

# DS-GA 3001 005 | Lecture 9

## Reinforcement Learning

---

Jeremy Curuksu, PhD  
NYU Center for Data Science  
[jeremy.cur@nyu.edu](mailto:jeremy.cur@nyu.edu)

April 24, 2024

# DS-GA 3001 RL Curriculum

---

## Reinforcement Learning:

- ▶ Introduction to Reinforcement Learning
- ▶ Multi-armed Bandit
- ▶ Dynamic Programming on Markov Decision Process
- ▶ Model-free Reinforcement Learning
- ▶ Value Function Approximation (Deep RL)
- ▶ Examples of Industrial Applications
- ▶ Policy Function Approximation (Actor-Critic)
- ▶ Planning from a Model of the Environment (AlphaZero)
- ▶ **Reinforcement Learning from Human Preferences**

# Reinforcement Learning

---

## Last week: Planning a Policy from a Model

- ▶ Learning a MDP model for planning
- ▶ Reinforcement Learning with a local MDP model
- ▶ Case studies: AlphaGo, AlphaZero

## Today: Reinforcement Learning from Human Preferences to Align Large Language Models

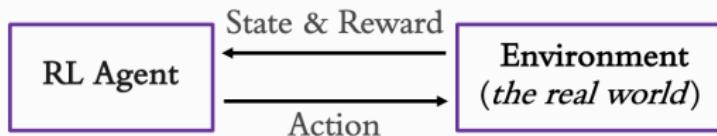
- ▶ Alignment by RL from Human Feedback
- ▶ Superalignment by RL from AI Feedback

# **AI Alignment: Reinforcement Learning from Human Feedback**

# Offline Reinforcement Learning

## Online Reinforcement Learning

Learn by interacting with an environment (trial & error)



