Does RL-based training generalize?

Austen Liao, Kuleen Sasse

Two Axes

Catastrophic Forgetting

RL's Razor: Why Online Reinforcement Learning Forgets Less

Generalization

SFT Memorizes, RL Generalizes: A Comparative Study of Foundation Model Post-training

RL's Razor: Why Online Reinforcement Learning Forgets Less

Idan Shenfeld, Jyothish Pari, Pulkit Agrawal

Introduction

- Long lived agents continuously need to adapt to new situations
- Forgetting previous blocks continual adaptation
- Why does **on-policy RL** offer a path to preserve past skills over **SFT** and **off-policy RL**?

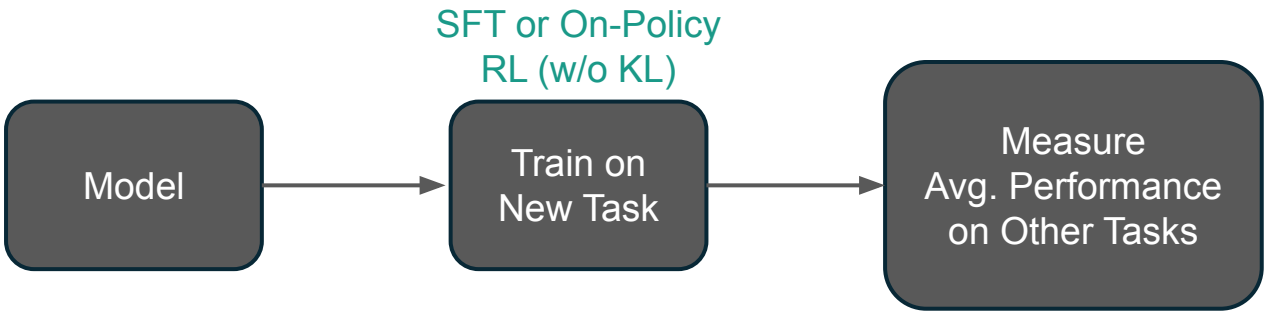
Background

- Foundation model post-training
- SFT versus RL
- Catastrophic forgetting

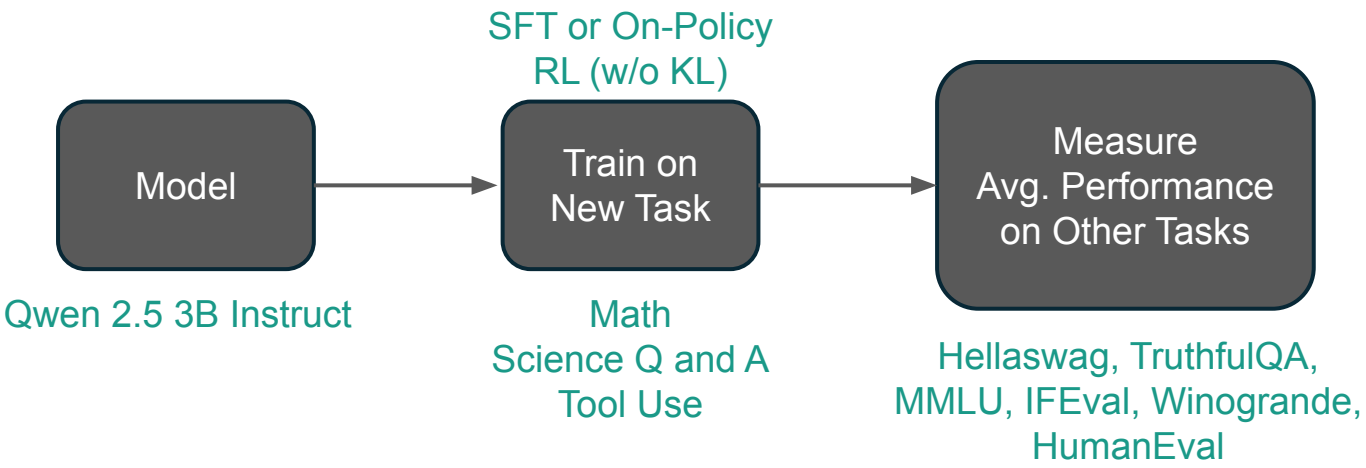
Key points

- On-policy RL adapts to new tasks with minimal forgetting
- KL divergence predicts catastrophic forgetting
- On-policy training drives the advantage

