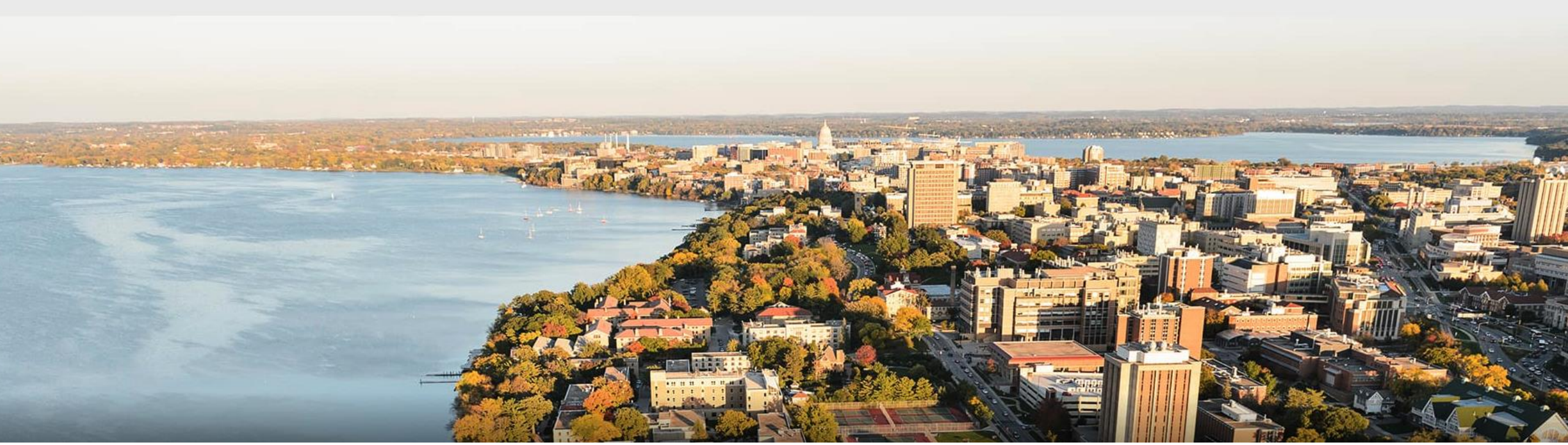


CS 839: Foundation Models
Reinforcement Learning With Verifiable Rewards

Fred Sala

University of Wisconsin-Madison

Oct. 7, 2025

Announcements

- **Logistics:**
  - Presentation information here:
    https://pages.cs.wisc.edu/~fredsala/cs839/fall2025/hw/cs839_presentation_info_25.pdf
  - Sign-up:
    https://docs.google.com/spreadsheets/d/1T_h6Xtn7GcWgozAtLFlgsV4j7MCkL3IvbvBf_klhUCE/edit?usp=sharing
  - **Homework 2:** Out
    - Homework 1: Grading (will be about 2 weeks)
- Class roadmap:

Tuesday Oct. 7	RLVR
Thursday Oct. 9	Efficient Training
Tuesday Oct. 14	Efficient Inference

Outline

- **RLHF Review**
  - Motivation for alignment, overall pipeline, preference and reward models, PPO objective
- **Variations and Some Open Questions**
  - Direct preference optimization (DPO), RLAIF, other techniques
- **Verifiable Rewards**
  - Verifiers, applications, GRPO, extensions

Outline

- **RLHF Review**
  - Motivation for alignment, overall pipeline, preference and reward models, PPO objective
- **Variations and Some Open Questions**
  - Direct preference optimization (DPO), RLAIF, other techniques
- **Verifiable Rewards**
  - Verifiers, applications, GRPO, extensions