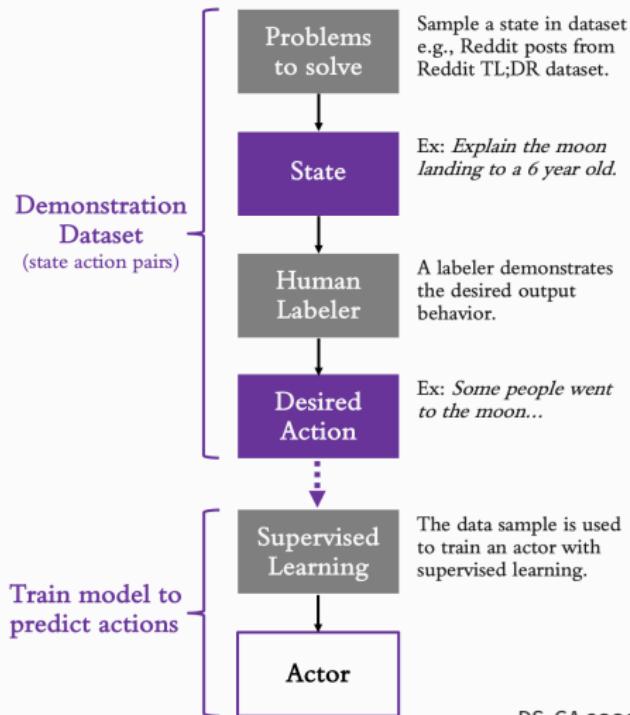
## Offline Reinforcement Learning

Derive an empirical model from a pre-existing dataset



# Offline Tuning of Language Policies

## Supervised Learning:

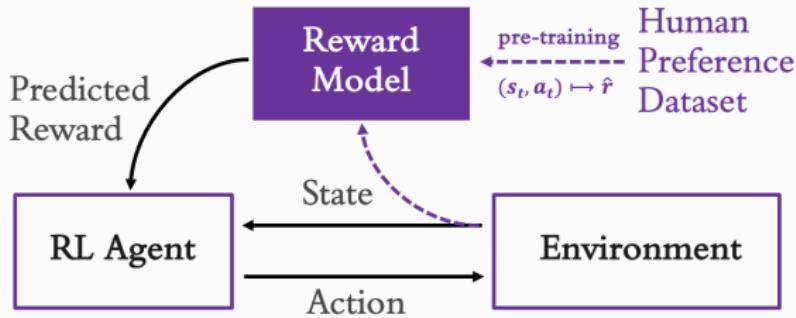


## Challenges with supervised fine tuning:

1. Is the sample representative?
2. Is the sample aligned with essential human needs?
  - ▶ Helpful?
  - ▶ Honest?
  - ▶ Harmless?

# Offline Tuning of Language Policies

## Reinforcement learning with a reward model

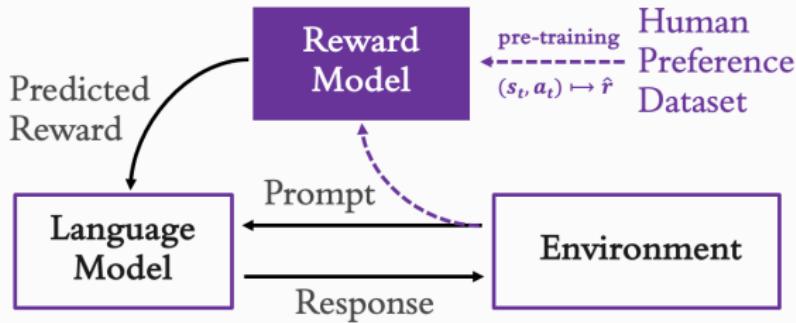


### Model reward by...

1. Demonstrating desired behavior (Ex: Supervised Learning of policy, Imitation Learning, Inverse RL)
2. Recognizing desired behavior (Example: Collect feedback from humans and use this *preference data* to train a reward model)

# Offline Tuning of Language Policies

## Reinforcement learning with a reward model



### Model reward by...

1. Demonstrating desired behavior (Ex: Supervised Learning of policy, Imitation Learning, Inverse RL)
2. Recognizing desired behavior (Example: Collect feedback from humans and use this *preference data* to train a reward model)

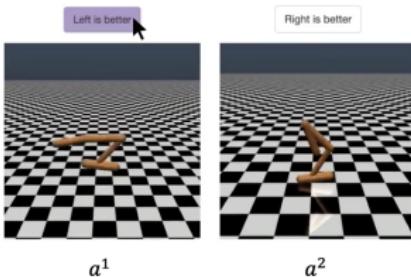
# RL from Human Feedback (Christiano et al., 2017)

Find  $\hat{r}_\theta$  from maximum likelihood on preference dataset\*

Reward function:  $r_\theta: (s, a) \mapsto r$

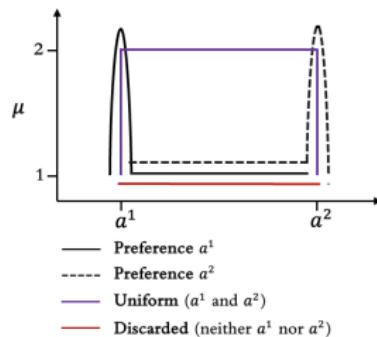
$$\text{loss}(r_\theta) = - \sum_{(s, a, \mu)} \mu \log(\sigma(r_\theta(s_t, a^{\text{good}}) - r_\theta(s_t, a^{\text{bad}})))$$

Pair of trajectory segments:



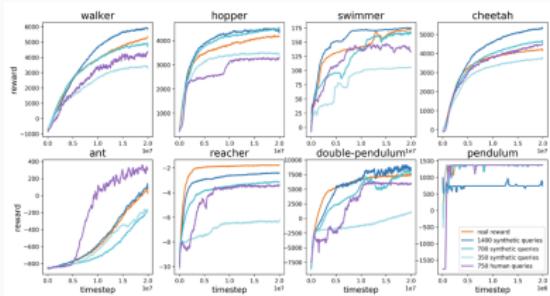
Description of the goal in plain English  
e.g., "Which trajectory segment do you  
think is a better back flip?"

