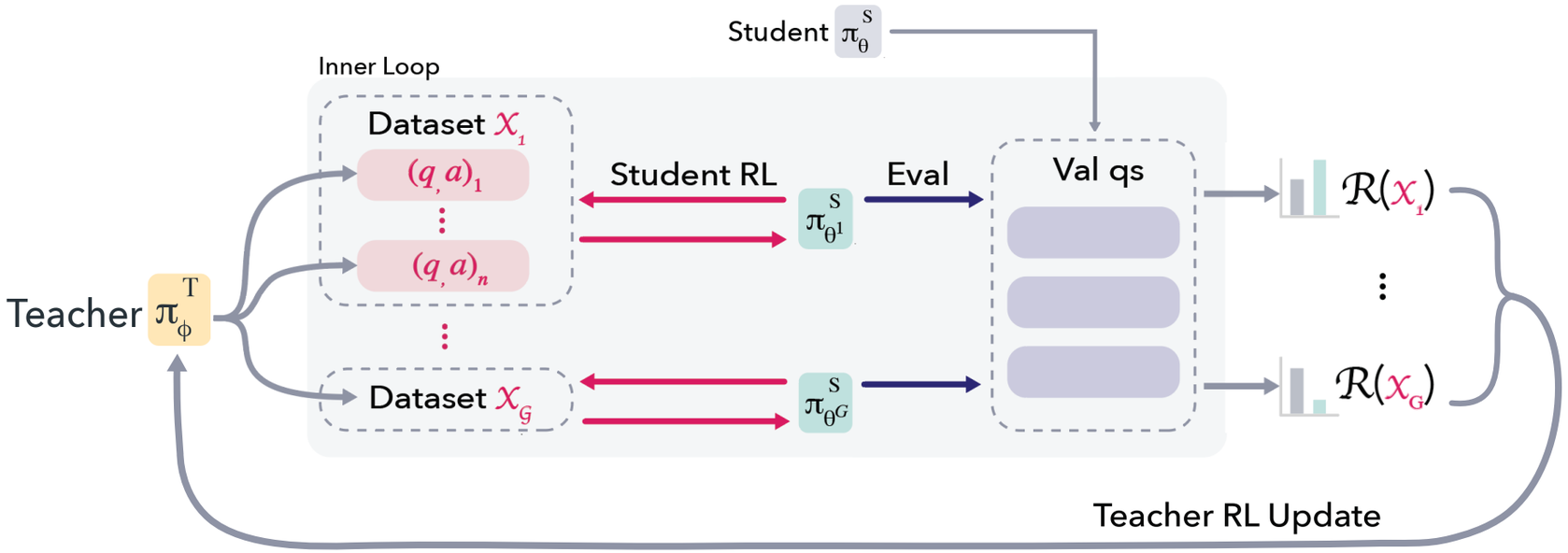
- Track moving average of teacher rewards $\bar{R}_t$
- If $\bar{R}_t > \mathcal{T}$, **promote** the student baseline:

$$\theta \leftarrow \theta'_{k^*}$$

(k^* is the index of the best dataset at that step)

- Accumulate questions that led to promotions

$$\mathcal{D}_{best}$$



Grounded v. intrinsic rewards

- **Intrinsic rewards –** Common in self-play (learnability, self-consistency, etc)
  - **Learnability** – Reward problems with student pass rate of 50%
  - Risks reward-hacking, degenerate outputs, drift

- **Grounded rewards** – Only rewards (q, a) pairs that improve student performance
  - **Black box signal** that tethers curriculum to learning progress

Experiment Setup

Model: Llama-3.2-3B-Instruct

Hard Dataset: Filter from MATH or HARP for failure to pass @128 (OlympiadBench held-out)

Evaluate:

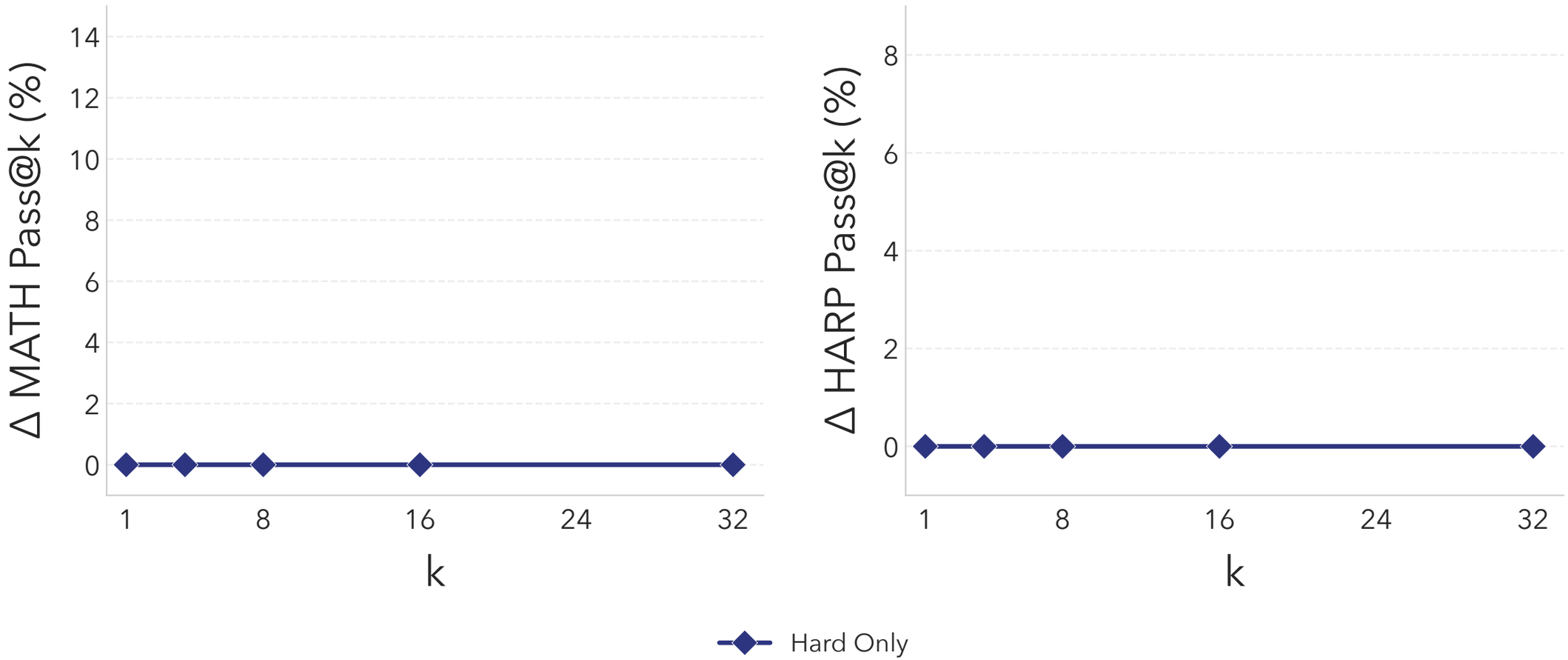
- Promotion questions – Accumulated $\mathscr{D}_{best}$
- Promoted student – Updated student at the end of meta-RL loop
- Questions sampled from a teacher trained with learnability

Roadmap of experiments

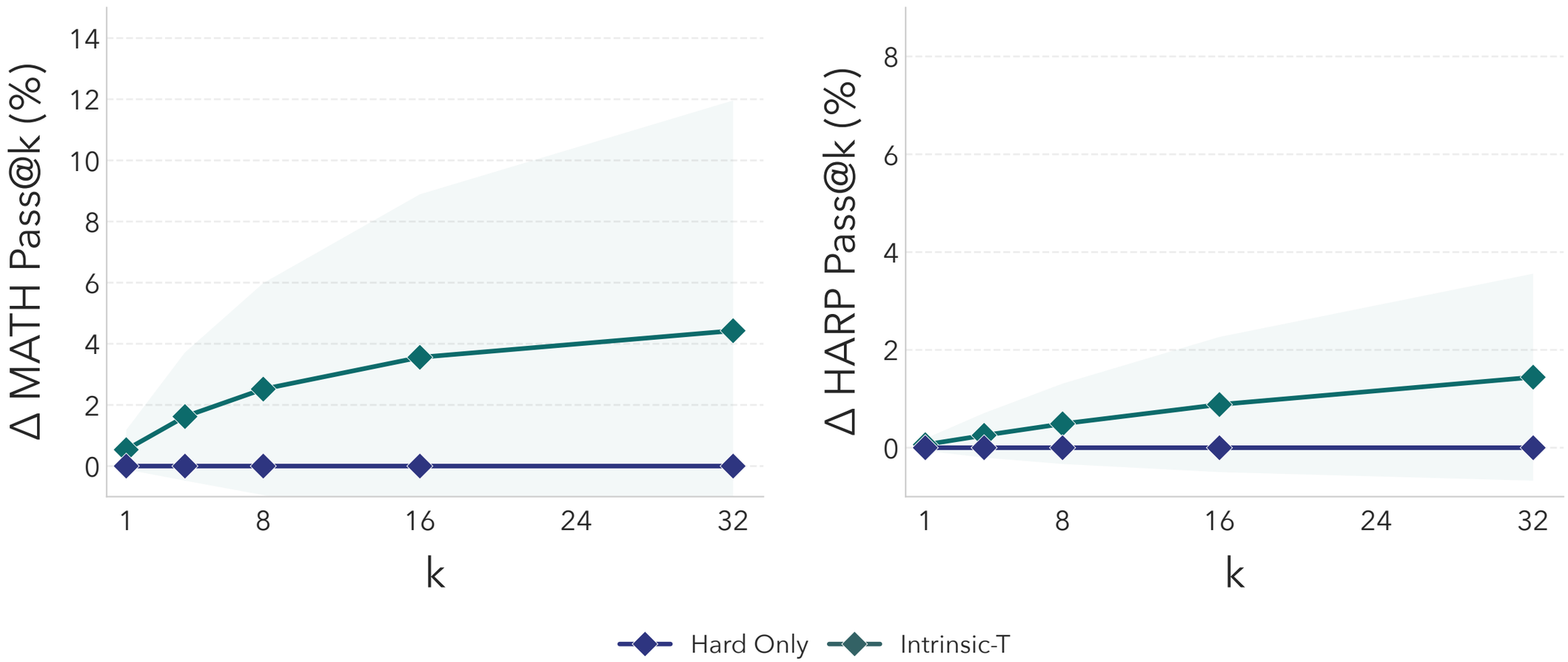
1. Decoupled teaching and solving (meta-RL works)
2. Grounded rewards > intrinsic rewards
3. Question structure over solution correctness