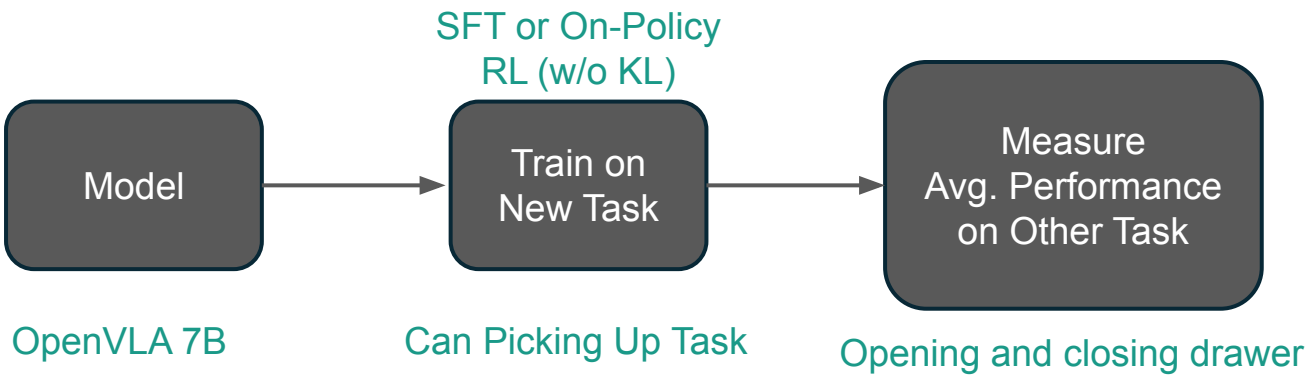
Minimal Forgetting - Experiment



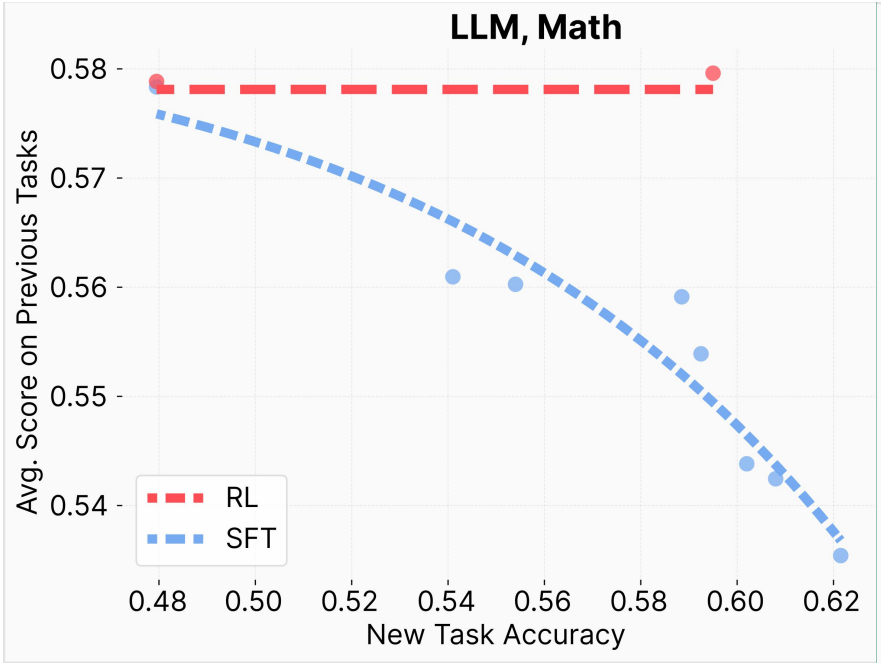
Minimal Forgetting (LLMs) - Experiment

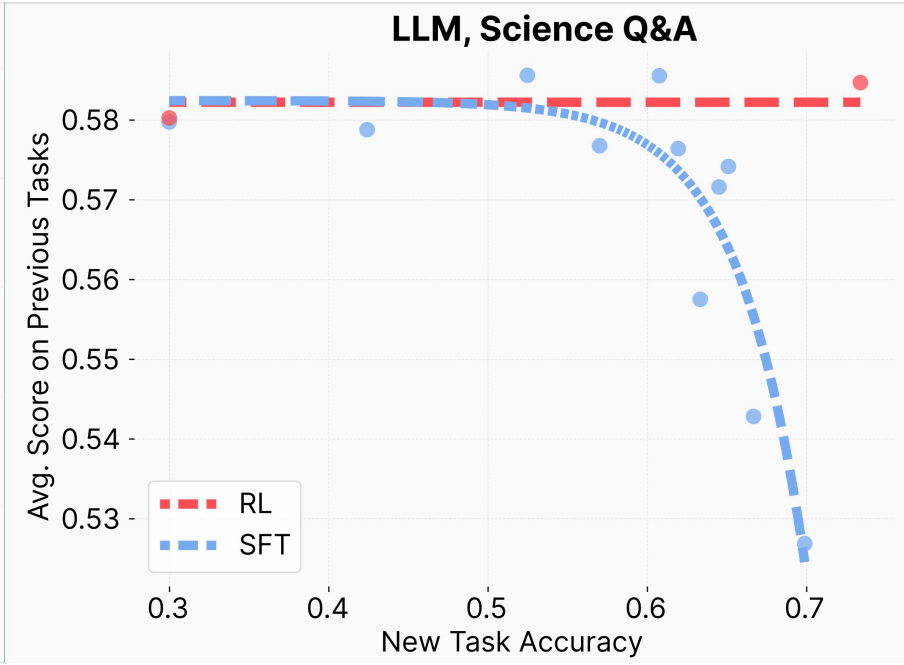


Minimal Forgetting (VLAs) - Experiment



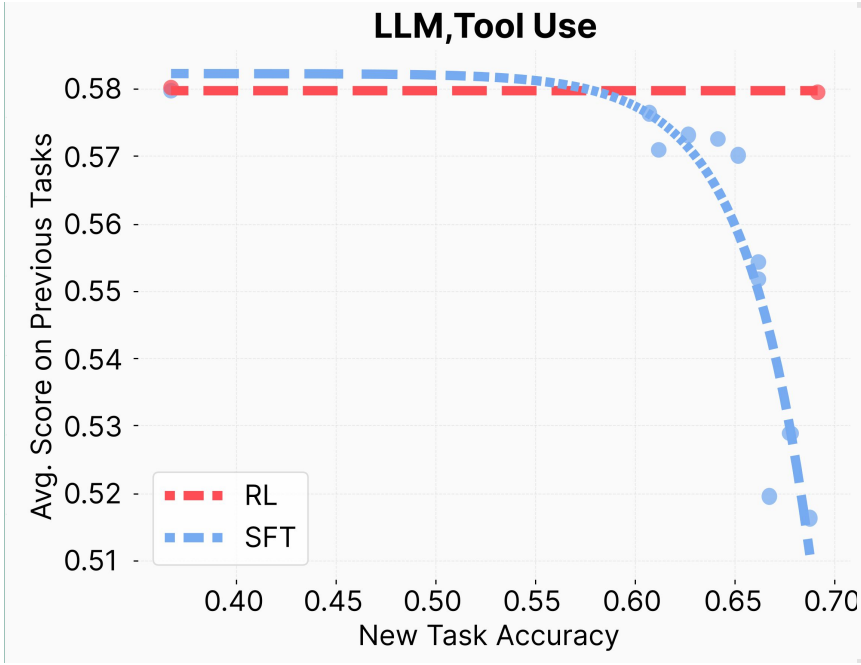
Minimal Forgetting - Results

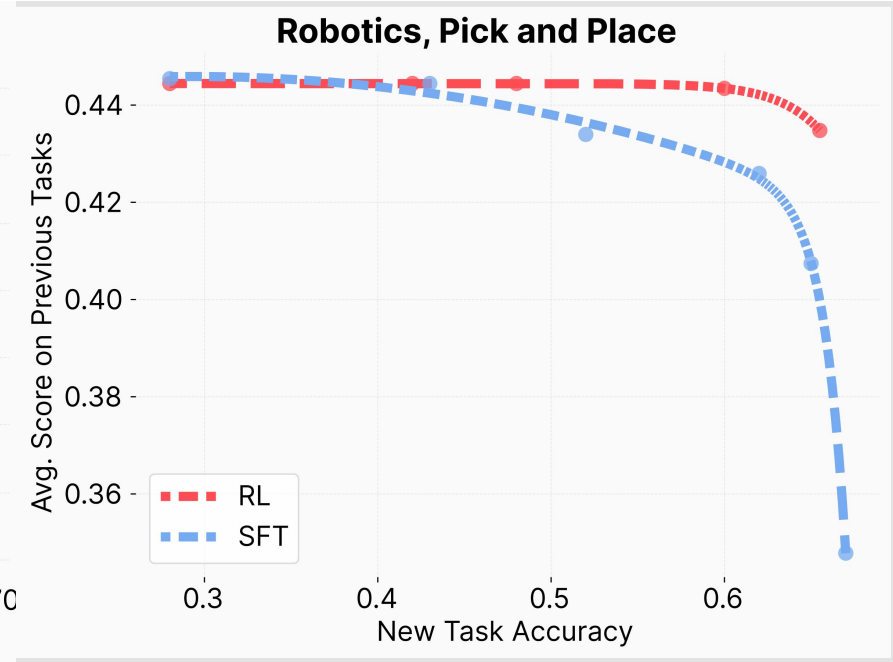




RL does not forget!