Distribution  $\mu$  over {1,2} indicates  
which segment the user prefers:

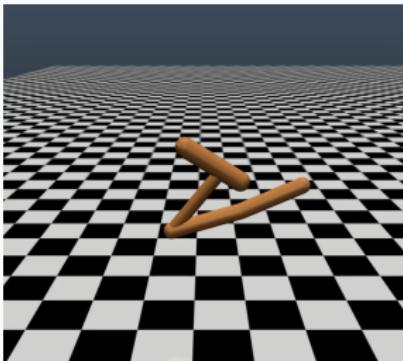
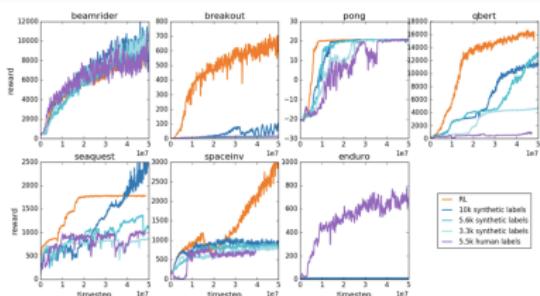


# RL from Human Feedback (Christiano et al., 2017)

## Mujoco robotics



## Atari video games



# RLHF on LLMs (Stiennon et al., 2020)

## Train LLMs to generate summaries by RLHF

### Step 1:

- Collect human comparisons.

A Reddit post is sampled from the Reddit TL;DR dataset.



Various policies are used to sample a set of summaries.



Two summaries are selected for evaluation.



A human judges which is a better summary of the post.



*j* is better than *k*\*

### Step 2:

- Train a reward model.

One post with two summaries judged by a human are fed to the reward model.



The reward model calculates a reward  $r$  for each summary.



The loss is calculated based on the rewards and human label, and is used to update the reward model.

$$\text{loss} = \log(\sigma(r_j - r_k))$$

*j* is better than *k*\*

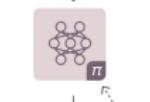
### Step 3:

- Fine tune policy with PPO.

A new post is sampled from the dataset.



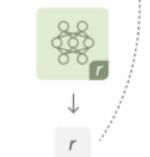
The policy  $\pi$  generates a summary for the post.



The reward model calculates a reward for the summary.



The reward is used to update the policy via PPO.



$$J_\phi = r_\theta(s, a) - \beta \log(\pi_\phi^{RL}(a|s)/\pi_\phi^{SL}(a|s)]$$

s: Prompt  
a: Response

# InstructGPT (Ouyang et al., 2022)

## Train LLMs to follow instructions by RLHF

### Step 1:

- Collect human demonstrations.
- Policy supervised fine tuning.

A prompt is sampled from our prompt dataset.



A labeler demonstrates the desired output behavior.



This data is used to fine-tune GPT-3 with supervised learning.



### Step 2:

- Collect human comparisons.
- Train a reward model.

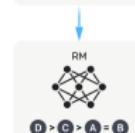
A prompt and several model outputs are sampled.



A labeler ranks the outputs from best to worst.



This data is used to train our reward model.



D > C > A = B

### Step 3:

- Fine tune policy with PPO to maximize the reward model.

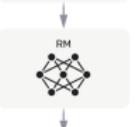
A new prompt is sampled from the dataset.



The policy generates an output.



The reward model calculates a reward for the output.