Meta-RL finds *effective questions*

Meta-RL questions kickstart learning

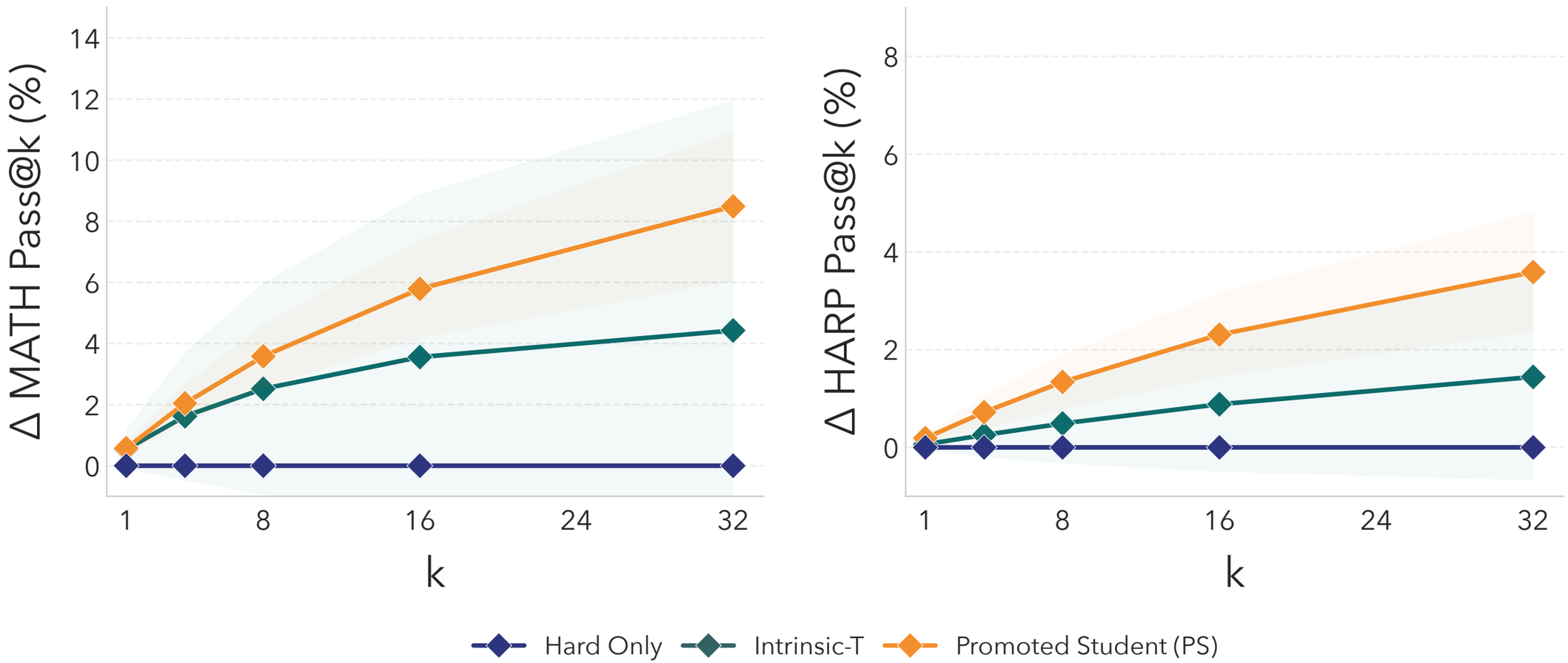


Meta-RL questions kickstart learning



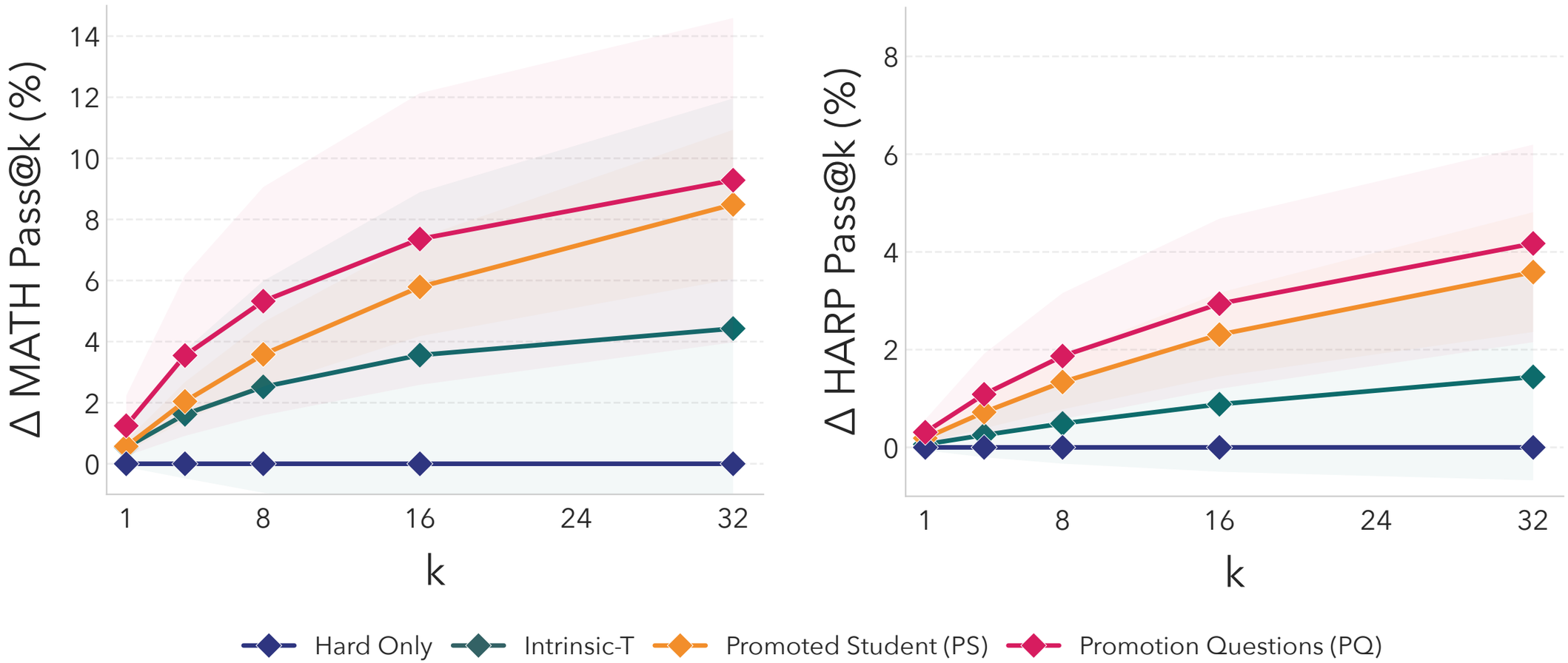
* Sample 128 questions from a teacher trained with an intrinsic reward (*Intrinsic-T*) and train on the sampled questions + fail@128 train set

Meta-RL questions kickstart learning



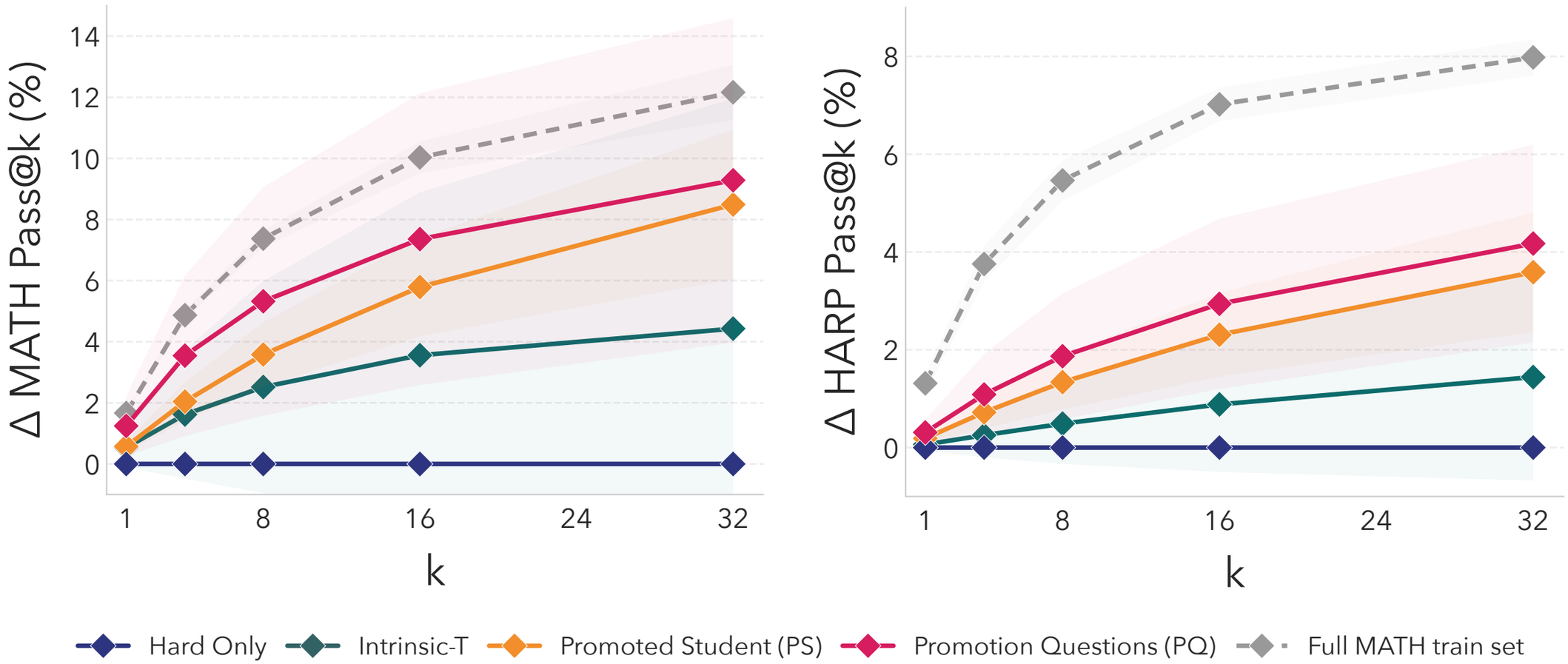
* Direct inference with promoted student

Meta-RL questions kickstart learning



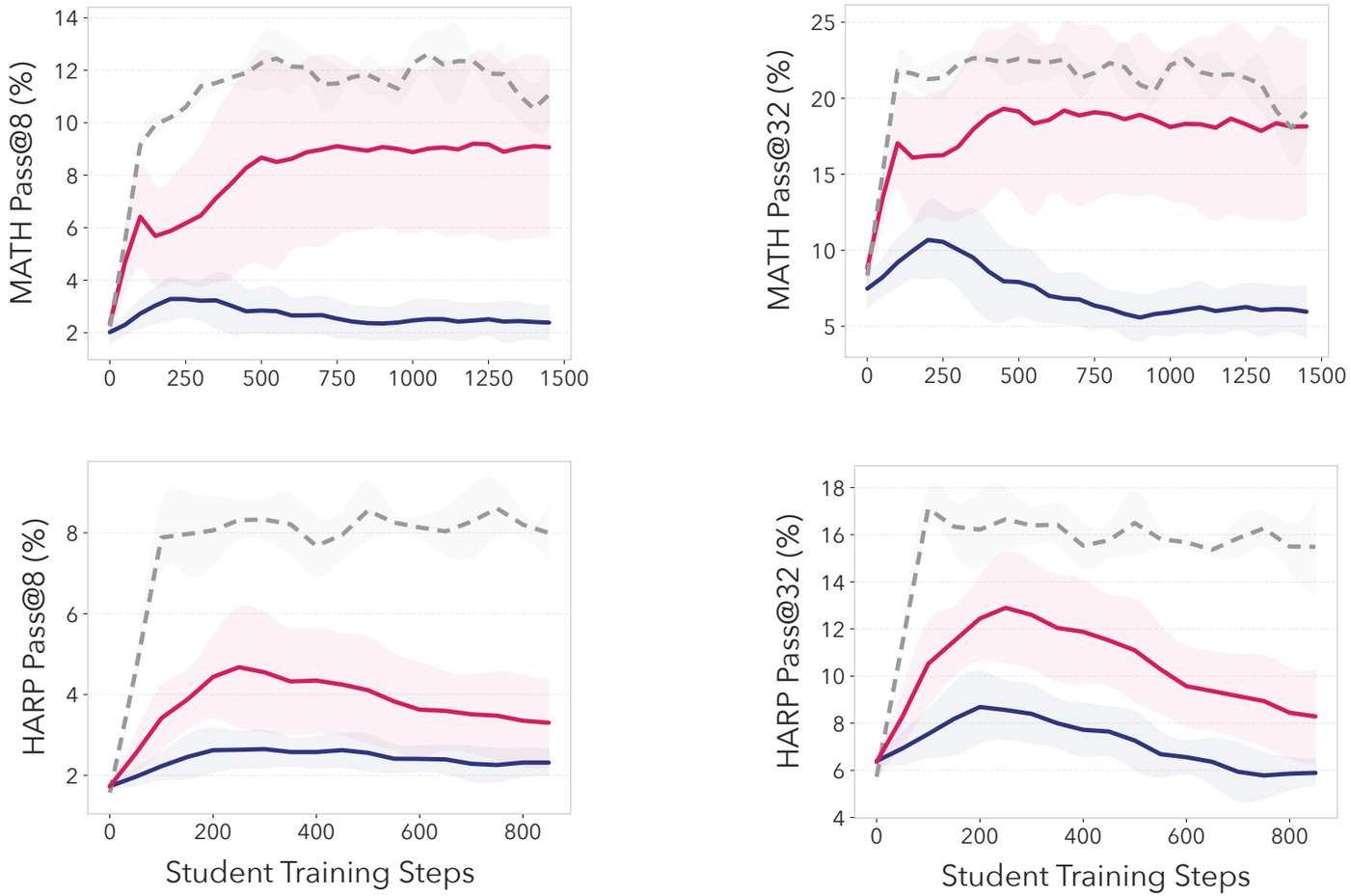
* Train on promotion questions + fail@128 train set