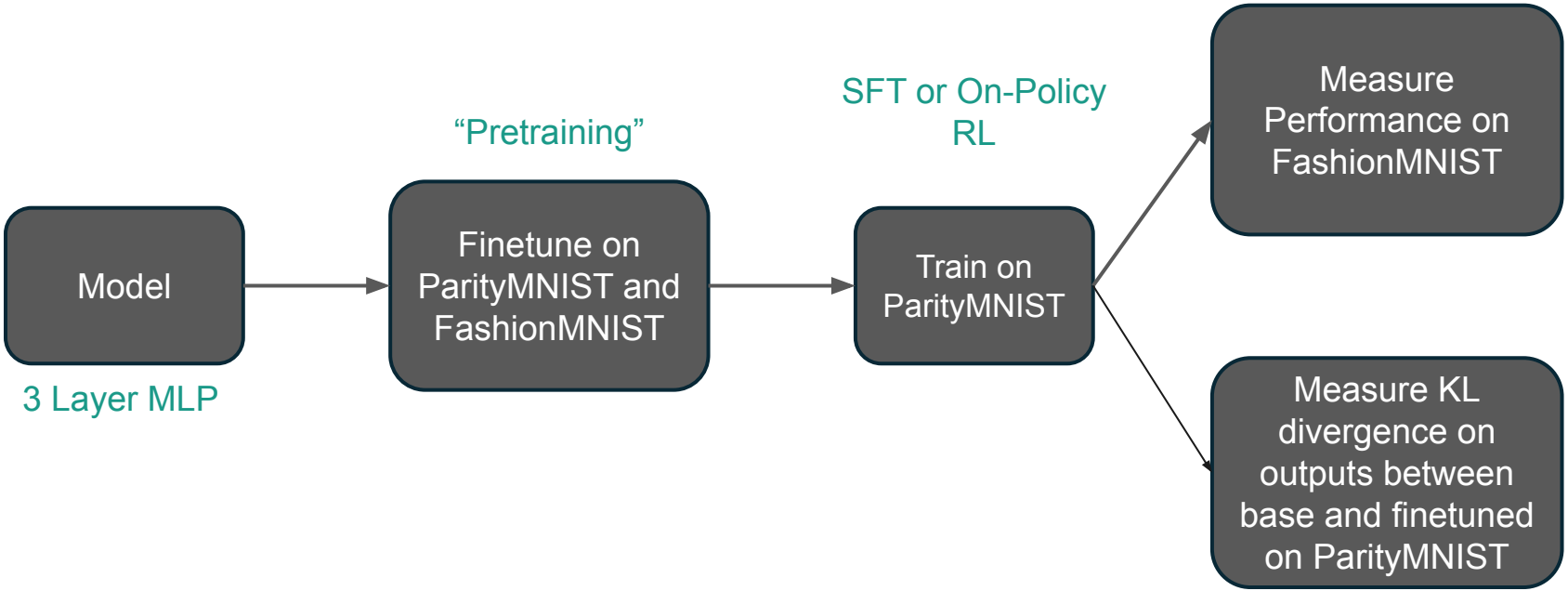
Minimal Forgetting - Results



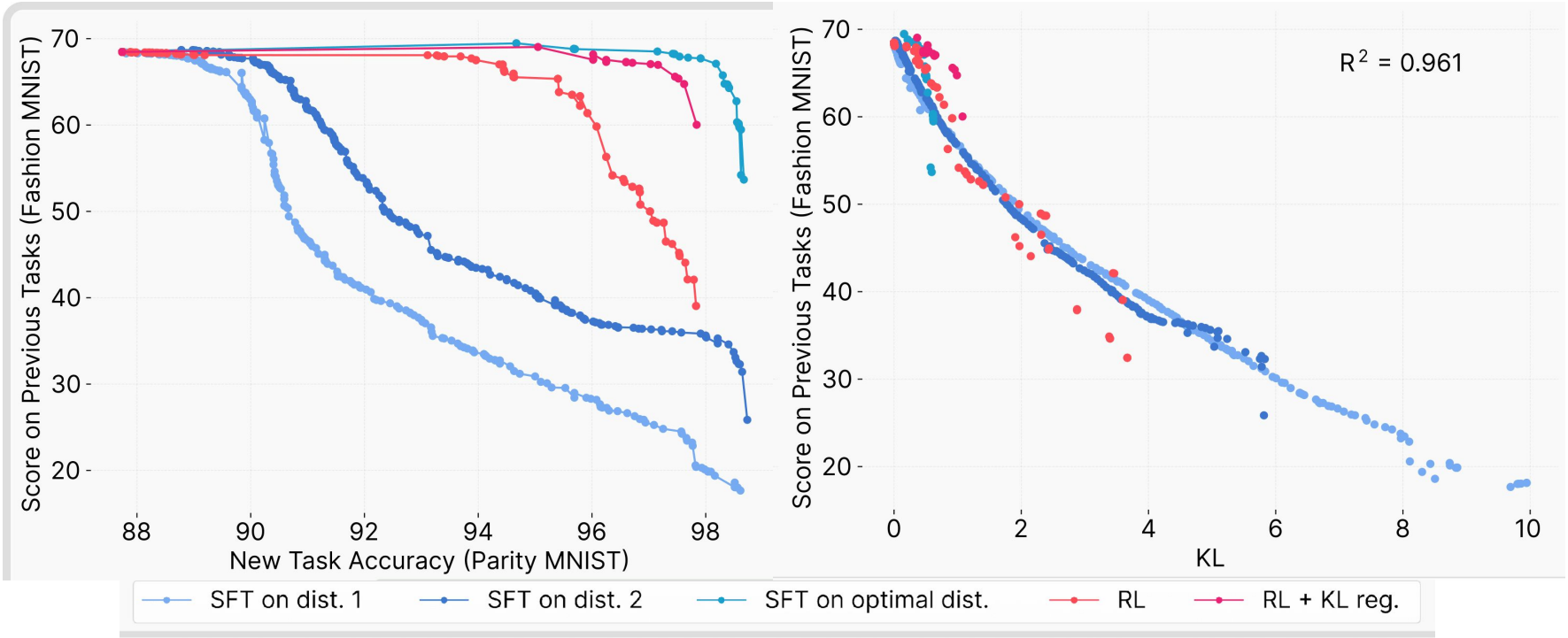


RL does not forget (on some tasks)!