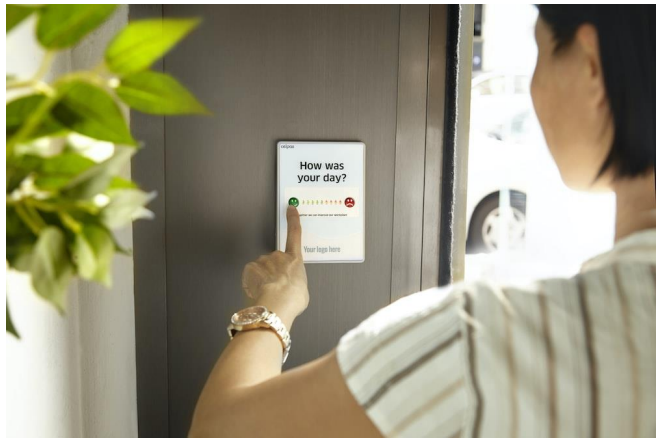
Review: Alignment **Basic Motivation**

Goal: produce language model outputs that users like better…

- **Hard** to specify exactly what this means,
- **Easy** to query users

Collect human feedback and use it to change the model

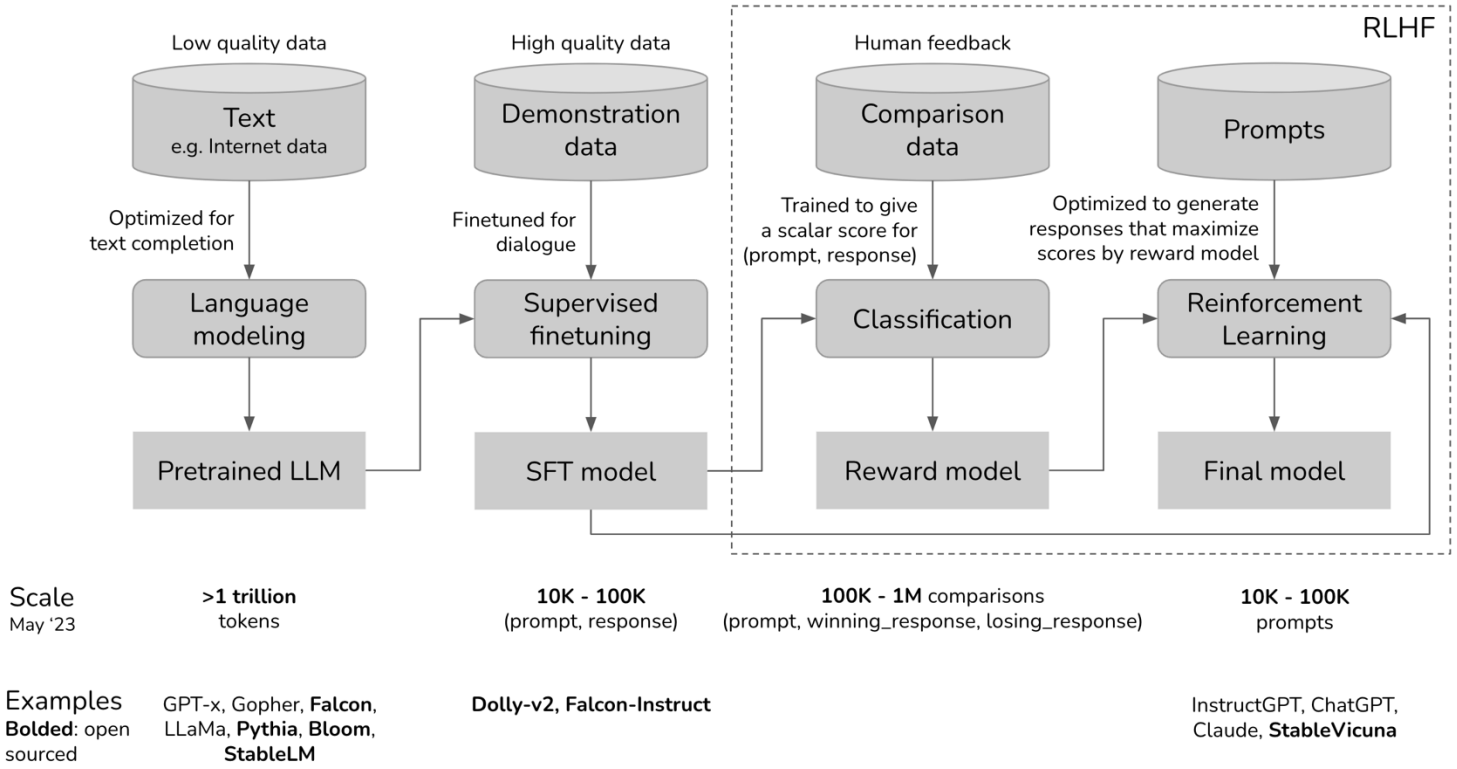
- Can do this by fine-tuning, especially with instructions
- Doesn’t quite capture what users want
- We’ll use other approaches, like RLHF



Review: RLHF **Setup**

Goal: produce language model outputs that users like better…

- Via RL with trained reward model (Ouyang et al ‘222)



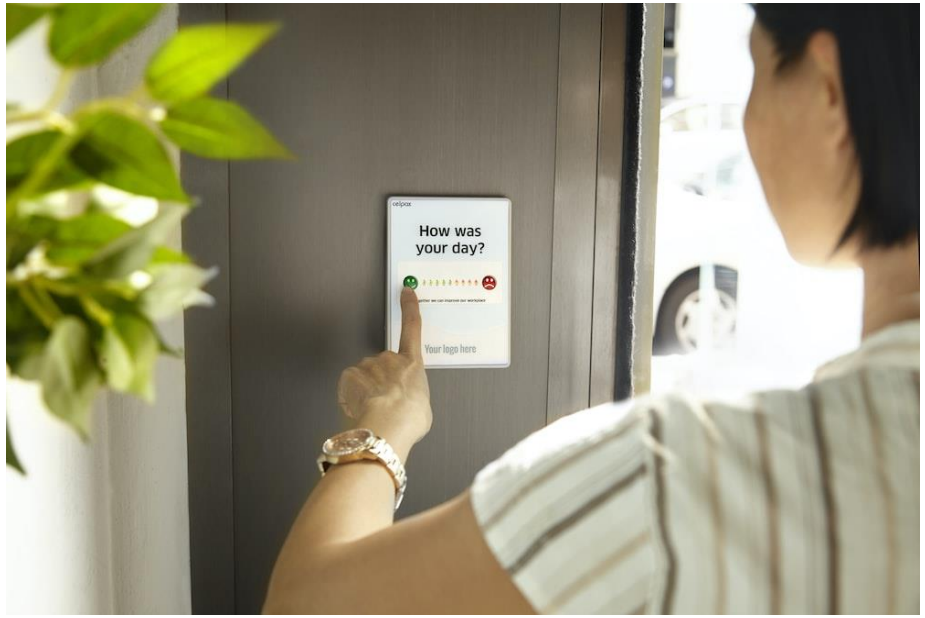
Chip Huyen

RLHF: **Feedback**

First stage: get **human feedback** to train reward model

- Fix a set of prompts
- Produce multiple outputs for each prompt
  - Can get them from the original model post-SFT, or otherwise

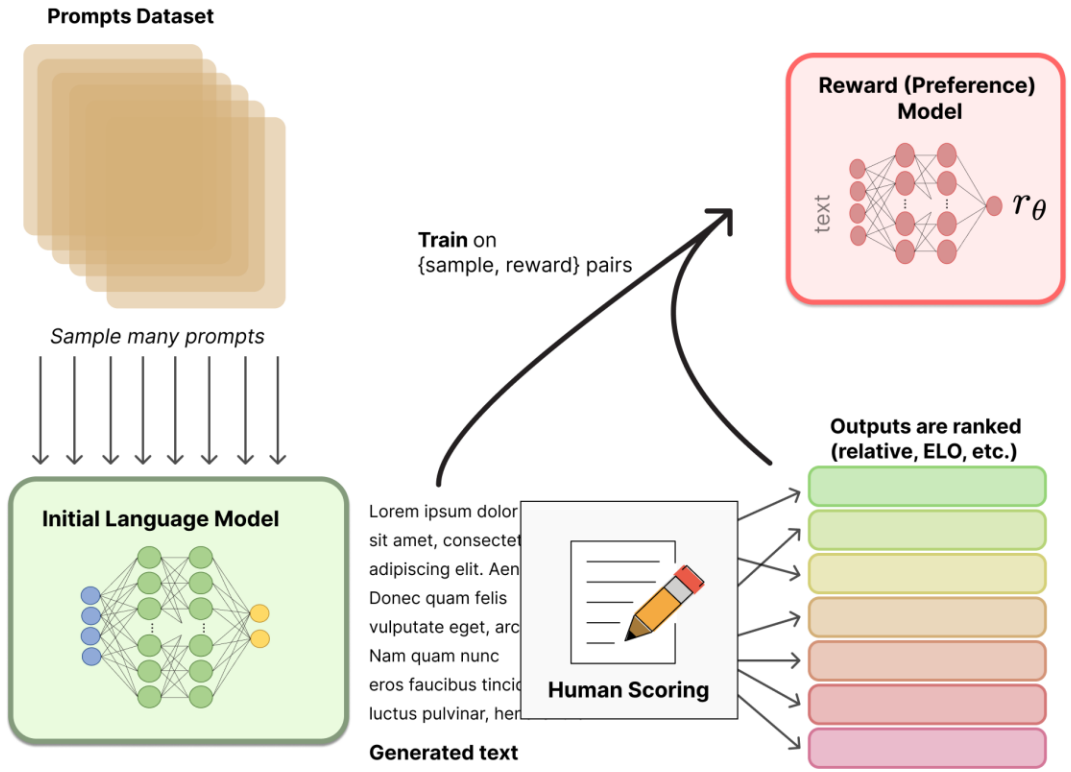
- Ask human users **which is better**
  - **Binary output**
  - Can do more
    - Rank more questions