Meta-RL questions kickstart learning

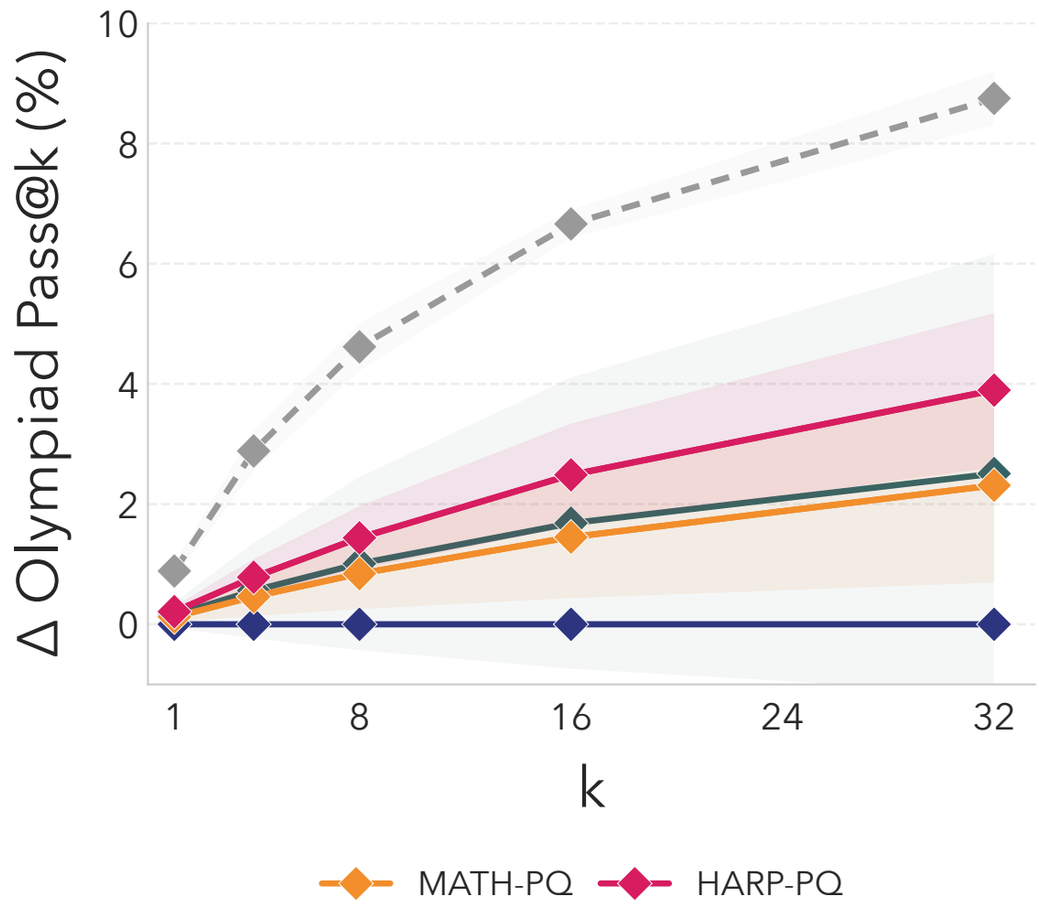


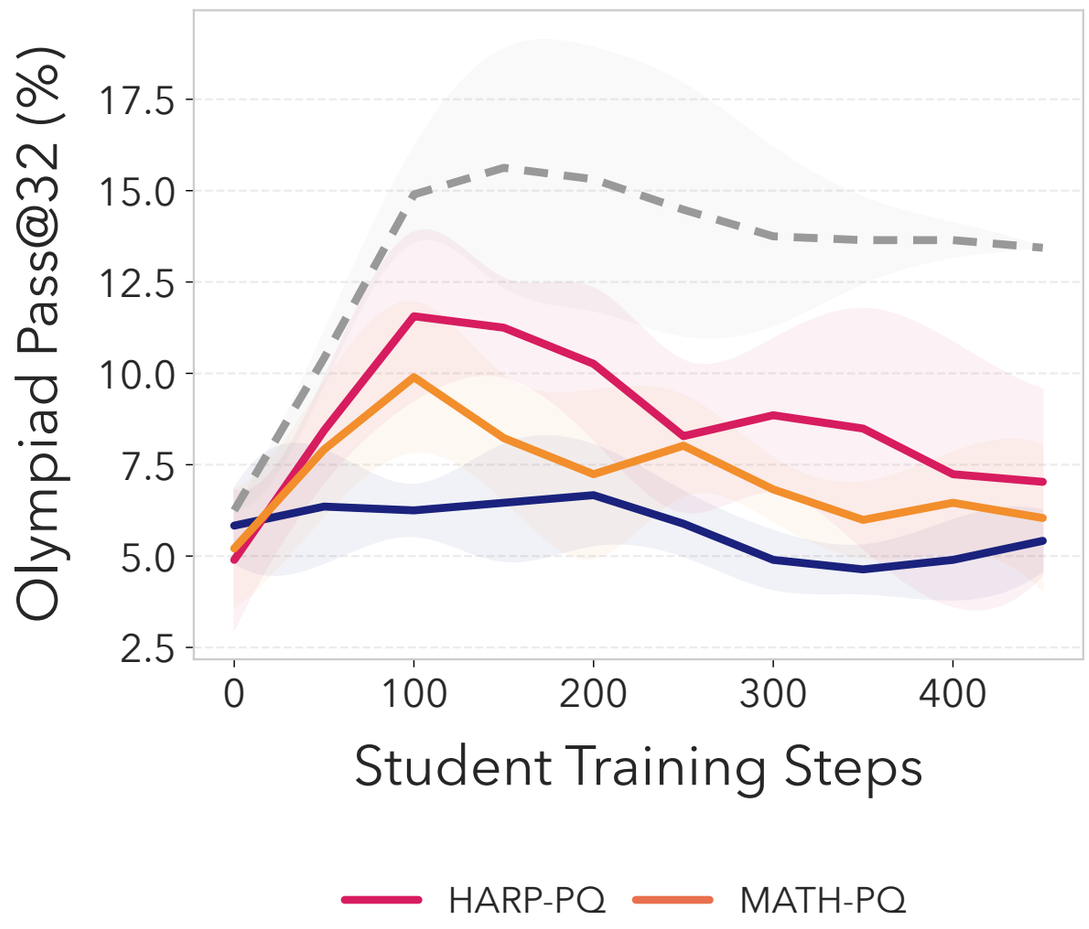
* Train on the full MATH train set + fail@128 train set

Meta-RL questions kickstart learning



OOD Generalization





Decoupled teaching and solving

Inference with the trained teacher policy doesn’t improve!

Finding 1: A model's *pedagogical* ability can be decoupled from its *task-solving ability*.

Grounded meta-RL (SOAR) expands the "learnability frontier" by surfacing synthetic questions that enable improvement over reasoning plateaus.

Grounded rewards lead to *better teachers*

Studying the trained teacher policy

Setup: Sample 128 questions from trained teacher, train on sampled qs + fail@128 train set

Teachers:

- Base model (Base-T)
- Intrinsic-trained model (Intrinsic-T)
- SOAR-trained model (Grounded-T)

Does the trained teacher generate useful questions?

MATH Pass@k (%) Test Accuracy on Fail@128

	k				
Method	**1**	**4**	**8**	**16**	**32**
Base Model Inference	0.3 ± 0.1	1.0 ± 0.2	2.0 ± 0.4	3.9 ± 0.8	7.5 ± 1.3
Hard-Only	0.5 ± 0.1	1.7 ± 0.4	3.2 ± 0.8	5.7 ± 1.5	9.6 ± 2.6
Hard-Only ($g = 128$)	1.4 ± 1.0	3.9 ± 2.6	6.1 ± 3.9	8.9 ± 5.5	12.4 ± 7.4
SOAR-PQ (Ours)	$\mathbf{1.7 \pm 1.0}$	$\mathbf{5.3 \pm 2.6}$	$\mathbf{8.5 \pm 3.7}$	13.0 ± 4.8	18.9 ± 5.3
SOAR-PS (Ours)	1.0 ± 0.2	3.8 ± 0.6	6.8 ± 1.1	11.5 ± 1.6	18.1 ± 2.4

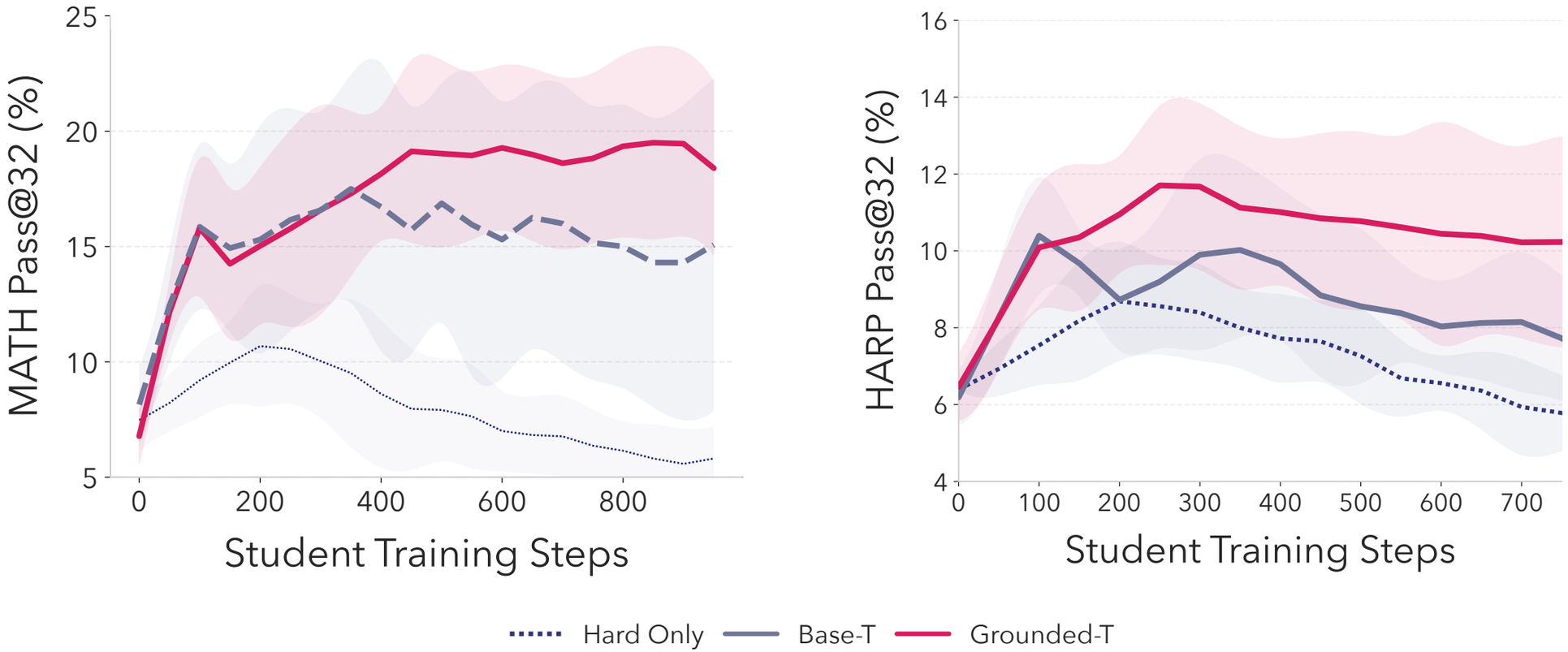
The trained teacher generate useful questions