KL Predicts Forgetting - Experiment

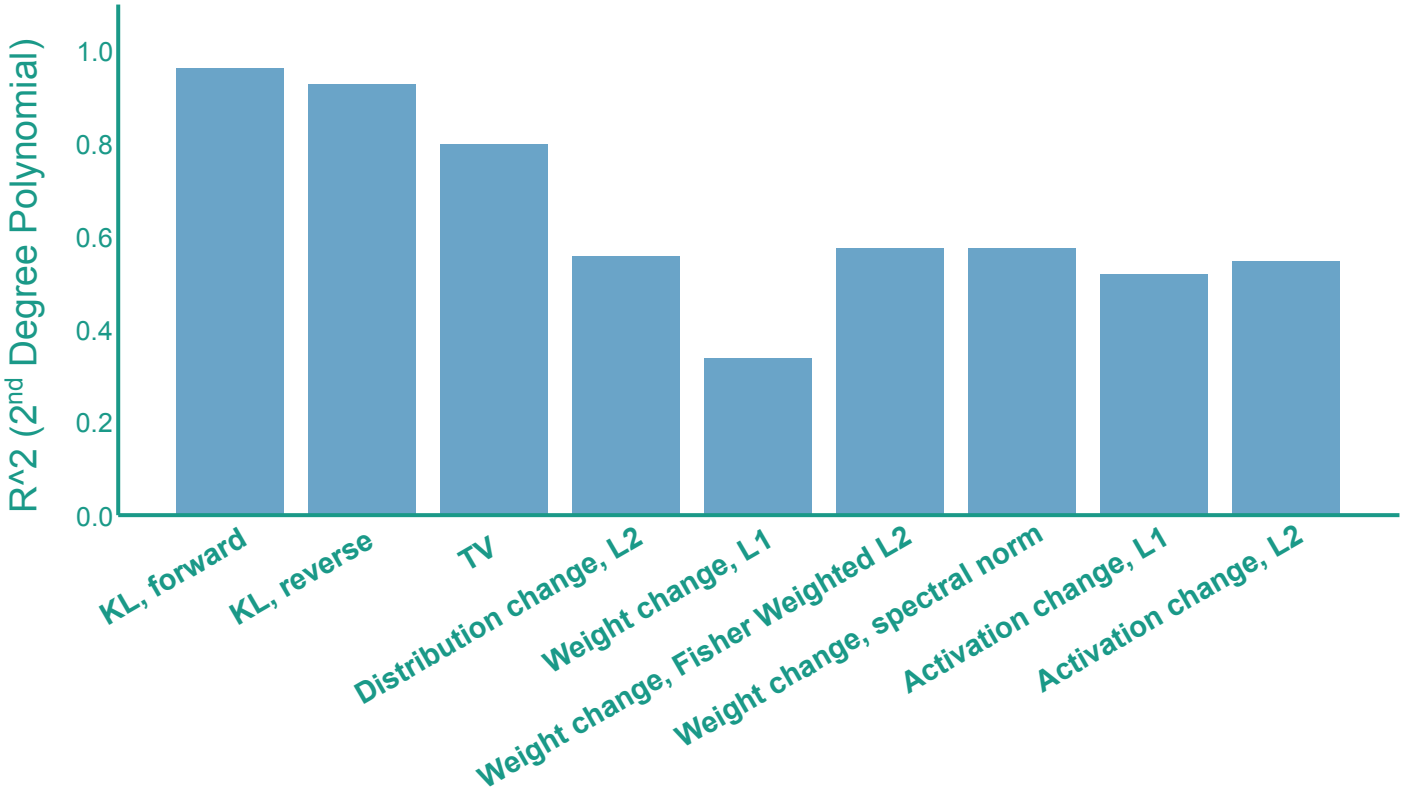


KL Predicts Forgetting - Results



Across algorithms: KL ∝ Score!