RLHF: **Reward/Preference Model**

Second stage: train reward model

- Use the human feedback to train/fine-tune another model to reproduce the metric
- **Preference model**



https://huggingface.co/blog/rlhf

RLHF: **Reward/Preference Model**

Second stage: train reward model

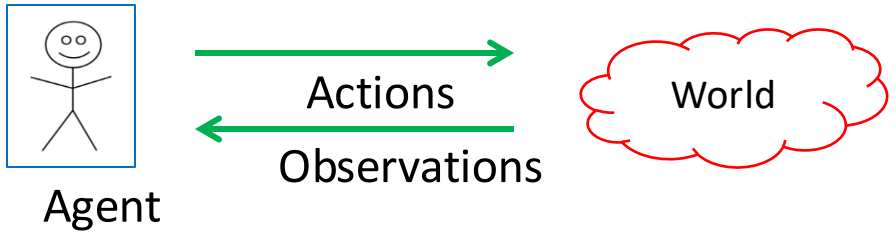
- Use the human feedback to train/fine-tune another model to reproduce the metric
- **Loss?** Based on preference models,
  - Example: Bradley-Terry model

$$p^*(y_1 \succ y_2 \mid x) = \frac{\exp\left(r^*(x, y_1)\right)}{\exp\left(r^*(x, y_1)\right) + \exp\left(r^*(x, y_2)\right)}.$$

  - Then, our reward model loss is based on the log likelihood,

$$\mathcal{L}_R(r_\phi, \mathcal{D}) = -\mathbb{E}_{(x, y_w, y_l) \sim \mathcal{D}} \left[\log \sigma(r_\phi(x, y_w) - r_\phi(x, y_l))\right]$$

RLHF: **Fine-Tuning with RL**



Third stage: RL

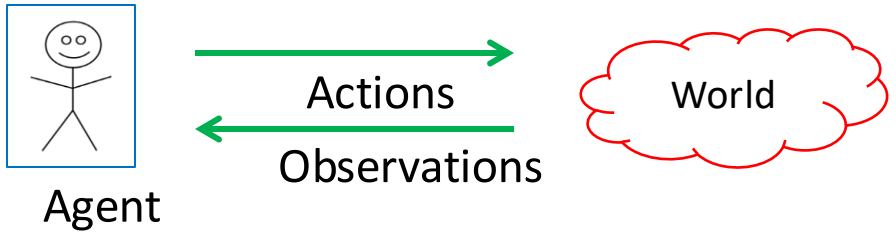
- Use an RL algorithm
- **Goal:** produce outputs that have high reward

RL formulation:

- **Action space**: all the tokens possible to output
- **State space**: all the sequences of tokens
- **Reward function**: the trained reward model
- **Policy**: the new version of the LM, taking in state and returning tokens

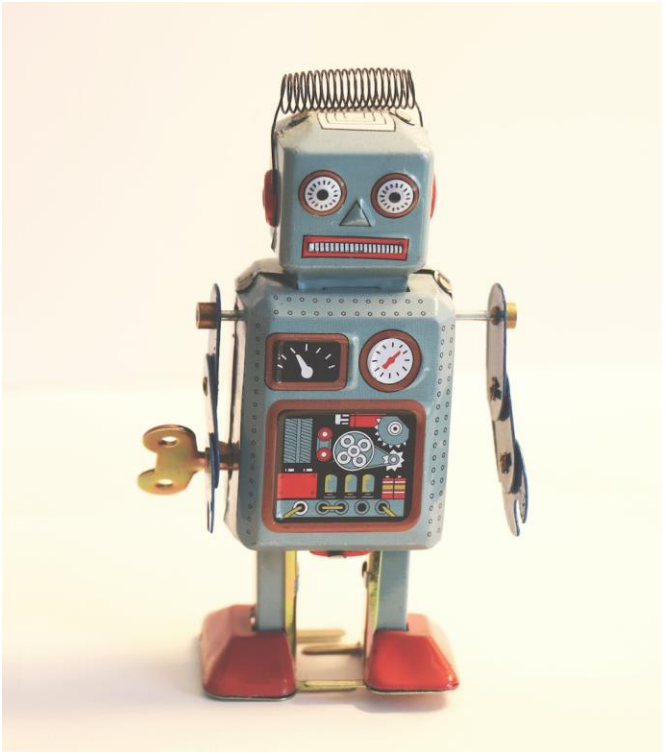
RLHF: **RL Approach**

What approach for RL stage?



- Many deep RL methods available
- Policy gradient methods
- Popular: PPO (Proximal Policy Optimization)
  - Main difference from vanilla policy gradient, you constrain change to policy at each step (Schulman et al)

$$\max_{\pi_\theta} \mathbb{E}_{x\sim\mathcal{D},y\sim\pi_\theta(y|x)}\left[r_\phi(x,y)\right] - \beta\mathbb{D}_{\mathrm{KL}}\left[\pi_\theta(y \mid x) \,||\, \pi_{\mathrm{ref}}(y \mid x)\right]$$