MATH Pass@k (%) Test Accuracy on Fail@128

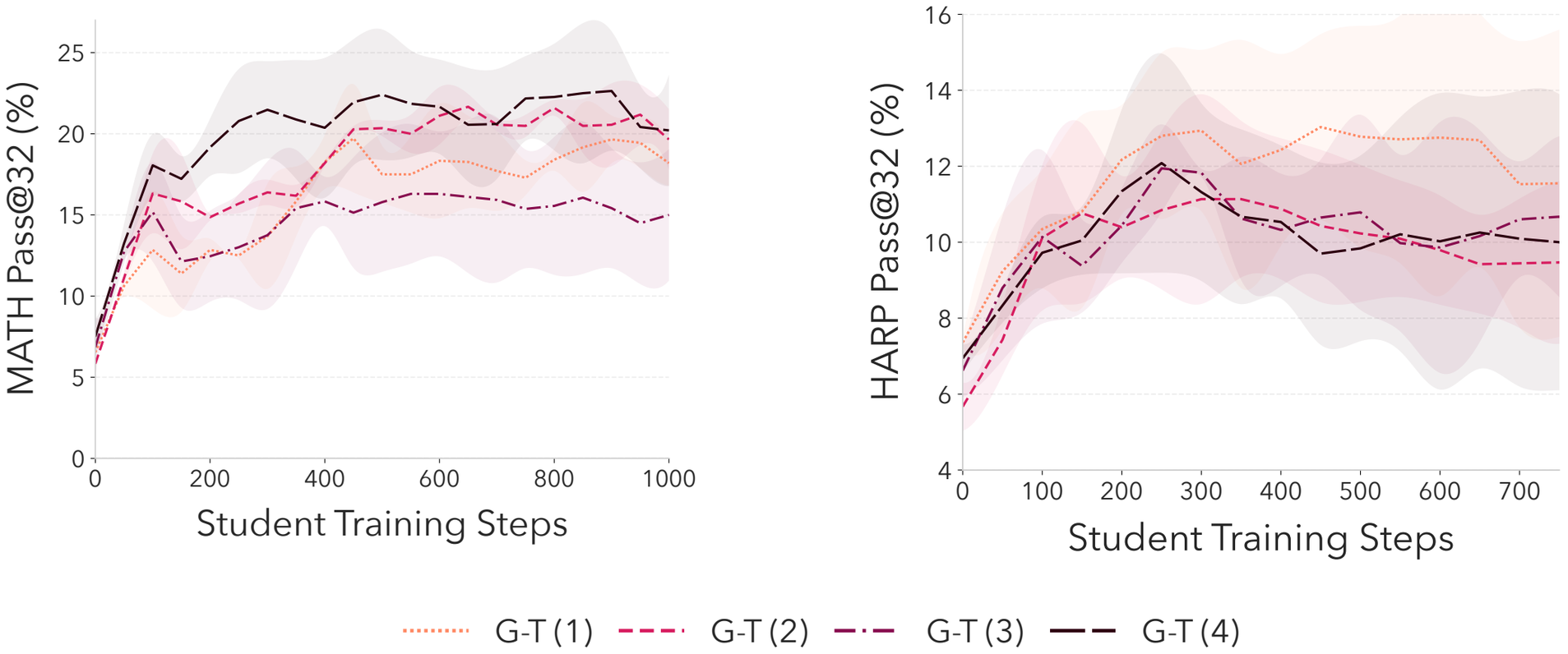
	k				
Method	**1**	**4**	**8**	**16**	**32**
Base Model Inference	0.3 ± 0.1	1.0 ± 0.2	2.0 ± 0.4	3.9 ± 0.8	7.5 ± 1.3
Hard-Only	0.5 ± 0.1	1.7 ± 0.4	3.2 ± 0.8	5.7 ± 1.5	9.6 ± 2.6
Hard-Only ($g = 128$)	1.4 ± 1.0	3.9 ± 2.6	6.1 ± 3.9	8.9 ± 5.5	12.4 ± 7.4
SOAR-PQ (Ours)	$\mathbf{1.7 \pm 1.0}$	$\mathbf{5.3 \pm 2.6}$	$\mathbf{8.5 \pm 3.7}$	13.0 ± 4.8	18.9 ± 5.3
SOAR-PS (Ours)	1.0 ± 0.2	3.8 ± 0.6	6.8 ± 1.1	11.5 ± 1.6	18.1 ± 2.4
Grounded-T (Ours)	1.6 ± 0.5	5.1 ± 1.4	8.4 ± 2.1	$\mathbf{13.1 \pm 2.9}$	$\mathbf{19.1 \pm 3.7}$

Same true for HARP/Olympiad! (tables in paper)

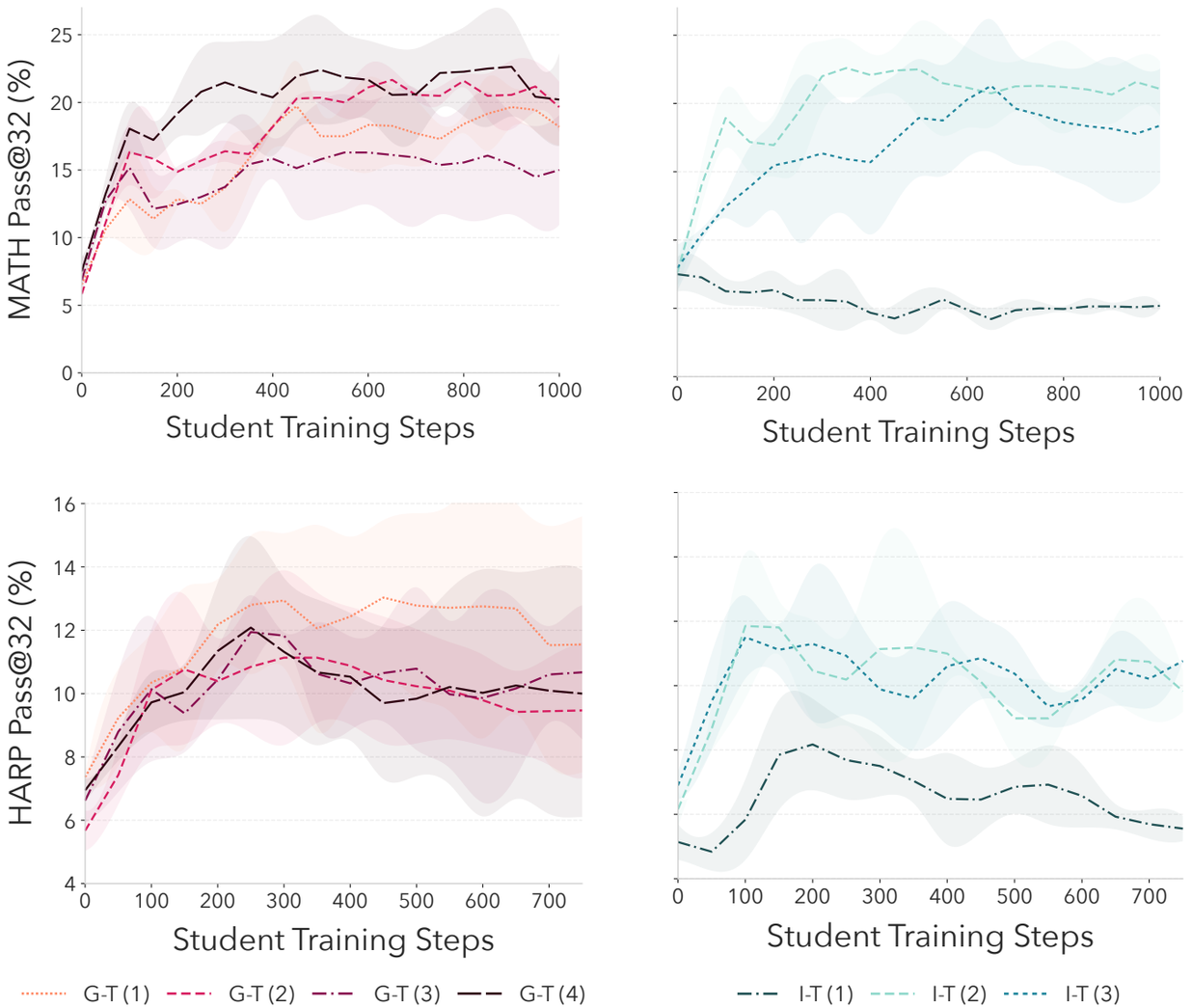
Grounded rewards sharpen the question distribution



Intrinsic rewards lead to less stable teacher policies



Intrinsic rewards lead to less stable teacher policies



Grounded rewards sustain diversity

Vendi score of Qwen3-8B embeddings for sampled questions

Method	**Vendi Score (VS)**	**Std. Dev (σ)**
Base-T	**34.91**	**1.74**
Grounded-T (HARP)	34.66	1.74
Grounded-T (MATH)	31.99	1.54
PQ	28.33	1.55
Intrinsic-T	10.82	1.01

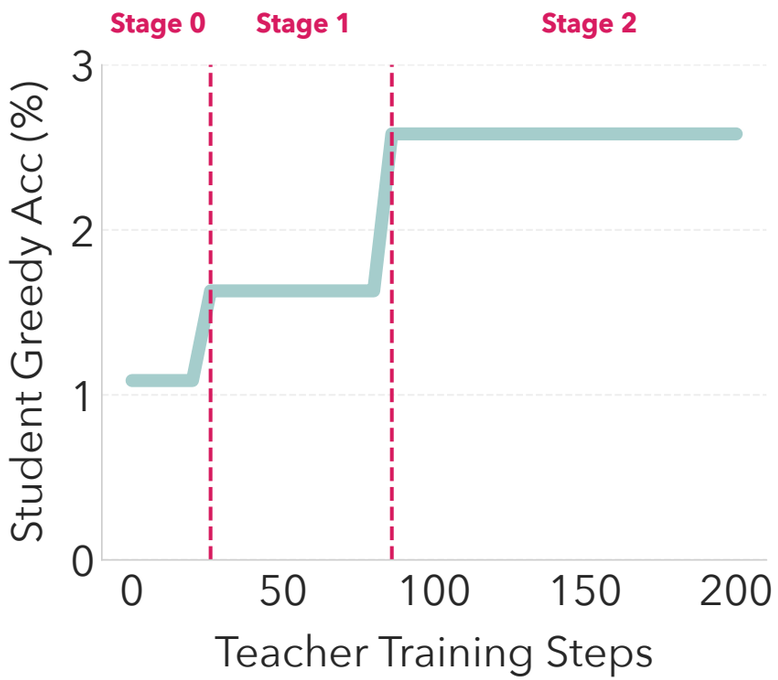
Much lower for Intrinsic-T!

Finding 2: Effective questions are latent in the base model, but hard to find.