KL Predicts Forgetting - Results

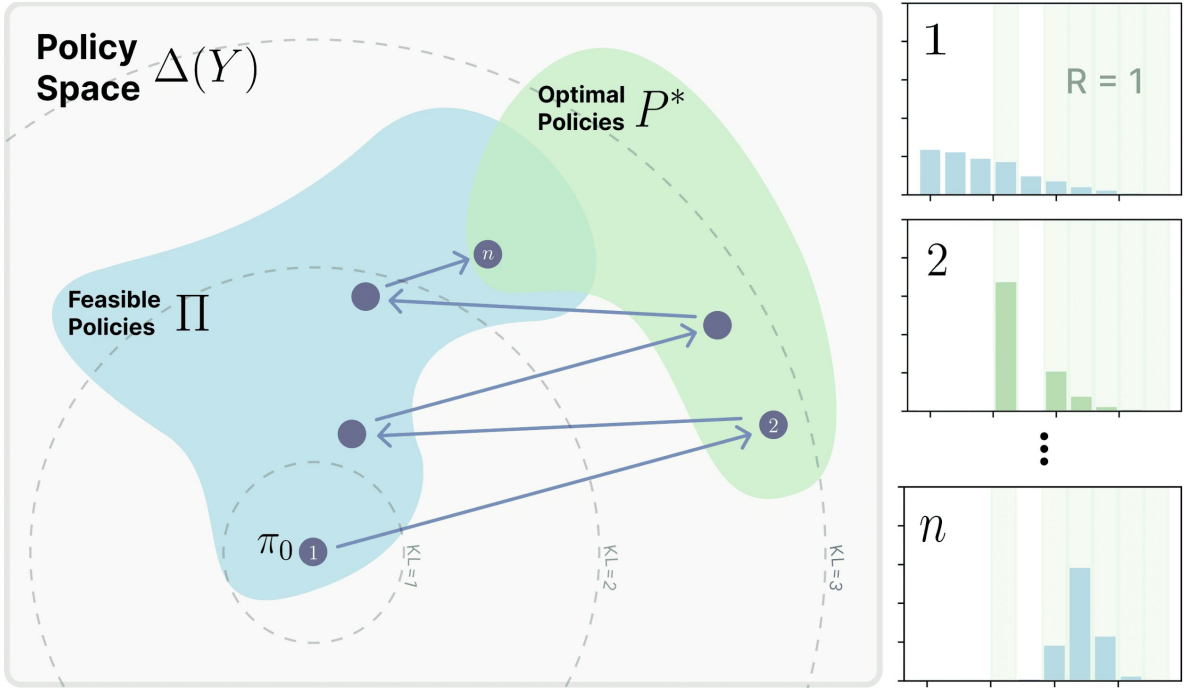


On-Policy > Everything - Empirical

	Pos Examples	Pos + Neg Examples
On Policy	1-0 Reinforce	GRPO
Offline	SFT	SIMPO

On-Policy > Everything - Theory

- Policy gradient keeps closer to base model
- Reweights likely outputs → shifts mass to higher reward
- Updates relative to own distribution → small, local KL shift



Conclusion (Convinced?)

- On-policy RL has minimal catastrophic forgetting in comparison to off policy RL and SFT
- KL divergence has a strong correlation with catastrophic forgetting
- On-policy training drives the advantage theoretically and empirically

SFT Memorizes, RL Generalizes: A Comparative Study of Foundation Model Post-Training

Tianzhe Chu, Yuexiang Zhai, Yihan Yang, Shengbang Tong, Saining Xie, Dale Shuurmans, Quoc V. Le, Sergey Levine, Yi Ma

Paper Introduction

- Research question – what effects do post-training methods like RL and SFT have on a model's ability to generalize?
- The authors investigate by post-training and evaluating models on **different variants** of both text-only and multimodal tasks
    - rule-based variants
    - visual distribution variants

Findings

- **RL proficiently generalizes across both variant axes, while SFT does not**
- SFT is still crucial prior to RL to *stabilize model outputs* & instill instruction-following ability
- *increasing # of verification steps* => better generalization

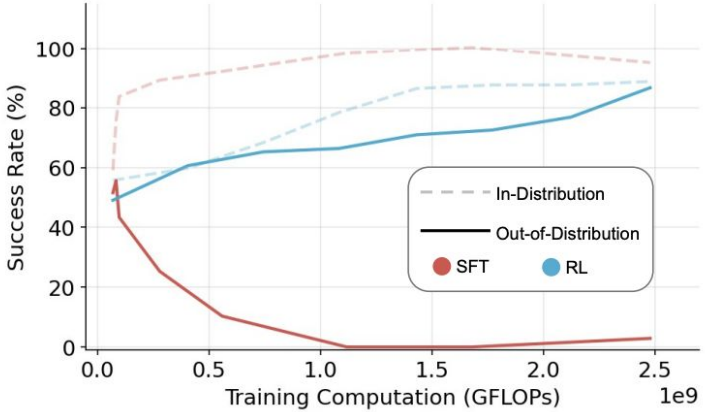
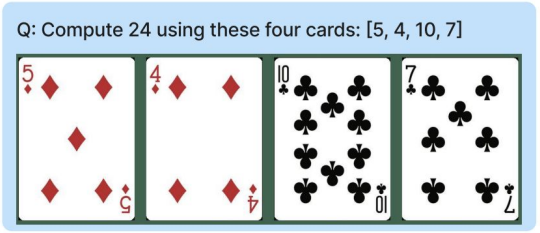


Figure 1: **A comparative study of RL and SFT on the visual navigation environment** `V-IRL` **(Yang et al., 2024a) for OOD generalization.** OOD curves represent performance on the same task, using *a different textual action space*. See detailed descriptions of the task in Section 5.1.