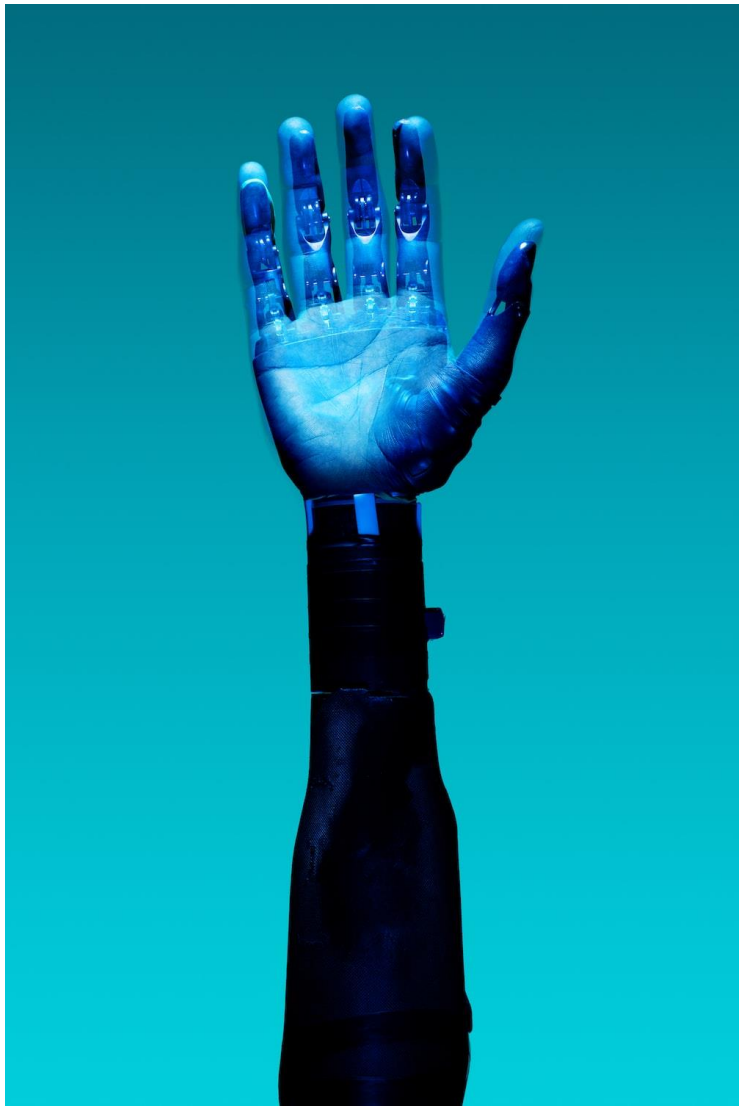


Why RLHF?

Why should we do this?

- Why does supervised fine-tuning by itself not give our goal results?

- Many hypotheses; this section inspired by Yoav Goldberg's blog:
  - https://gist.github.com/yoavg/6bff0fecd65950898eba1bb321cfbd81
  - Itself based on Schulman's talk
  - https://www.youtube.com/watch?v=hhiLw5Q_UFg



Why RLHF? **Ways To Interact**

Three “modes of interaction”:

- **text-grounded**: provide the model with text, instruction ("what are the chemical names mentioned in this text“),
- **knowledge-seeking**: provide the model with question or instruction, and expect a (truthful) answer based on the model's internal knowledge
- **creative**: provide the model with question or instruction, expect some creative output. ("Write a story about...")

Why RLHF? **Knowledge-seeking**

Three “modes of interaction”:

- **knowledge-seeking**: provide the model with question or instruction, and expect a (truthful) answer based on the model's internal knowledge

- This is hypothesized to require RL. Why does **SL fail?**
  - Case 1: know the answer: fine.
  - Case 2: don’t know the answer. Supervised learning forces memorization, cannot produce “don’t know”.
  - Worse, SL on case 2 encourages **model to lie…**

Why RLHF? **Knowledge-seeking with RL**

Three “modes of interaction”:

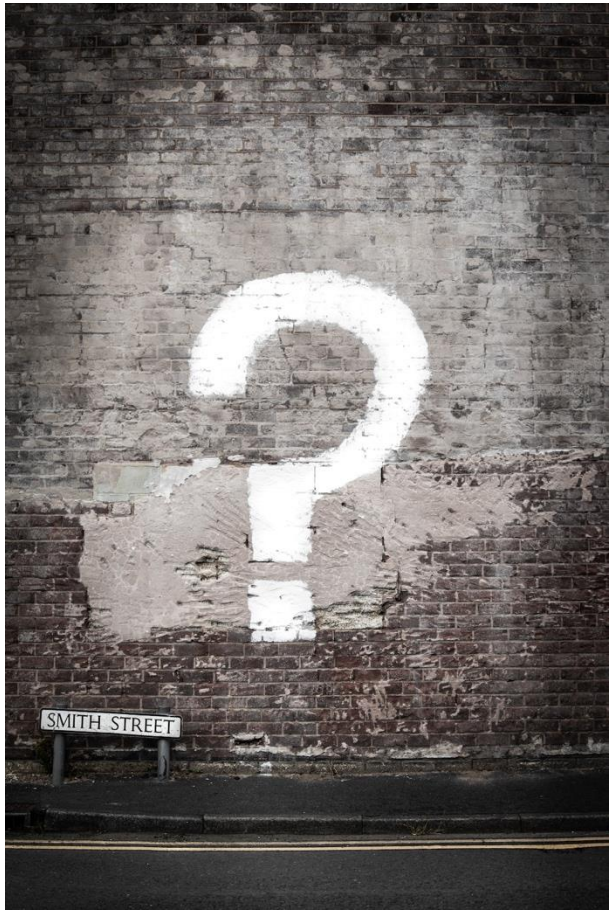
- **knowledge-seeking**: provide the model with question or instruction, and expect a (truthful) answer based on the model's internal knowledge

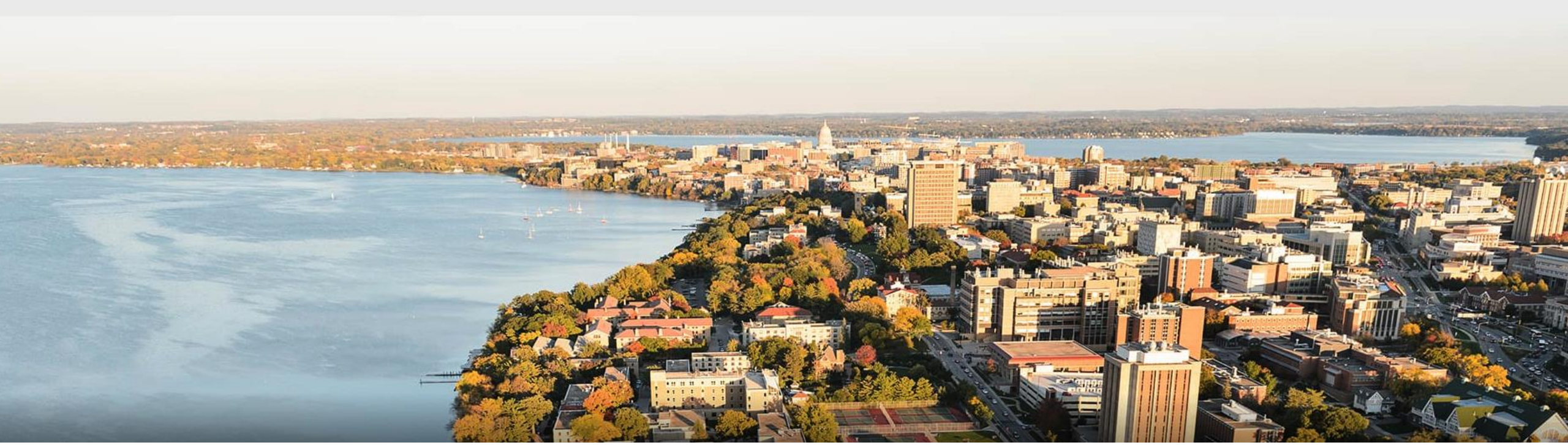
- Why does RL succeed?
  - Case 1: know the answer: fine. Get a reward
  - Case 2: don’t know the answer. Sometimes make it up and get a reward if lucky, most of the time low reward
  - **Encourages truth telling.**