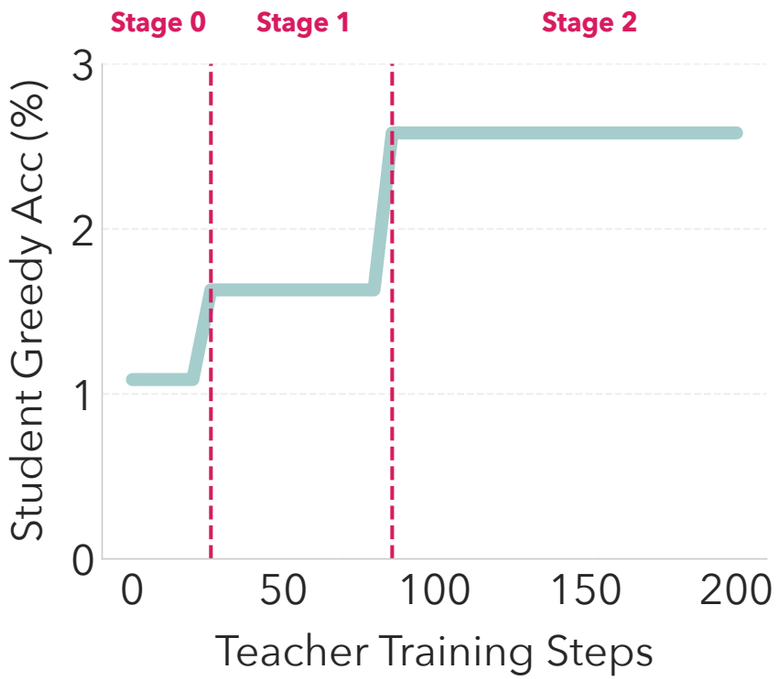
Grounding rewards in student progress "sharpens" the teacher’s noisy distribution of questions into a stable, diversity preserving policy, whereas intrinsic rewards are prone to instability and diversity collapse.

Question structure over solution correctness

Questions adapt over learning stages



Questions adapt over learning stages



Stage 1

Q: Tom has a rectangular garden with a length of 15 meters and a width of 8 meters. The cost of buying a tarp to cover the garden is 10.50 per square meter. If Tom buys more than one tarp, what is the least cost he can expect to pay?
A: 1260

Q: What is the area of the path that is added around a rectangular garden with dimensions 15 meters by 8 meters when a 1-meter-wide garden path is built?
A: 50

Q: What is the probability of randomly picking a blue or green item from a bag containing 8 red, 5 blue, and 5 green items?
A: 5/9

Stage 2

Q: Given the polynomial $f(x) = x^3 - 6x^2 + 11x - 6$, factorize and solve for its roots.
A: (2, 3, 1)

Q: Find the dynamics of the function $f(x) = 2\sin^2(x+3\pi/2) + 3\cos^2(x-2\pi/5)$, where $0 \le x \le 2\pi$.
A: $-\cos^2(x + 3\pi/2) + 3\sin^2(x + 3\pi/2)$

Q: What is the value of x in the equation: $x^3-y^3 = (2x + 3y)y^2$?
A: 3

Q: Compute the value of x for the equation: $f(x) = (x^2 + 1) / (x - 1)$?
A: 2

Questions adapt between stages

They look coherent, but answers aren't necessarily correct!

Questions structure over solution correctness

Category	Base	Intrinsic	Grounded	PQ
Well-Posed	53.6%	63.5%	70.0%	64.6%
Correct	23.2%	55.5%	36.5%	32.8%
Error Taxonomy (% of total samples)				
Arithmetic Error	23.7%	5.7%	29.0%	25.0%
Logic Error	5.7%	2.3%	6.9%	6.5%
Impossibility Error	4.7%	2.9%	8.2%	4.7%
Ambiguity Error	42.4%	33.6%	21.3%	31.3%
Total Samples	384	384	375	384

Questions structure over solution correctness

Category	Base	Intrinsic	Grounded	PQ
Well-Posed	53.6%	63.5%	70.0%	64.6%
Correct	23.2%	55.5%	36.5%	32.8%
Error Taxonomy (% of total samples)				
Arithmetic Error	23.7%	5.7%	29.0%	25.0%
Logic Error	5.7%	2.3%	6.9%	6.5%
Impossibility Error	4.7%	2.9%	8.2%	4.7%
Ambiguity Error	42.4%	33.6%	21.3%	31.3%
Total Samples	384	384	375	384

Well-posedness and correctness both go up, but well-posedness is higher

Intrinsic has higher correctness but performs worse!

Finding 3: For models at learning plateaus, problems that have conceptually diverse and coherent *questions* can provide useful gradient signal even without having precisely correct *answers*.

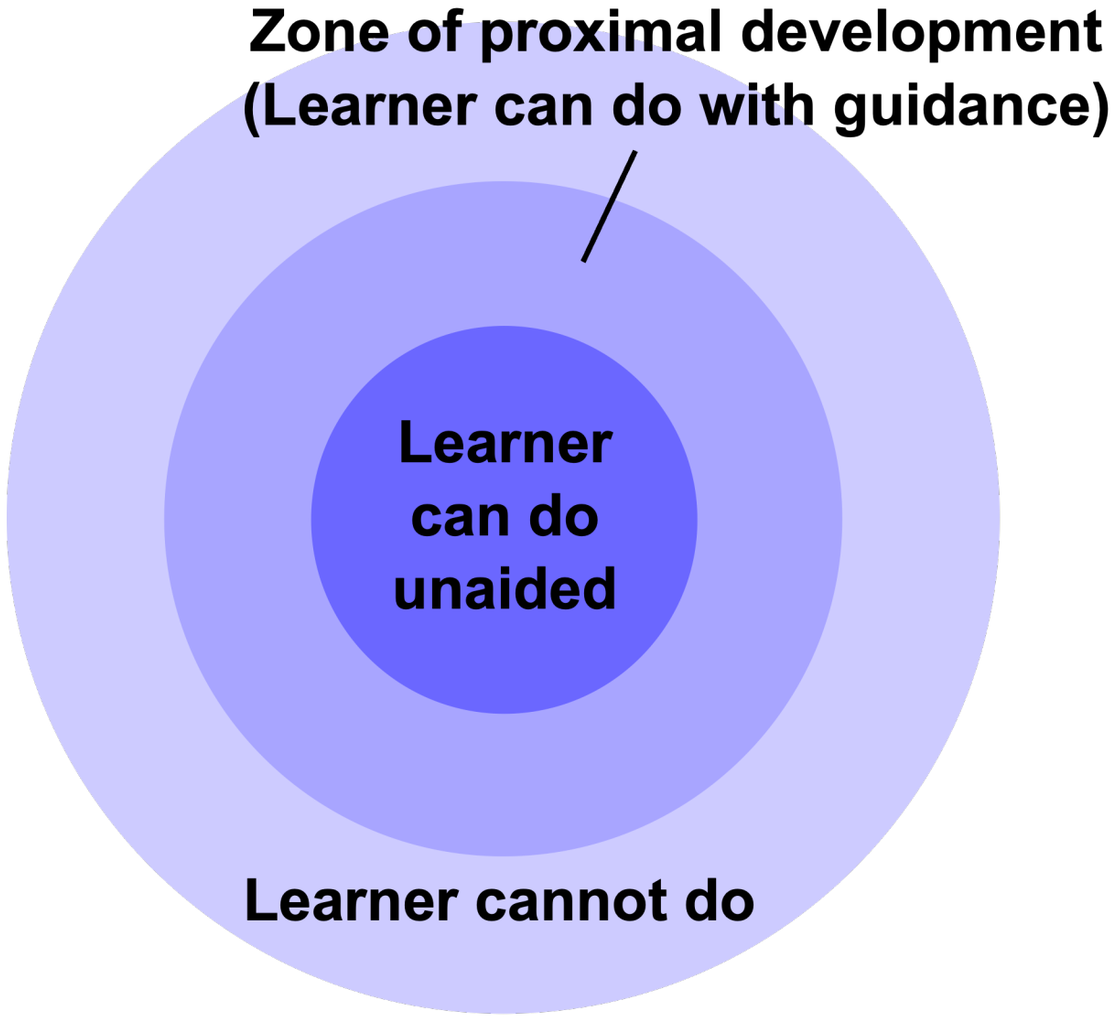
Limitations & future work

- Compute! Bilevel RL is expensive ☹
  - Allocating extra compute to direct training on hard problems doesn't recover the same improvements
- Same trends with bigger models? (some new results on 8B models say yes!)
- Non-verifiable domains? (outside of math/code)

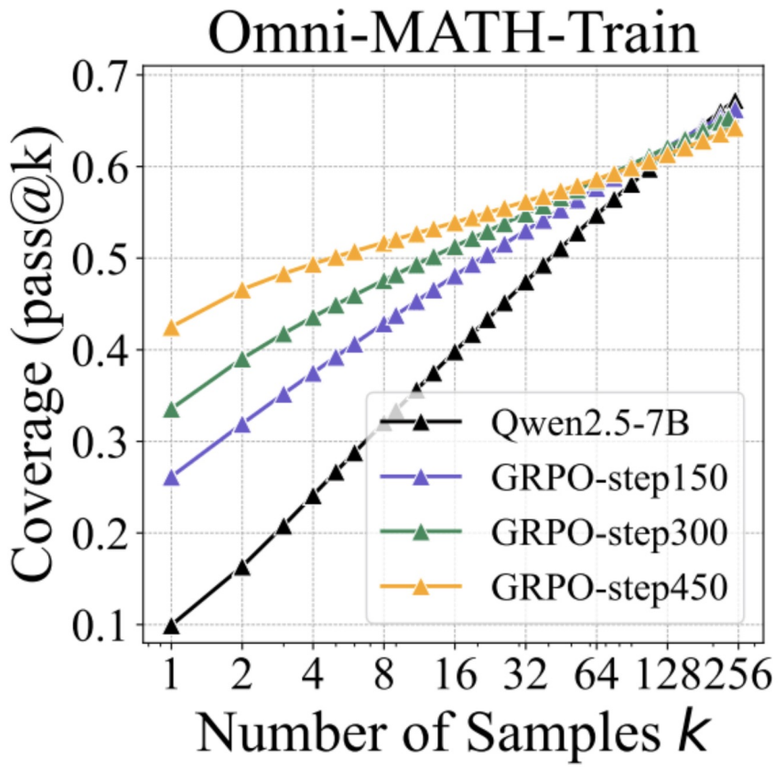
Implications

- Models can be trained to kickstart themselves where direct training fails

- Conceptual takeaways
  - Decoupled pedagogical/solving abilities
  - Grounded > intrinsic rewards
  - Questions structure over correctness


Zone of proximal development
(Learner can do with guidance)
Learner
can do
unaided
Learner cannot do

Broader debate – does RL expand the learnability frontier?