Evaluation Tasks

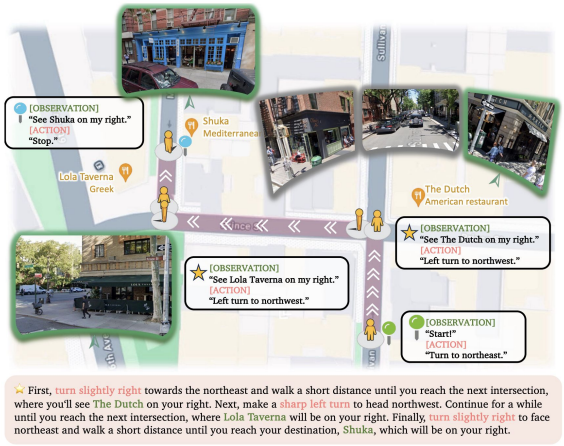
GeneralPoints (GP)

- A more general version of Points24
- Goal: given 4 cards, produce an equation that equals a target number (e.g. 24)
- task modalities:
  - GP-L: input = description of 4 cards
  - GP-VL: input = an image of 4 cards laid out



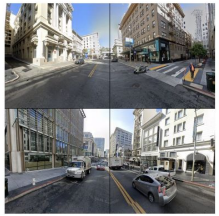
V-IRL

- takes place in V-IRL environment (Yang, 2024a)
- Goal: given set of instructions *containing spatial info*, navigate to a target location in the world
- task modalities:
  - V-IRL-L: info w.r.t. notable landmarks near the agent are automatically provided
  - V-IRL-VL: vision input provided at every timestep - agent is tasked w/ visual recognition



System Prompt (v_0^{in})

[Task Description]
You are an expert in navigation. You will receive a sequence of instructions to follow while observing your surrounding street views. You are also provided with your observation and action history in text. your goal is to take the action based on the current observation and instruction.



[Instruction]
1. First, turn left to face east.
2. Move forward until you reach the next intersection where Hotel 32One is on your right behind.
3. Turn right to face north.
4. Move forward until you reach the next intersection where Dragon Gate Chinatown SF is on your right front.
5. Turn left to face east.
6. Move forward until the destination Café de la Presse is on your right.

[Current observation]
You observe a 2x2 grid of street view images with the following headings:
[front, right
back, left]
You need to identify if any of the landmarks in the instruction are visible in the street view grid.

[Action space]
- "forward()": indicates moving forward for 1 step;
- "turn_direction(x)": indicates turn direction to the target heading, where x∈['north', 'northeast', 'east', 'southeast', 'south', 'southwest', 'west', 'northwest'];
- "stop()": indicates the navigation is finished;

[Observations and actions sequence]
O_1: No landmarks nearby;
A_1: turn_direction(east)
O_2: No landmarks nearby;
A_2: forward()
O_3: No landmarks nearby;
A_3: forward()
O_4: You observe an image of 4 views; You observe an intersection
A_4:

System Prompt (v_0^{in})

[Task Description]
You are an expert in navgation. You will receive a sequence of instructions to follow. You are also provided with your observation and action histroy in text. Your goal is to first analyze the instruction and identify the next sentence to be executed. Then, you need to provide the action to be taken based on the current observation and instruction.

[Instruction]
1. First, turn left to face east.
2. Move forward until you reach the next intersection where Hotel 32One is on your right behind.
3. Turn right to face north.
4. Move forward until you reach the next intersection where Dragon Gate Chinatown SF is on your right front.
5. Turn left to face east.
6. Move forward until the destination Café de la Presse is on your right.

[Action space]
- "forward()": indicates moving forward for 1 step;
- "turn_direction(x)": indicates turn direction to the target heading, where x∈['north', 'northeast', 'east', 'southeast', 'south', 'southwest', 'west', 'northwest'];
- "stop()": indicates the navigation is finished;

[Observations and actions sequence]
O_1: No landmarks nearby;
A_1: turn_direction(east)
O_2: No landmarks nearby;
A_2: forward()
O_3: No landmarks nearby;
A_3: forward()
O_4: Hotel 32One is on your right behind; You observe an intersection
A_4:

Takeaway: V-IRL-VL adds visual recognition to the task, whereas V-IRL-L handles it for you

Task Variants

GeneralPoints

Rule-Based Variants

- In-distribution (ID): J, Q, K = 11, 12, 13
- Out-of-distribution (OOD): J, Q, K = all 10

Visual Variants

- ID: cards are all black suits
- OOD: cards are all red suits

Points24

Rule-Based Variants

- ID: “absolute orientation” action space
  - A ∈ [north, northeast, east, …]
- OOD: “relative orientation” action space
  - A ∈ [left, right, slightly left, slightly right]

Visual Variants

- ID: routing tasks from in New York City
- OOD: routing tasks collected in various cities around the world

Results

Results - Generalization Across Rules

- authors measured delta of performance on each task's OOD variant after applying RL or SFT
- findings:
  - RL - significant gains
  - SFT - catastrophic losses

	RL	SFT
GP-L	**+3.5** (11.5 → 15.0)	**-8.1** (11.5 → 3.4)
V-IRL-L	**+11.0** (80.8 → 91.8)	**-79.5** (80.8 → 1.3)
GP-VL	**+3.0** (11.2 → 14.2)	**-5.6** (11.2 → 5.6)
V-IRL-VL	**+9.3** (35.7 → 45.0)	**-33.2** (35.7 → 2.5)

Results - Generalization in Visual OOD Tasks

- authors measured delta of performance on each task's OOD variant after applying RL or SFT
- findings:
  - RL - significant gains
  - SFT - catastrophic losses

	RL	SFT
GP-VL	**+17.6** (23.6 → 41.2)	**-9.9** (23.6 → 13.7)
V-IRL-VL	**+61.1** (16.7 → 77.8)	**-5.6** (16.7 → 11.1)