Why RLHF? **Abstains**

Additionally, **we'd like our model to abstain**

- SL will really struggle with this
  - Usually no abstains in datasets
  - Even if there were, "generalization" here means abstaining on similar questions? Difficult
- RL still challenging, need to produce high reward for "don't know", but specific to model
- One way to craft a reward function:
  - High reward: correct answers
  - Medium reward: abstain
  - Negative reward: incorrect





Break & Questions

Outline

- **RLHF Review**
  - Motivation for alignment, overall pipeline, preference and reward models, PPO objective
- **Variations and Some Open Questions**
  - Direct preference optimization (DPO), RLAIF, other techniques
- **Verifiable Rewards**
  - Verifiers, applications, GRPO, extensions

RLHF Problems

Lots of challenges!

- **Casper et al**, “Open Problems and Fundamental Limitations of Reinforcement Learning from Human Feedback”

- Challenges everywhere, all three phases:
  - **In human feedback,**
  - **In obtaining reward model,**
  - **In obtaining the policy**

RLHF Problems: **Human Feedback**

- Need to obtain some kind of “representative” collection of feedback providers
- **Simpler:**
  - Some people have biases
  - Mistakes due to lack of care (standard in crowdsourcing)
  - Adversarial data poisoners
- **Harder:**
  - In tough settings, what is “good” output?
  - Possible to manipulate humans

RLHF Problems: **Human Feedback**

- Additionally, **need high-quality data**.
- Expensive to hand-craft good prompts to drive feedback

- Feedback quality:
  - Tradeoffs in feedback levels
  - Ideally, rich
  - But harder to work with to train reward

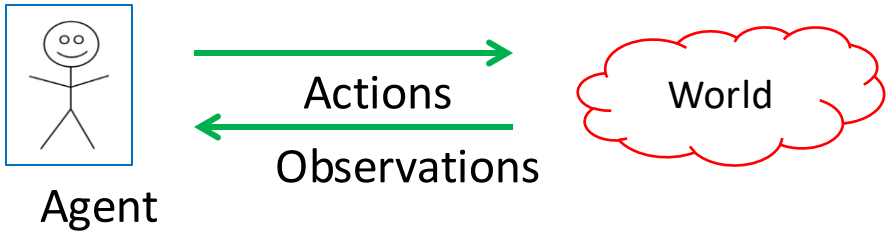
RLHF Problems: **Reward Model**

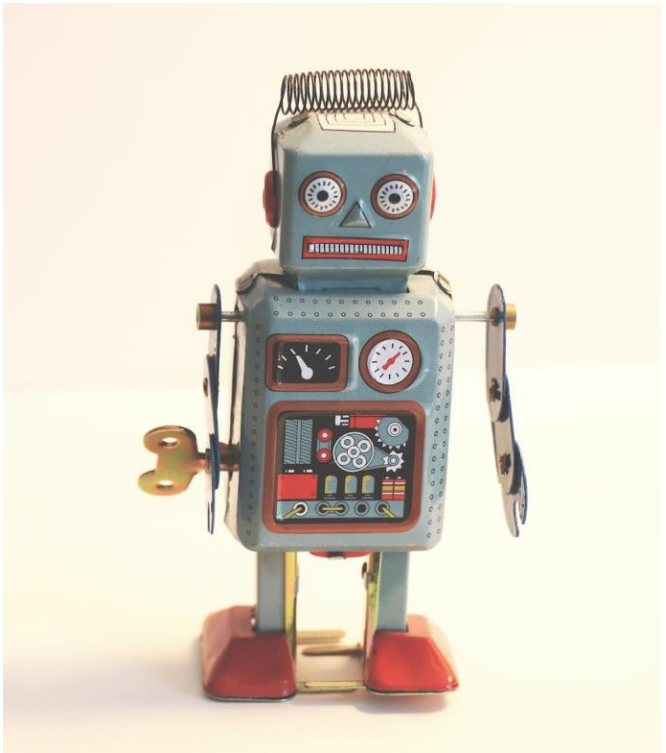
- Values can be difficult to express as a reward function
- May need to combine multiple reward functions:
  - What's a "universal" one? People are different

- Reward Hacking
  - In tough settings, what is "good" output?
  - Possible to manipulate humans

RLHF Problems: **Training**

- The RL in RLHF can be difficult
- Also, learned policies **do not necessarily generalize to other environments**





RLHF **Alternatives**

- **Direct preference optimization** (DPO)
  - Bypass separate trained reward model: just use preference information **directly** (Rafailov et al,'23)
  - **How?** Model a preference distribution from samples, integrate into a single loss (one-stage approach)

$$\mathcal{L}_{\text{DPO}}(\pi_\theta; \pi_{\text{ref}}) = -\mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}}\left[\log\sigma\left(\beta\log\frac{\pi_\theta(y_w \mid x)}{\pi_{\text{ref}}(y_w \mid x)} - \beta\log\frac{\pi_\theta(y_l \mid x)}{\pi_{\text{ref}}(y_l \mid x)}\right)\right].$$

  - **Gradient step:**

$$\nabla_\theta\mathcal{L}_{\text{DPO}}(\pi_\theta; \pi_{\text{ref}}) =$$

$$-\beta\mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}}\Big[\underbrace{\sigma(\hat{r}_\theta(x,y_l) - \hat{r}_\theta(x,y_w))}_{\text{higher weight when reward estimate is wrong}}\Big[\underbrace{\nabla_\theta\log\pi(y_w \mid x)}_{\text{increase likelihood of } y_w} - \underbrace{\nabla_\theta\log\pi(y_l \mid x)}_{\text{decrease likelihood of } y_l}\Big]\Big]$$

RLHF **Alternatives**

- **Many new approaches:**
  - A good survey: Ji et al ’24
- New approaches to rewards, new forms of feedback (including AI feedback), etc
- Popular research area!