Yue et al. "Does Reinforcement Learning Really Incentivize Reasoning Capacity in LLMs Beyond the Base Model?", NeurIPS'25

Implications

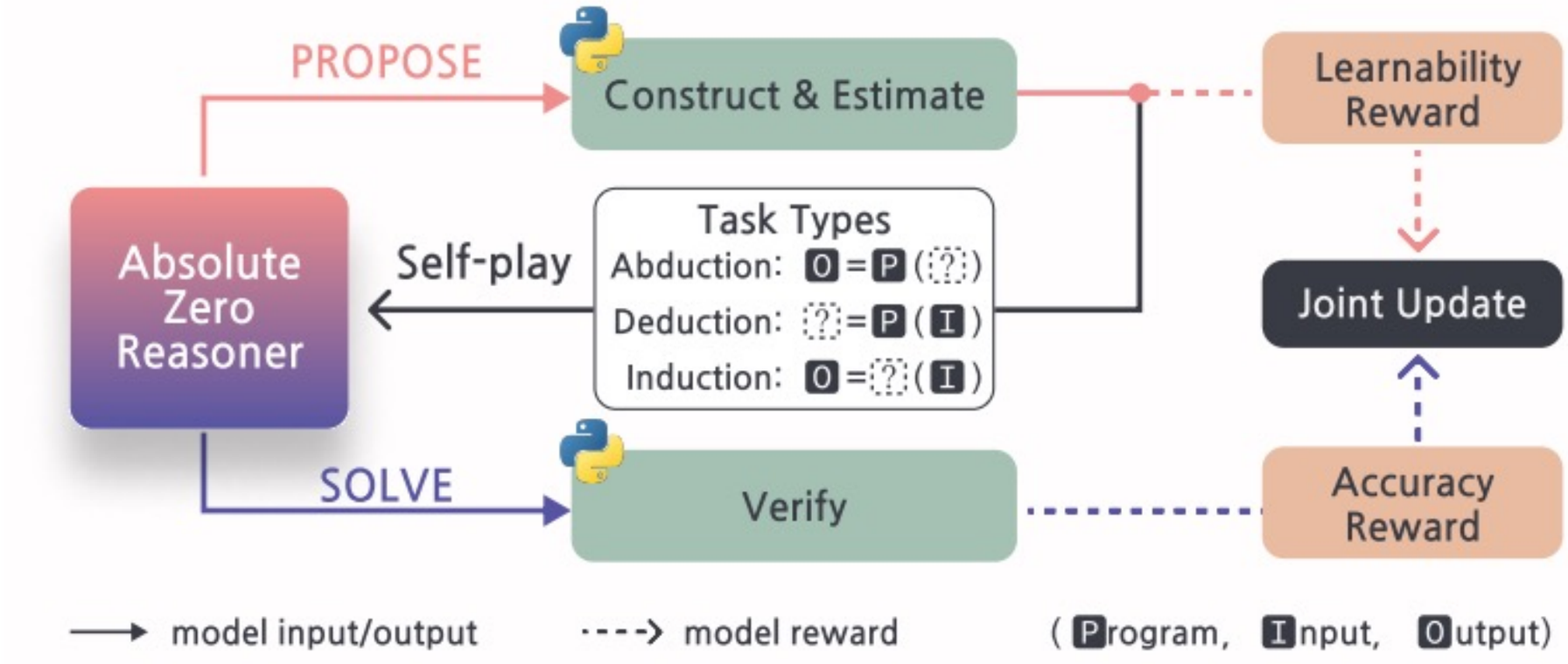
Our work: *Reaching latent knowledge that exists in the pretrained model's distribution, but is inaccessible with normal methods, by sharpening a more easily-accessible ability to generate intermediate questions.*

Other directions I'm excited about!

- How can we get better convergence and guarantees from data-free, asymmetric self-play?
- Towards open-ended self-improving *agents* (beyond model weights)
- Exploring different environments outside of math & coding scaffolds

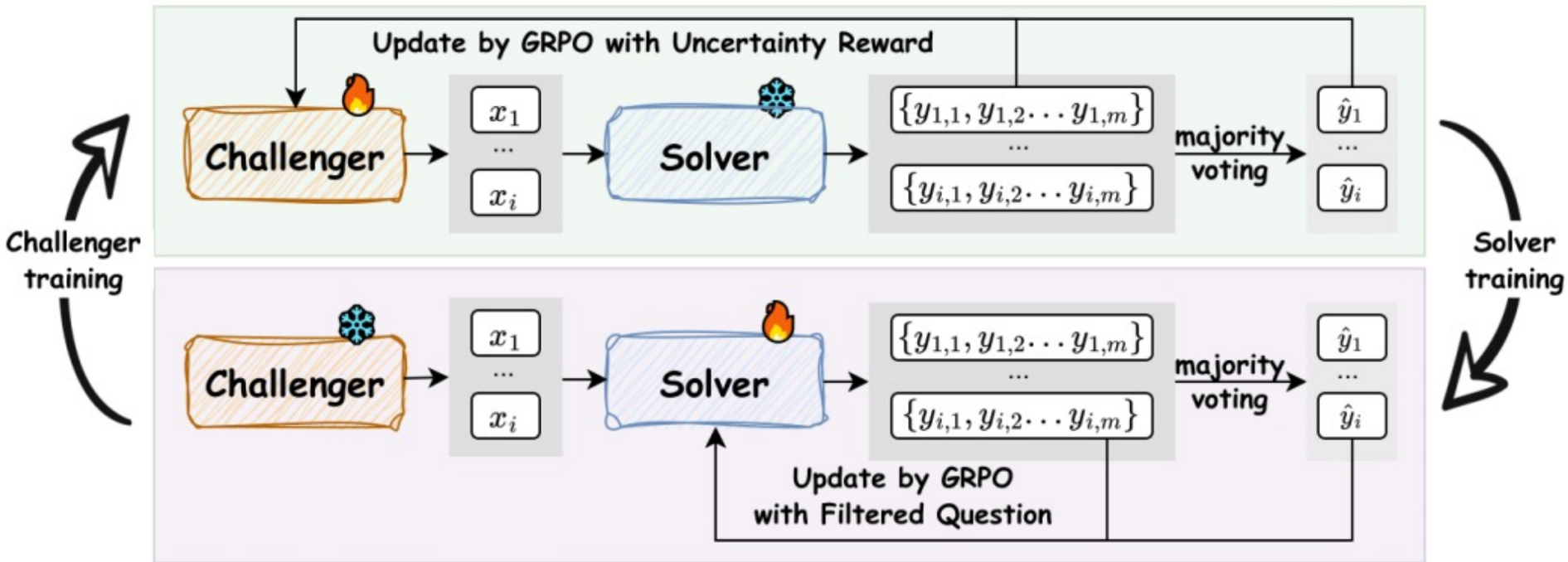
(this is just a small slice!)

Intrinsic exploration



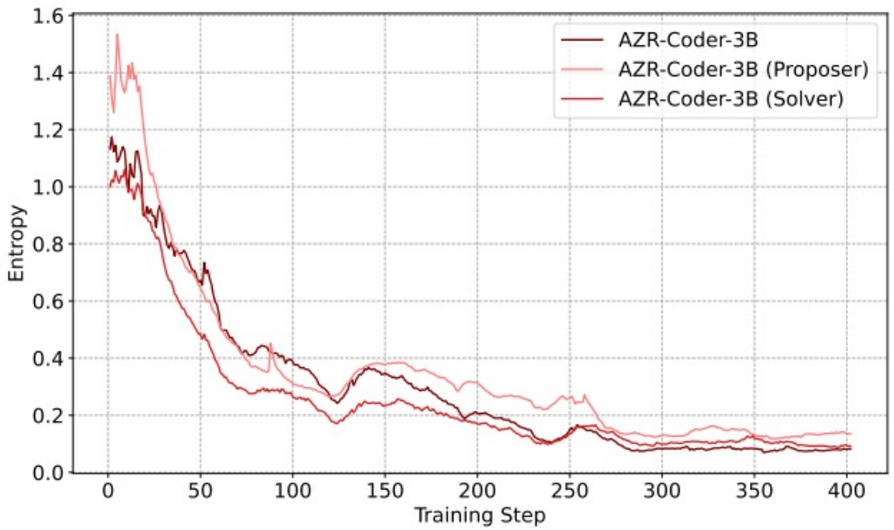
Zhao et al. "Absolute Zero: Reinforced Self-play Reasoning with Zero Data"

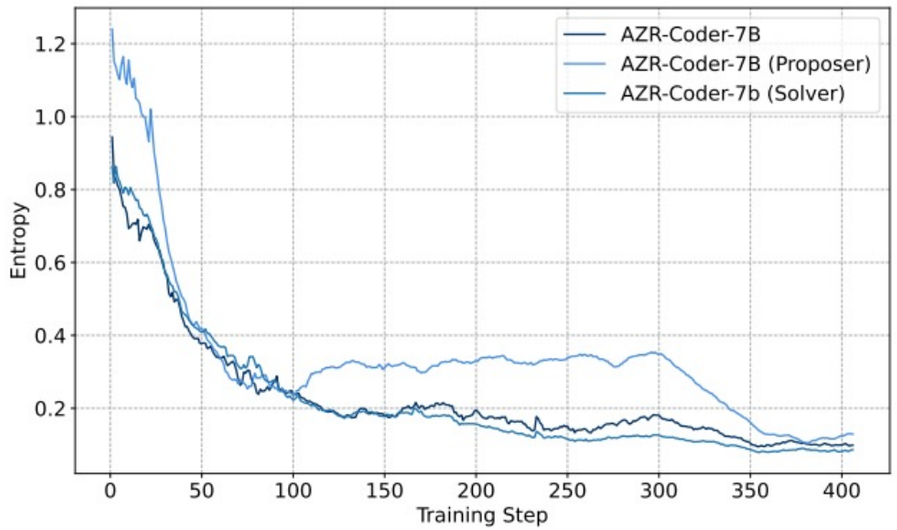
Intrinsic exploration



Huang et al. "R-Zero: Self-Evolving Reasoning LLM from Zero Data"

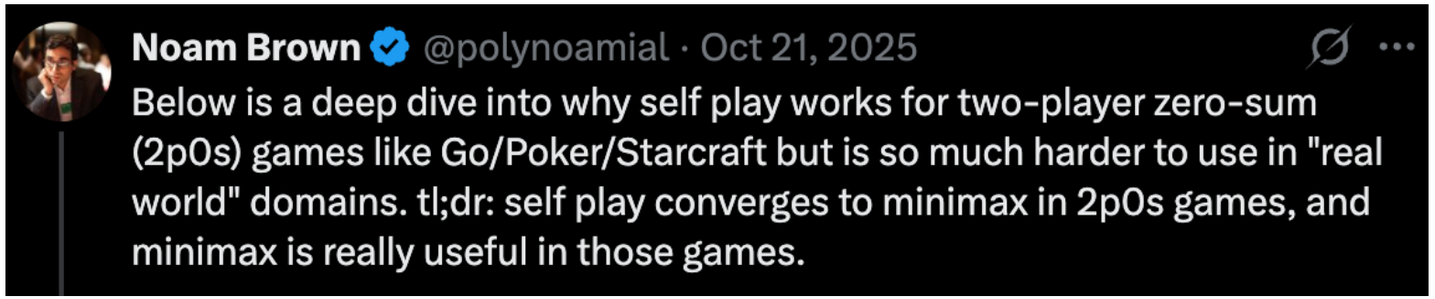
Entropy collapse

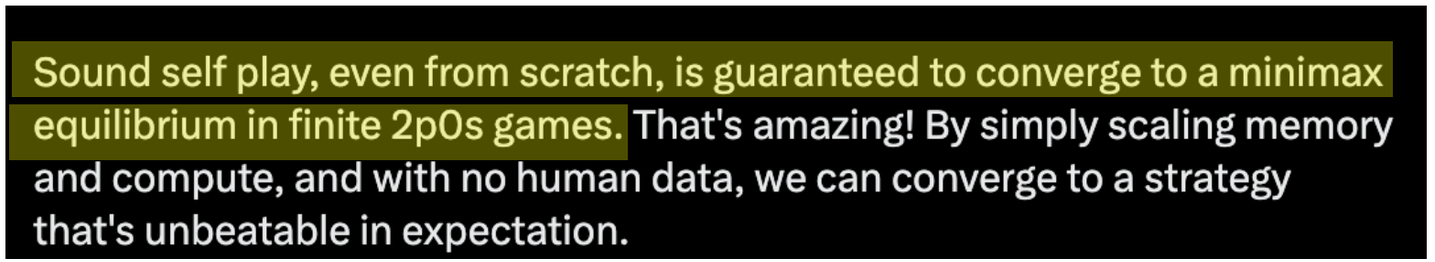


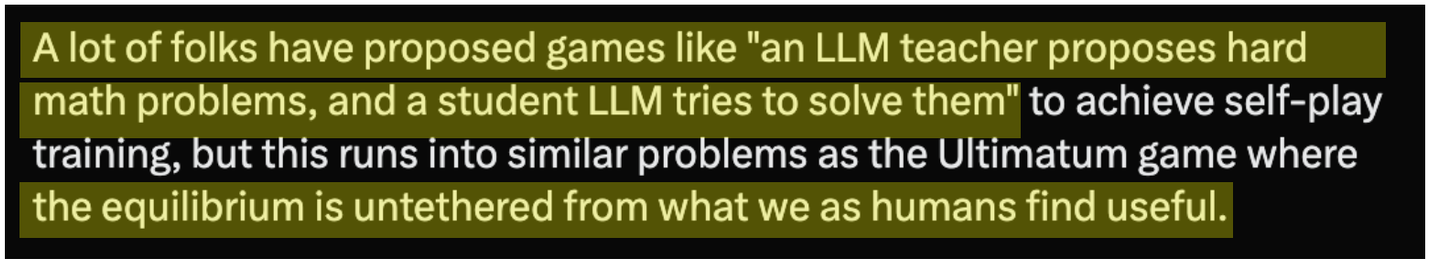


Chae et al. "Towards Understanding Self-Play for LLM Reasoning"

Insights from game theory?