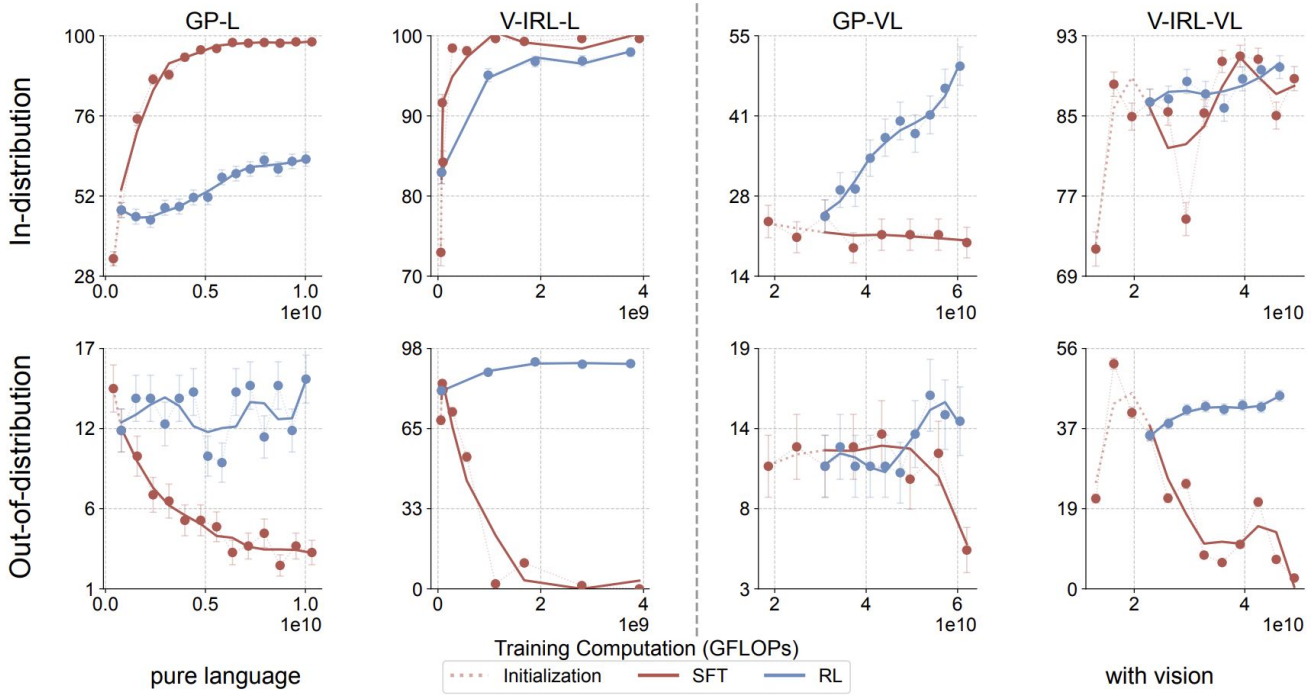


Figure 5: **Success rate (%) - GFLOPs trendlines for RL and SFT on `GeneralPoints` and `V-IRL`.** The top row shows in-distribution performance, while the bottom row shows out-of-distribution performance. Results are presented for both pure language (`-L`) and vision-language (`-VL`) variants of each task. For `GeneralPoints`, we report the episode success rate, while for `V-IRL`, we report per-step accuracy with overall success rate in Figures 1 and 18. Detailed evaluation setups (and curve smoothing) are provided in Appendix C.3.

Analyzing Visual Recognition Accuracy

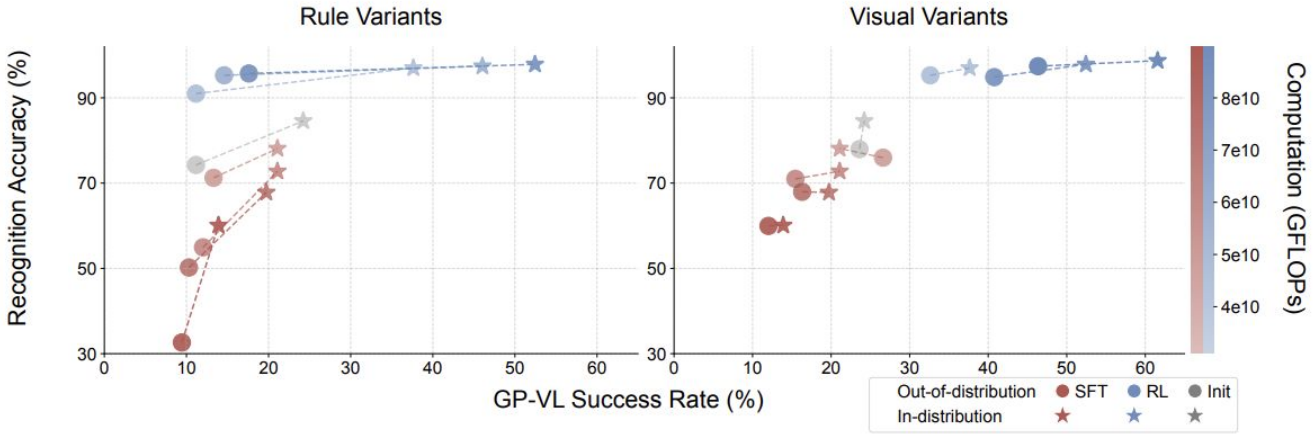
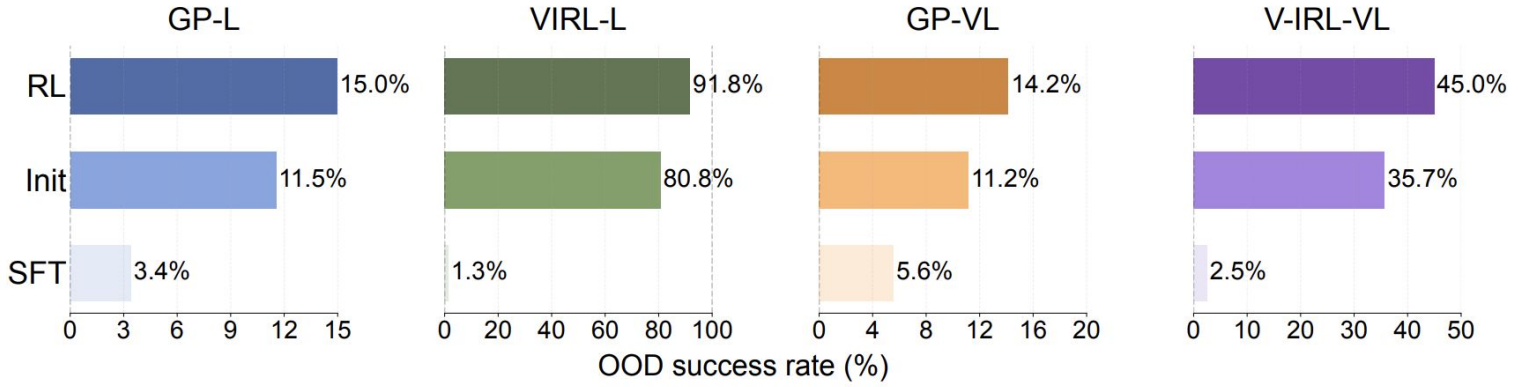