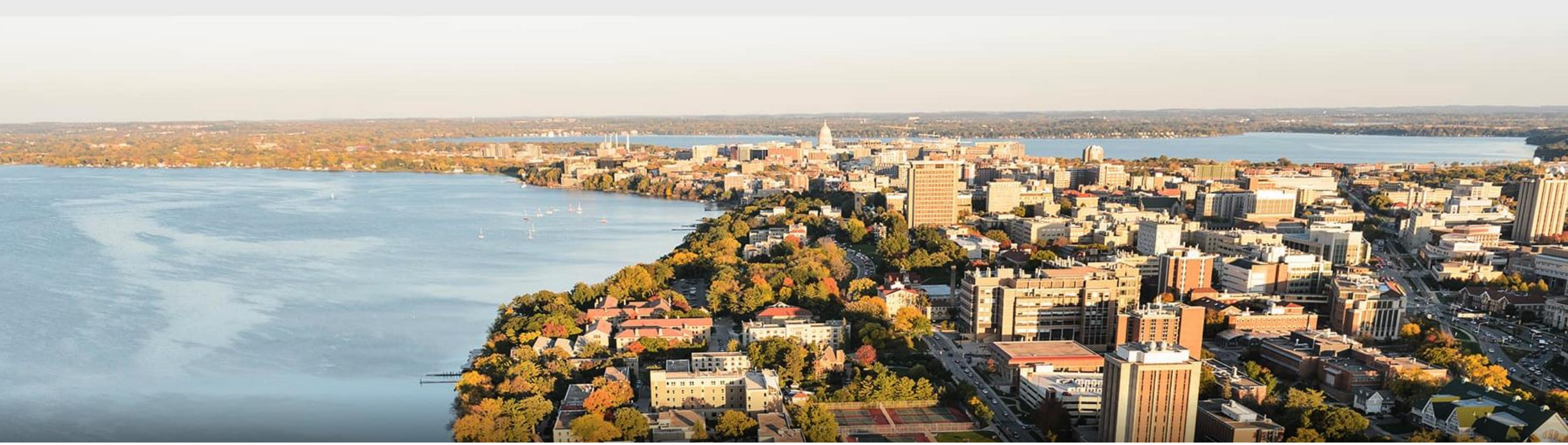
AI Alignment: A Comprehensive Survey

Jiaming Ji[*,1] Tianyi Qiu[*,1] Boyuan Chen[*,1] Borong Zhang[*,1] Hantao Lou[1] Kaile Wang[1] Yawen Duan[2] Zhonghao He[2] Jiayi Zhou[1] Zhaowei Zhang[1] Fanzhi Zeng[1] Juntao Dai[1] Xuehai Pan[1] Kwan Yee Ng Aidan O’Gara[5] Hua Xu[1] Brian Tse Jie Fu[4] Stephen McAleer[3] Yaodong Yang[1,✉] Yizhou Wang[1] Song-Chun Zhu[1] Yike Guo[4] Wen Gao[1]

[1]Peking University [2]University of Cambridge [3]Carnegie Mellon University
[4]Hong Kong University of Science and Technology [5]University of Southern California

Abstract

AI alignment aims to make AI systems behave in line with human intentions and values. As AI systems grow more capable, so do risks from misalignment. To provide a comprehensive and up-to-date overview of the alignment field, in this survey, we delve into the core concepts, methodology, and practice of alignment. First, we identify four principles as the key objectives of AI alignment: Robustness, Interpretability, Controllability, and Ethicality (**RICE**). Guided by these four principles, we outline the landscape of current alignment research and decompose them into two key components: **forward alignment** and **backward alignment**. The former aims to make AI systems aligned via alignment training, while the latter aims to gain evidence about the systems’ alignment and govern them appropriately to avoid exacerbating misalignment risks. On forward alignment, we discuss techniques for learning from feedback and learning under distribution shift. Specifically, we survey traditional preference modeling methods and reinforcement learning from human feedback, and further discuss potential frameworks to reach scalable oversight for tasks where effective human oversight is hard to obtain. Within learning under distribution shift, we also cover data distribution interventions such as adversarial training that help expand the distribution of



Break & Questions

Outline

- **RLHF Review**
  - Motivation for alignment, overall pipeline, preference and reward models, PPO objective
- **Variations and Some Open Questions**
  - Direct preference optimization (DPO), RLAIF, other techniques
- **Verifiable Rewards**
  - Verifiers, applications, GRPO, extensions

RL Outside of Alignment

- Let's get back to building **a good model**-doesn't need to be within the context of alignment
  - This means we don't have human preference data, but potentially something else

Where does RL fit in here?

- And what are the new reward models going to look like?
- One simple approach: "rewards" are just the correct answers
  - But, unlike in the supervised case, not just one solution

Verifiers

- Suppose we have a way of determining an answer is correct
  - Note: doesn't mean a full generation has to be identical to a ground-truth answer.

This is especially useful for e.g., math problems

Problem 12

The set of points in 3-dimensional coordinate space that lie in the plane $x+y+z=75$ whose coordinates satisfy the inequalities

$$x-yz<y-zx<z-xy$$

forms three disjoint convex regions. Exactly one of those regions has finite area. The area of this finite region can be expressed in the form $a\sqrt{b}$, where a and b are positive integers and b is not divisible by the square of any prime. Find $a+b$.

Reasoned about geometry and inequalities for 2m 47s ›

We will show that the (unique) finite region is a triangle. In fact, one may prove that among the four half-planes into which the two equalities

$$x-yz=y-zx \quad \text{and} \quad y-zx=z-xy,$$

split the plane $x+y+z=75$, exactly one of the four resulting sets is bounded. (In fact, a short analysis shows that writing

$$x-yz=y-zx \implies (y-x)(z-1)=0,$$

and

$$y-zx=z-xy \implies (z-y)(1+x)=0,$$

one deduces that the boundary lines are

$$L_1:\ z=1, \quad L_2:\ x=-1, \quad L_3:\ y=x, \quad L_4:\ y=z.$$

A careful discussion of the four cases – according to whether $z-1$ and $1+x$ are positive or negative – shows that the unique bounded (finite) region is given by

$$(y-x)(z-1)>0 \quad \text{and} \quad (z-y)(1+x)>0 \quad \text{with} \quad x+y+z=75,$$

which (when one accounts for the signs) is equivalent to the inequalities

$$1+x>0, \quad z-1>0, \quad y-x>0, \quad z-y>0,$$

i.e.

$$x>-1, \quad z>1, \quad x<y<z.$$

When one then "peels off" the boundary by considering the two equalities (which occur when either $y=x$ or $z=y$), one finds that the boundary lines of the finite region are exactly

$$x=-1, \quad y=x, \quad y=z.$$

A short computation shows that on the plane $x+y+z=75$ the three lines $x=-1$, $y=x$, and $y=z$ meet at the points

$$A=(-1,-1,77), \quad B=(25,25,25), \quad C=(-1,38,38).$$

These three points form a triangle that is the finite region in question.