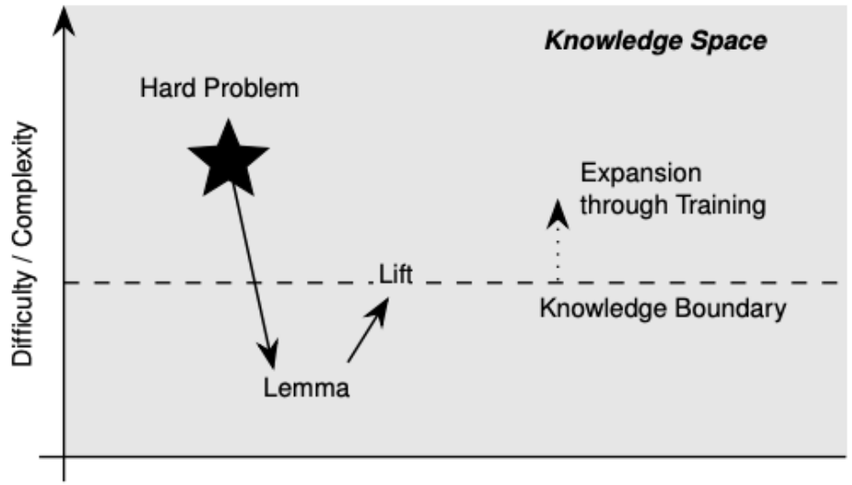


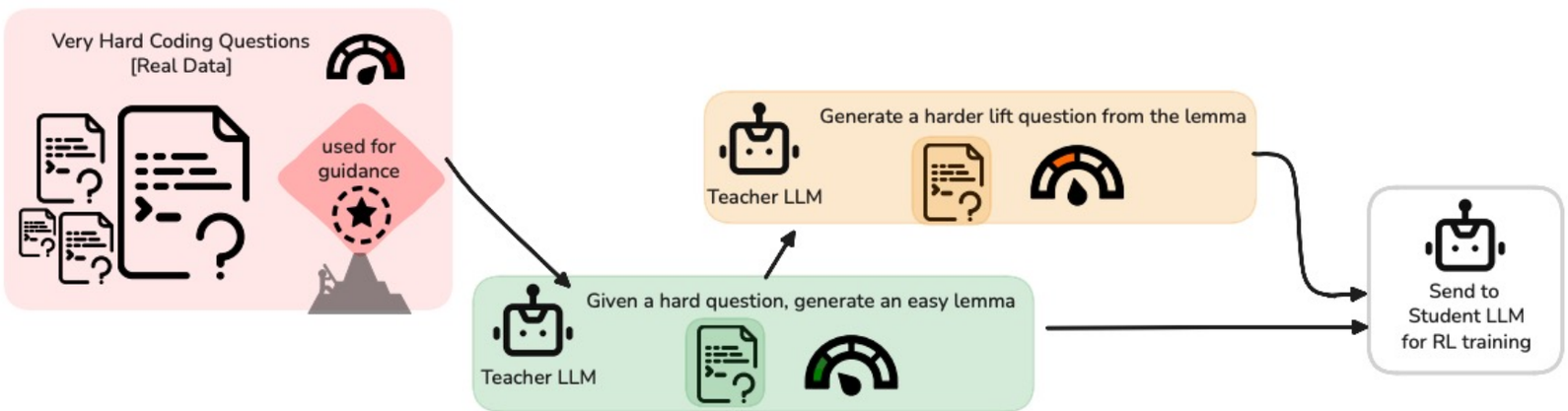




https://x.com/polynoamial/status/1985615741983605073

Guiding self-play with goal post problems



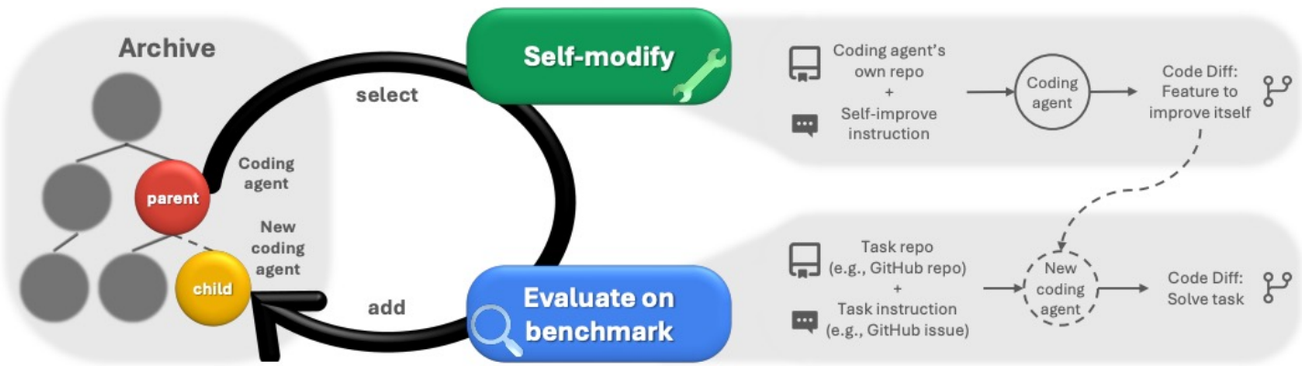


Jana et al. "GASP: Guided Asymmetric Self-Play for Coding LLMs". 2026.

Expanding the unit of improvement

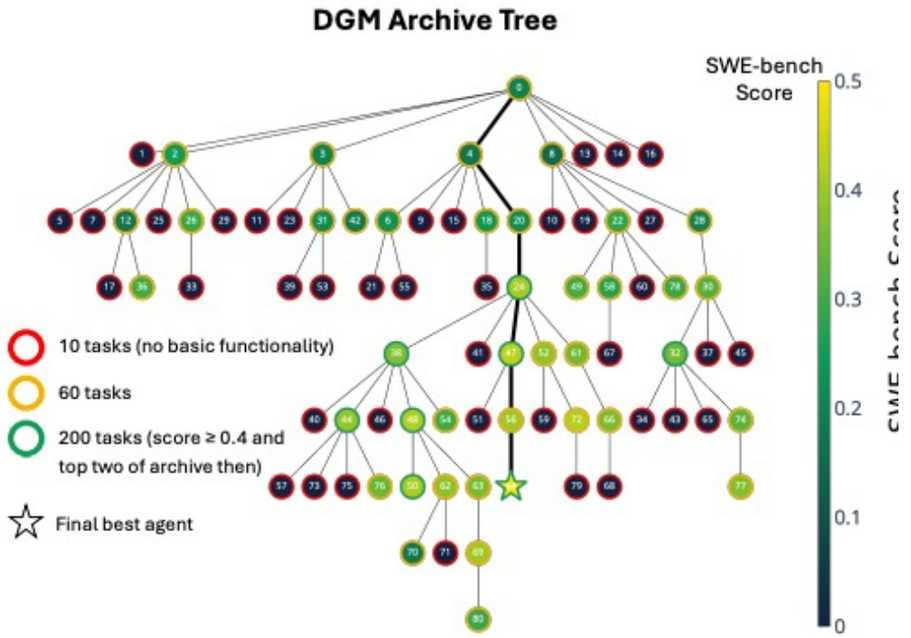
DARWIN GÖDEL MACHINE: OPEN-ENDED EVOLUTION OF SELF-IMPROVING AGENTS

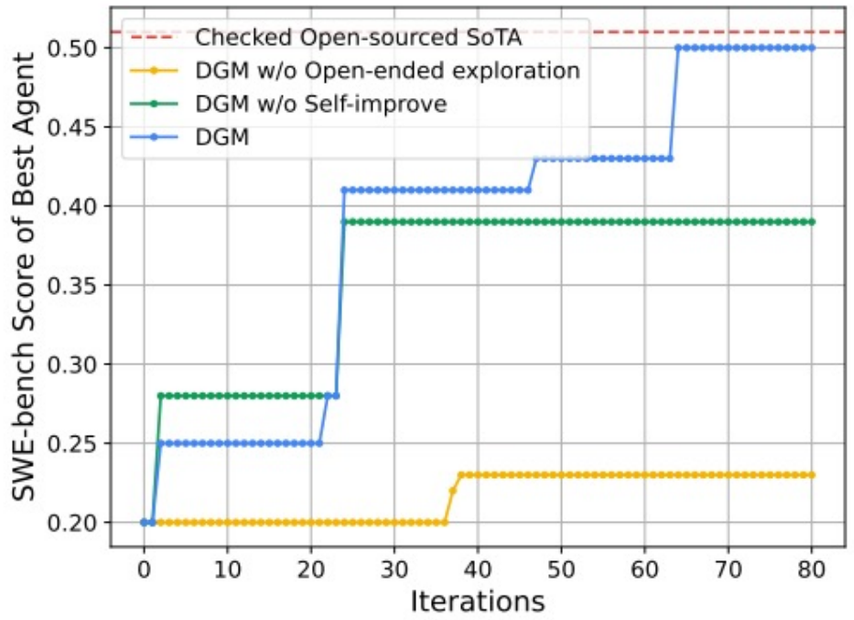
Jenny Zhang[*,1,2] Shengran Hu[*,1,2,3] Cong Lu[1,2,3] Robert Lange[†,3] Jeff Clune[†,1,2,4]



Zhang et al. "Darwin Godel Machine: Open-Ended Evolution of Self-Improving Agents". 2025.

Expanding the unit of improvement





Zhang et al. "Darwin Godel Machine: Open-Ended Evolution of Self-Improving Agents". 2025.

Self-play in toy environments

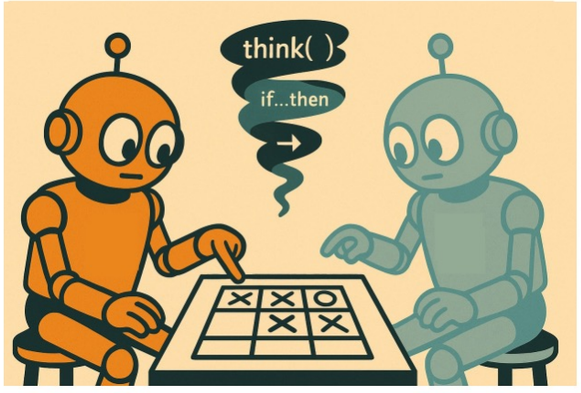
SPIRAL: Self-Play on Zero-Sum Games Incentivizes Reasoning via Multi-Agent Multi-Turn Reinforcement Learning