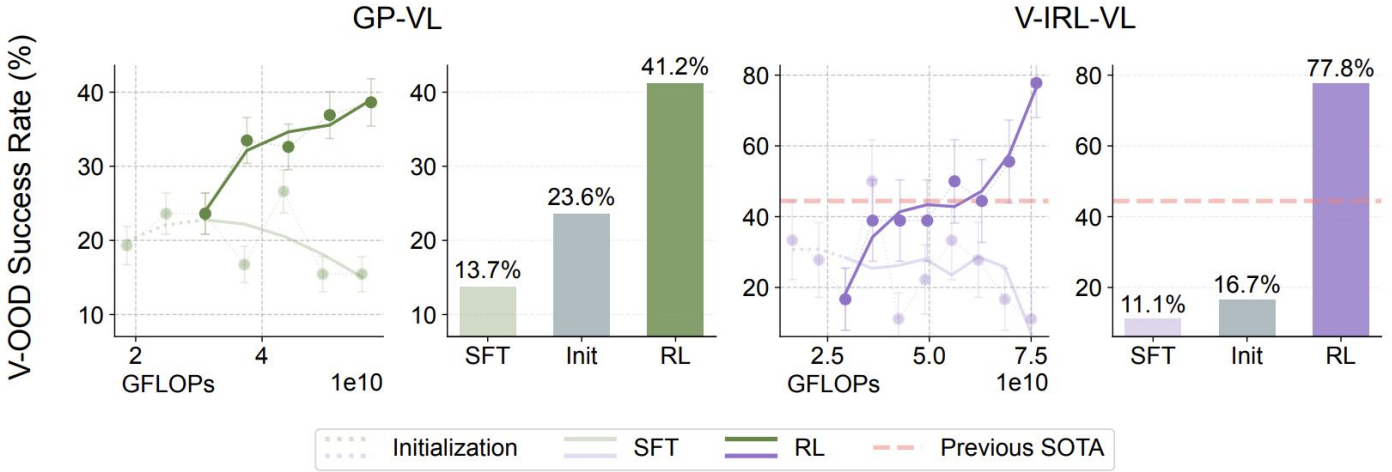
Figure 8: **Recognition vs. success rate for RL and SFT under different variants in GP-VL.** We report both in-

- correlation between success and accuracy
- trajectory of different colors
- relative horiz. position of circles / stars
- relative vert. position of circles / stars

Rule-based:

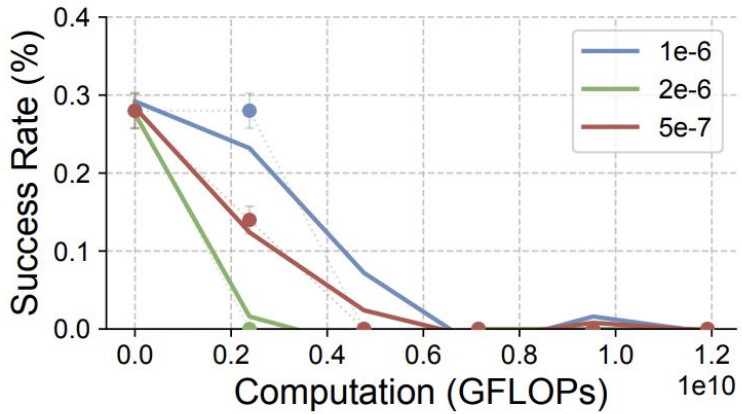


Visual-based:



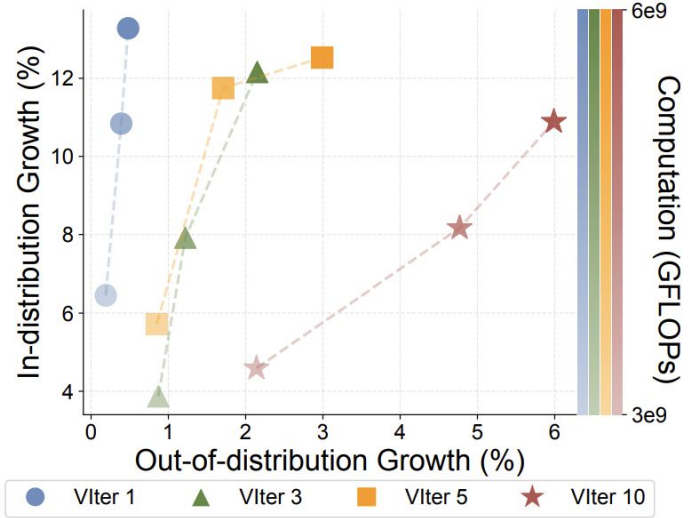
Other Results

SFT is still necessary to stabilize outputs prior to RL



RL generally improves visual capabilities (esp. visual recognition)

More verification steps => better generalization through RL



Conclusion

This paper finds that, in both language-based and multimodal settings, RL exhibits a positive effect on generalization, while SFT appears to bring about pure memorization.

Which result is more convincing?

Are they even convincing in the first place?

How do we explain these results?