To compute its area we may use the cross product. Define

$$\vec{AB}=B-A=(25-(-1),\ 25-(-1),\ 25-77)=(26,26,-52),$$

$$\vec{AC}=C-A=(-1-(-1),\ 38-(-1),\ 38-77)=(0,39,-39).$$

Their cross product is

$$\vec{AB}\times\vec{AC}=(1014,\ 1014,\ 1014),$$

so that

$$\|\vec{AB}\times\vec{AC}\|=1014\sqrt{3}.$$

Thus the area of the triangle is

$$\text{Area}=\frac{1}{2}\|\vec{AB}\times\vec{AC}\|=\frac{1014\sqrt{3}}{2}=507\sqrt{3}.$$

Since the area is given in the form $a\sqrt{b}$ with $a=507$ and $b=3$, the final answer is

$$a+b=507+3=510.$$

Thus, the answer is $\boxed{510}$.

Verifiers

- Note that verifiers don't need to just be answer checks
  - For example, we can write unit tests for code and use them for verification
  - Plus, lots of these out there!
  - As a result, much of **RLVR** is aimed at math and code

```
[TestMethod]
public void TestZoom()
{
  bool gestureDetected = false;
  var threadHolder = new AutoResetEvent(false);

  GestureTestFramework.Validate("Zoom", "TouchInteraction02",
    // On successful gesture detection
    (sender, e) =>
    {
      gestureDetected = true;
      if (e.Error == null)
      {
        var distanceChanged = e.Values.Get<DistanceChanged>();
        // User defined validation code
      }
      else
      {
        Assert.Fail(e.Error.Message);
      }
    });
```

Back to RL: PPO Details

- Note that we could directly apply PPO to train
- We would integrate some notion of verifier correctness into the reward
- Let's dive a bit deeper into PPO

$$\hat{\mathbb{E}}_t\left[\frac{\pi_\theta(a_t \mid s_t)}{\pi_{\theta_{\text{old}}}(a_t \mid s_t)}\hat{A}_t - \beta\,\mathrm{KL}[\pi_{\theta_{\text{old}}}(\cdot \mid s_t), \pi_\theta(\cdot \mid s_t)]\right]$$

- Two forms

(that we can combine)

Advantage $A_t = Q(s_t, a_t) - V(s_t)$

$$\hat{\mathbb{E}}_t\left[\min(r_t(\theta)\hat{A}_t, \mathrm{clip}(r_t(\theta), 1-\epsilon, 1+\epsilon)\hat{A}_t)\right]$$

PPO to GRPO

- GRPO (Group Relative Policy Optimization)
  - Shao et al, DeepSeekMath

$$\mathcal{J}_{GRPO}(\theta) = \mathbb{E}[q \sim P(Q), \{o_i\}_{i=1}^{G} \sim \pi_{\theta_{old}}(O|q)]$$
$$\frac{1}{G}\sum_{i=1}^{G}\left(\min\left(\frac{\pi_\theta(o_i|q)}{\pi_{\theta_{old}}(o_i|q)}A_i, \text{clip}\left(\frac{\pi_\theta(o_i|q)}{\pi_{\theta_{old}}(o_i|q)}, 1-\varepsilon, 1+\varepsilon\right)A_i\right) - \beta\mathbb{D}_{KL}\left(\pi_\theta||\pi_{ref}\right)\right).$$

- Most elements are the same compared to PPO, but note that we sample a **group** of G responses.
- Advantage:

$$A_i = \frac{r_i - \text{mean}(\{r_1, r_2, \cdots, r_G\})}{\text{std}(\{r_1, r_2, \cdots, r_G\})}$$

GRPO/DeepSeek R1 Rewards

- How to use verifiers in rewards?

$$A_i = \frac{r_i - \text{mean}(\{r_1, r_2, \cdots, r_G\})}{\text{std}(\{r_1, r_2, \cdots, r_G\})}$$

Very simple: DeepSeek R1 uses:

- **Accuracy rewards**: The accuracy reward model evaluates whether the response is correct. For example, in the case of math problems with deterministic results, the model is required to provide the final answer in a specified format (e.g., within a box), enabling reliable rule-based verification of correctness. Similarly, for LeetCode problems, a compiler can be used to generate feedback based on predefined test cases.
- **Format rewards**: In addition to the accuracy reward model, we employ a format reward model that enforces the model to put its thinking process between '<think>' and '</think>' tags.

Note the thinking tokens!

Strong Performance on Math

AIME Results:

Overall | **AIME 2025 I** | AIME 2025 II | HMMT February 2025 | USAMO 2025

Model	Acc	Cost	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
gemini-2.5-pro ⚠	83.33%	N/A															
o3-mini (high)	80.00%	$3.19															
o1 (medium)	78.33%	$44.40															
o3-mini (medium)	73.33%	$1.67															
DeepSeek-R1	65.00%	$4.91															
QwQ-32B ⚠	60.00%	$1.24															
DeepSeek-V3-03-24 ⚠	53.33%	$0.25															
o3-mini (low)	53.33%	$0.62															
DeepSeek-R1-Distill-32B	53.33%	N/A															
gemini-2.0-flash-thinking	51.67%	N/A															
DeepSeek-R1-Distill-14B	50.00%	$1.15															
DeepSeek-R1-Distill-70B	50.00%	$1.35															
Claude-3.7-Sonnet (Think) ⚠	46.67%	$22.17															
QwQ-32B-Preview	36.67%	$0.58															
gemini-2.0-flash	30.00%	$0.06															

https://matharena.ai/

And Physics

Theoretical Physics Benchmark (TPBench) - a Dataset and Study of AI Reasoning Capabilities in Theoretical Physics

Daniel J.H. Chung[1], Zhiqi Gao[2], Yurii Kvasiuk[1], Tianyi Li[1], Moritz Münchmeyer[1,5], Maja Rudolph[3], Frederic Sala[2], and Sai Chaitanya Tadepalli[4]

[1]Department of Physics, University of Wisconsin-Madison
[2]Department of Computer Science, University of Wisconsin-Madison
[3]Data Science Institute (DSI), University of Wisconsin-Madison
[4]Department of Physics, Indiana University, Bloomington
[5]NSF-Simons AI Institute for the Sky (SkAI), Chicago