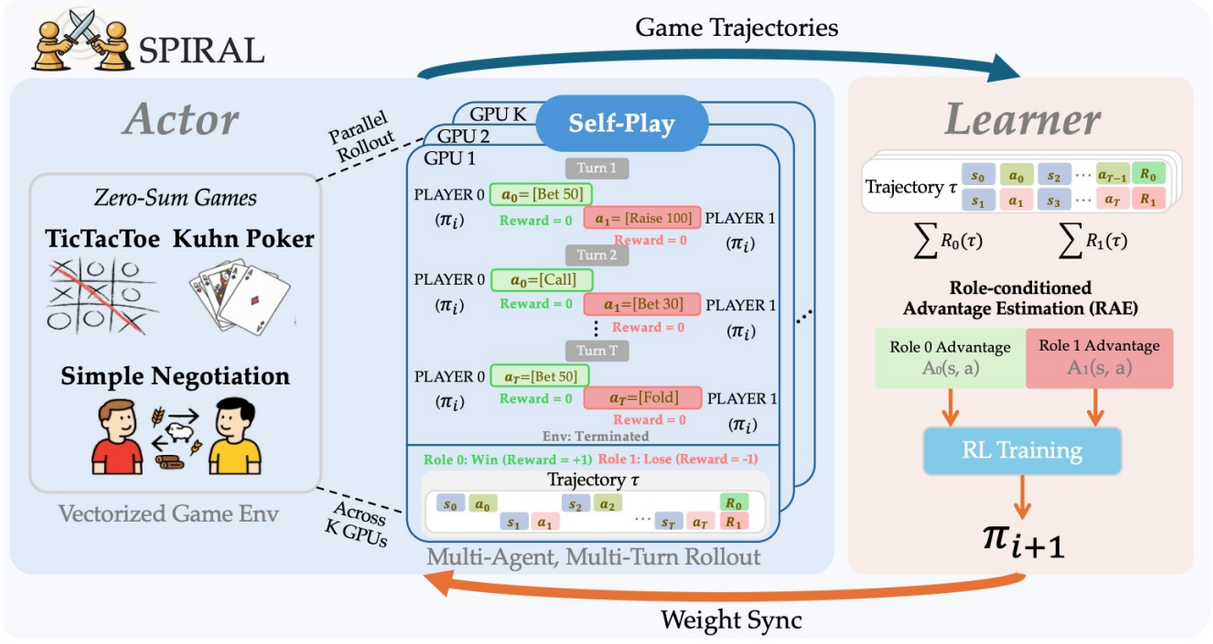
Bo Liu*[1], **Simon Yu***[2], **Zichen Liu***[1,3], **Leon Guertler***[4]
Penghui Qi[1,3], **Daniel Balcells**[5], **Mickel Liu**[6], **Cheston Tan**[4], **Weiyan Shi**[2], **Min Lin**[3], **Wee Sun Lee**[1]
Natasha Jaques†[6]
[1]National University of Singapore [2]Northeastern University [3]Sea AI Lab
[4]Centre for Frontier AI Research (CFAR), A*STAR [5]Plastic Labs [6]University of Washington

Self-play in toy environments

SPIRAL: Self-Play on Zero-Sum Games Incentivizes Reasoning via Multi-Agent Multi-Turn Reinforcement Learning

Bo Liu[*1], **Simon Yu**[*2], **Zichen Liu**[*1,3], **Leon Guertler**[*4]
Penghui Qi[1,3], **Daniel Balcells**[5], **Mickel Liu**[6], **Cheston Tan**[4], **Weiyan Shi**[2], **Min Lin**[3], **Wee Sun Lee**[1]
Natasha Jaques[†6]
[1]National University of Singapore [2]Northeastern University [3]Sea AI Lab
[4]Centre for Frontier AI Research (CFAR), A*STAR [5]Plastic Labs [6]University of Washington





Open Questions

- How to get truly open-ended self-improvement?
- Can we get guaranteed, useful convergence outside of two-player zero-sum game settings?
- Expanding the unit of improvement beyond model weights?
- Can we leverage synthetic environments that enable composable complexity?

North Star thought experiment (speculative)

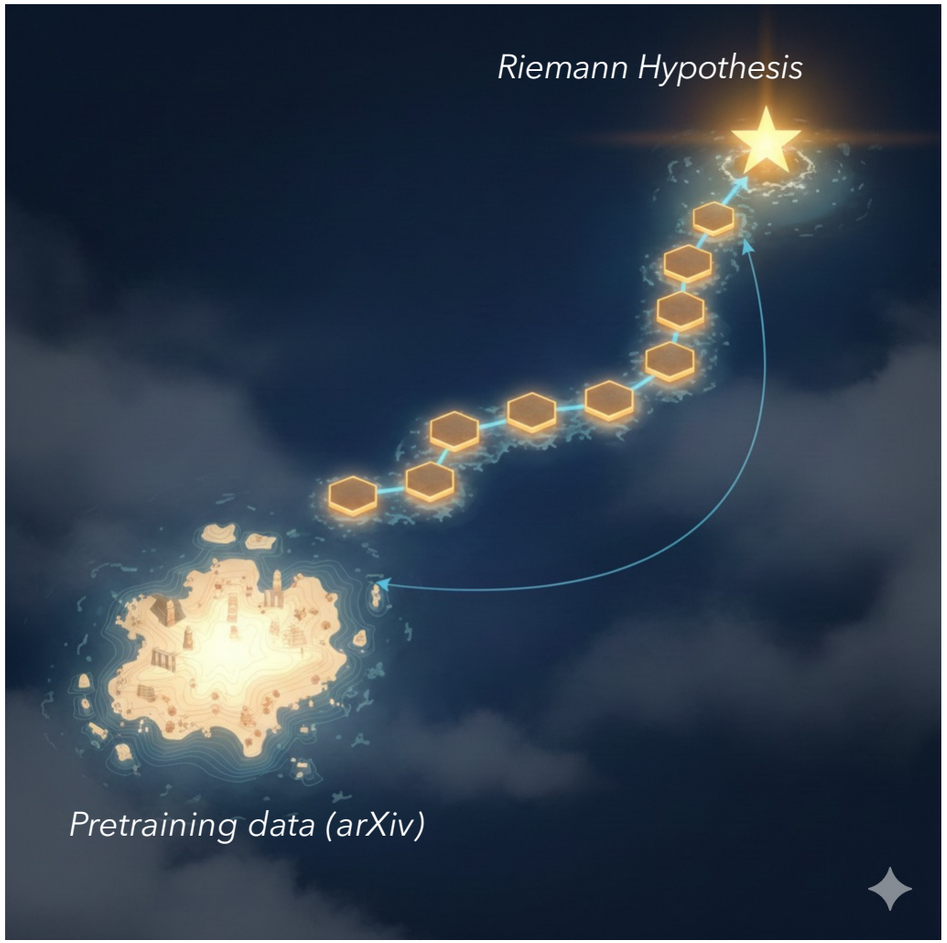


Figure from Gemini

Method	k				
	1	4	8	16	32
Base Model Inference	0.3 ± 0.1	1.0 ± 0.2	2.0 ± 0.4	3.9 ± 0.8	7.5 ± 1.3
Hard-Only	0.5 ± 0.1	1.7 ± 0.4	3.2 ± 0.8	5.7 ± 1.5	9.6 ± 2.6
Hard-Only ($g = 128$)	1.4 ± 1.0	3.9 ± 2.6	6.1 ± 3.9	8.9 ± 5.5	12.4 ± 7.4
SOAR-PQ (Ours)	$\mathbf{1.7 \pm 1.0}$	$\mathbf{5.3 \pm 2.6}$	$\mathbf{8.5 \pm 3.7}$	13.0 ± 4.8	18.9 ± 5.3
SOAR-PS (Ours)	1.0 ± 0.2	3.8 ± 0.6	6.8 ± 1.1	11.5 ± 1.6	18.1 ± 2.4
Grounded-T (Ours)	1.6 ± 0.5	5.1 ± 1.4	8.4 ± 2.1	$\mathbf{13.1 \pm 2.9}$	$\mathbf{19.1 \pm 3.7}$
Intrinsic-T	1.0 ± 0.6	3.3 ± 2.1	5.7 ± 3.5	9.2 ± 5.3	14.1 ± 7.5
HARP train (128)	2.4 ± 1.0	7.2 ± 2.4	11.3 ± 3.1	16.5 ± 3.6	23.0 ± 3.9
MATH train (128)	2.1 ± 0.0	6.6 ± 0.1	10.5 ± 0.3	15.7 ± 0.5	21.8 ± 0.9
MATH train (Full)	2.7 ± 0.2	7.6 ± 0.7	11.5 ± 1.2	16.4 ± 1.8	22.0 ± 2.4

Table 4 MATH Pass@k (%) Test Accuracy on Fail@128. Mean and SD over seeds are averaged over a 200 step window

Method	k				
	1	4	8	16	32
Base Model Inference	0.2 ± 0.0	0.9 ± 0.0	1.7 ± 0.0	3.4 ± 0.0	6.4 ± 0.0
Hard-Only	0.4 ± 0.1	1.4 ± 0.2	2.6 ± 0.4	4.7 ± 0.6	8.2 ± 1.0
SOAR-PQ (Ours)	$\mathbf{0.7 \pm 0.3}$	$\mathbf{2.5 \pm 0.8}$	$\mathbf{4.5 \pm 1.3}$	$\mathbf{7.7 \pm 1.7}$	$\mathbf{12.3 \pm 2.0}$
SOAR-PS (Ours)	0.6 ± 0.1	2.1 ± 0.3	3.9 ± 0.6	7.0 ± 0.9	11.8 ± 1.2
Grounded-T (Ours)	0.5 ± 0.2	2.0 ± 0.5	3.8 ± 0.9	6.7 ± 1.3	11.2 ± 1.7
Intrinsic-T	0.4 ± 0.1	1.6 ± 0.5	3.1 ± 0.8	5.6 ± 1.4	9.6 ± 2.1
HARP train (128)	0.4 ± 0.0	1.4 ± 0.1	2.8 ± 0.2	5.0 ± 0.5	8.7 ± 1.1
MATH train (128)	0.6 ± 0.1	2.1 ± 0.4	4.0 ± 0.7	7.1 ± 0.9	11.9 ± 0.9
MATH train (Full)	1.7 ± 0.2	5.1 ± 0.4	8.1 ± 0.4	11.7 ± 0.3	16.2 ± 0.4

Table 5 HARP Pass@k (%) Test Accuracy on fail@128. Mean and SD over seeds are reported at the timestep determined

Method	k				
	1	4	8	16	32
Base Model Inference	0.2 ± 0.0	0.8 ± 0.1	1.6 ± 0.3	3.1 ± 0.5	5.8 ± 1.0
Hard-Only	0.3 ± 0.1	1.1 ± 0.3	2.1 ± 0.6	3.9 ± 1.3	6.9 ± 2.7
SOAR-PQ (MATH) (Ours)	0.5 ± 0.1	1.9 ± 0.5	3.6 ± 0.9	6.4 ± 1.6	10.6 ± 2.7
SOAR-PQ (HARP) (Ours)	0.5 ± 0.1	2.0 ± 0.5	$\mathbf{3.8 \pm 1.0}$	$\mathbf{7.0 \pm 1.8}$	$\mathbf{12.0 \pm 3.0}$
SOAR-PS (MATH) (Ours)	$\mathbf{0.6 \pm 0.1}$	$\mathbf{2.1 \pm 0.5}$	3.7 ± 0.8	6.2 ± 1.3	9.9 ± 2.2
SOAR-PS (HARP) (Ours)	0.5 ± 0.1	2.0 ± 0.4	$\mathbf{3.8 \pm 0.7}$	6.9 ± 1.1	11.7 ± 1.6
Grounded-T (MATH) (Ours)	0.4 ± 0.2	1.6 ± 0.8	2.9 ± 1.4	5.3 ± 2.4	9.0 ± 4.0
Grounded-T (HARP) (Ours)	0.5 ± 0.2	1.9 ± 0.6	3.6 ± 1.1	6.5 ± 1.8	11.1 ± 2.9
Intrinsic-T	0.4 ± 0.3	1.7 ± 1.2	3.1 ± 2.0	5.5 ± 3.4	9.1 ± 5.2
HARP train (128)	0.5 ± 0.1	2.0 ± 0.2	3.6 ± 0.4	6.5 ± 0.8	10.6 ± 1.7
MATH train (128)	1.0 ± 0.1	3.4 ± 0.1	5.9 ± 0.1	9.6 ± 0.4	14.6 ± 1.4
MATH train (Full)	0.9 ± 0.0	3.2 ± 0.1	5.6 ± 0.3	8.8 ± 0.7	13.1 ± 0.9

Table 6 Olympiad Pass@k (%) Test Accuracy on fail@128. Mean and SD over seeds are reported timestep 50 with full