February 25, 2025

Abstract

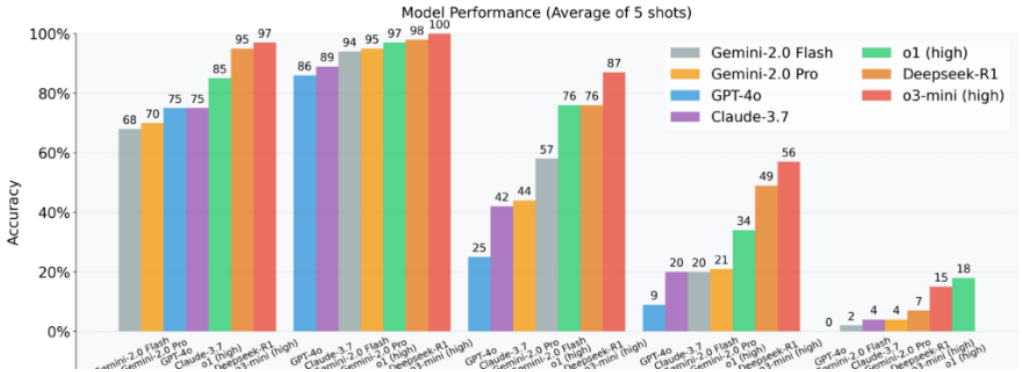
We introduce a benchmark to evaluate the capability of AI to solve problems in theoretical physics, focusing on high-energy theory and cosmology. The first iteration of our benchmark consists of 57 problems of varying difficulty, from undergraduate to research level. These problems are novel in the sense that they do not come from public problem collections. We evaluate our data set on various open and closed language models, including o3-mini, o1, DeepSeek-R1, GPT-4o and versions of Llama and Qwen. While we find impressive progress in model performance with the most recent models, our research-level difficulty problems are mostly unsolved. We address challenges of auto-verifiability and grading, and discuss common failure modes. While currently state-of-the art models are still of limited use for researchers, our results show that AI assisted theoretical physics research may become possible in the near future. We discuss the main obstacles towards this goal and possible strategies to overcome them. The public problems and solutions, results for various models, and updates to the data set and score distribution, are available on the website of the dataset tpbench.org.

TP Bench – Theoretical Physics Benchmark for AI

TPBench is a curated dataset and evaluation suite designed to measure the reasoning capabilities of AI models in theoretical physics. Our test problems span multiple difficulty levels—from undergraduate to frontier research—and cover topics such as cosmology, high-energy theory, general relativity, and more. By providing a unified framework for problem-solving and auto-verifiable answers, TPBench aims to drive progress in AI-based research assistance for theoretical physics.

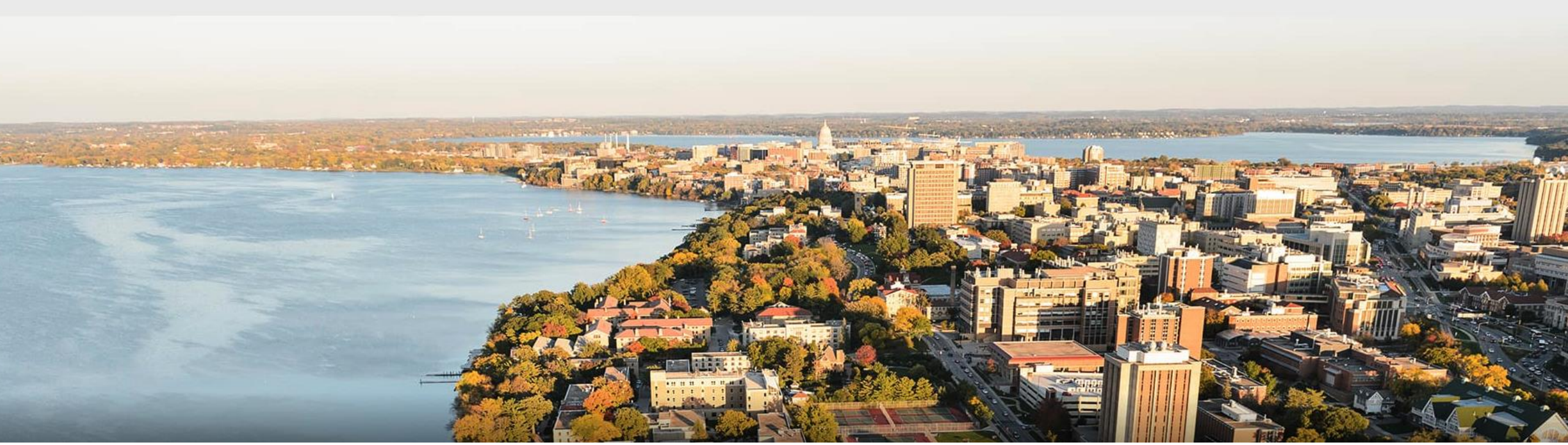
Read the TPBench Paper on arxiv
Access Public Dataset on Huggingface

Current Model Performance



Bibliography

- Chip Huyen: https://huyenchip.com/2023/05/02/rlhf.html
- Nathan Lambert et al: https://huggingface.co/blog/rlhf
- Ouyang et al '22: Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, Ryan Lowe, "Training language models to follow instructions with human feedback" (https://arxiv.org/abs/2203.02155)
- Lambert et al '24: Nathan Lambert, Valentina Pyatkin, Jacob Morrison, LJ Miranda, Bill Yuchen Lin, Khyathi Chandu, Nouha Dziri, Sachin Kumar, Tom Zick, Yejin Choi, Noah A. Smith, Hannaneh Hajishirzi , "RewardBench: Evaluating Reward Models for Language Modeling" (https://arxiv.org/abs/2403.13787)
- Schulman et al: John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, Oleg Klimov, "Proximal Policy Optimization Algorithms" (https://arxiv.org/abs/1707.06347)
- Yoav Golderbg: https://gist.github.com/yoavg/6bff0fecd65950898eba1bb321cfbd81
- Casper et al: Stephen Casper, Xander Davies, and many others, "Open Problems and Fundamental Limitations of Reinforcement Learning from Human Feedback" (https://arxiv.org/abs/2307.15217)
- Rafailov et al '23: Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, Chelsea Finn, "Direct Preference Optimization: Your Language Model is Secretly a Reward Model" (https://arxiv.org/abs/2305.18290)
- Ji et al '24: Jiaming Ji, Tianyi Qiu, Boyuan Chen, Borong Zhang, Hantao Lou, Kaile Wang, Yawen Duan, Zhonghao He, Jiayi Zhou, Zhaowei Zhang, Fanzhi Zeng, Kwan Yee Ng, Juntao Dai, Xuehai Pan, Aidan O'Gara, Yingshan Lei, Hua Xu, Brian Tse, Jie Fu, Stephen McAleer, Yaodong Yang, Yizhou Wang, Song-Chun Zhu, Yike Guo, Wen Gao, "AI Alignment: A Comprehensive Survey" (https://arxiv.org/abs/2310.19852)
- Shao et al '24: "DeepSeekMath: Pushing the Limits of Mathematical Reasoning in Open Language Models" (https://arxiv.org/abs/2402.03300)
- Guo et al/DeepSeek Team '25: "DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning" (https://arxiv.org/abs/2501.12948)



Thank You!