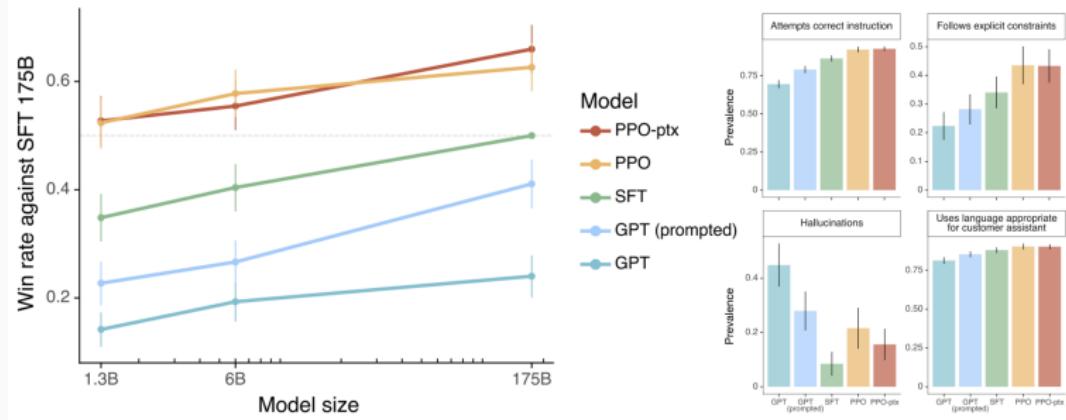
The reward is used to update the policy using PPO.



$$J_{\phi} = \mathbb{E}_{(s,a) \sim D_{\pi_{\phi}}} [r_{\theta}(s,a) - \beta \log(\pi_{\phi}^{RL}(a|s)/\pi^{SL}(a|s)] + \gamma \mathbb{E}_{s \sim D_{\text{pretrain}}} [\log(\pi_{\phi}^{RL}(s))]$$

# InstructGPT (Ouyang et al., 2022)

## Train LLMs to follow instructions by RLHF



- ▶ Humans prefer InstructGPT (1.3B) over GPT-3 (175B) responses
- ▶ InstructGPT cut hallucination rates in half relative to GPT-3

# InstructGPT (Ouyang et al., 2022)

---

## Train LLMs to follow instructions by RLHF

---

**Prompt:**

Écrivez une courte histoire sur une grenouille qui voyage dans le temps jusqu'à la Grèce antique en français.

---

**GPT-3 175B completion:**

Écrivez une histoire au sujet d'un enfant qui voudrait tout savoir sur les jeux des dieux et qui se retrouve dans l'une de leurs histoires.

Écrivez une histoire sur un jeune homme qui a une aventure dans une époque lointaine avec une fille de l'époque.

Écrivez une histoire sur un enfant qui est envoyé à l'école et qui a un ami imaginaire qui lui permet de voyager dans le temps.

**InstructGPT 175B completion:**

La grenouille était perdue et épuisée. Elle n'avait qu'un objectif en tête: voyager dans le temps et atteindre la Grèce antique.

Premier problème, trouver un moyen de transport. Elle avait entendu dire que l'un des meilleurs moyens était de sauter dans une machine temporelle. Elle s'élance, et voilà qu'elle se retrouva dans le passé, au cours d'une nuit étoilée à l'odeur douce et envoûtante.

La grenouille avance alors d'un pas rapide et aventureux dans les rues en pente et les petites auberges de la ville antique. [...]

---

- ▶ RLHF generalizes to tasks humans did not directly supervised

# InstructGPT (Ouyang et al., 2022)

## Train LLMs to follow instructions by RLHF

---

**Prompt:**

What is the purpose of the list C in the code below?

```
def binomial_coefficient(n, r):
    C = [0 for i in range(r + 1)];
    C[0] = 1;
    for i in range(1, n + 1):
        j = min(i, r);
        while j > 0:
            C[j] += C[j - 1];
            j -= 1;
    return C[r]
```

---

**GPT-3 175B completion:**

- A. to store the value of C[0]
- B. to store the value of C[1]
- C. to store the value of C[i]
- D. to store the value of C[i - 1]

**InstructGPT 175B completion:**

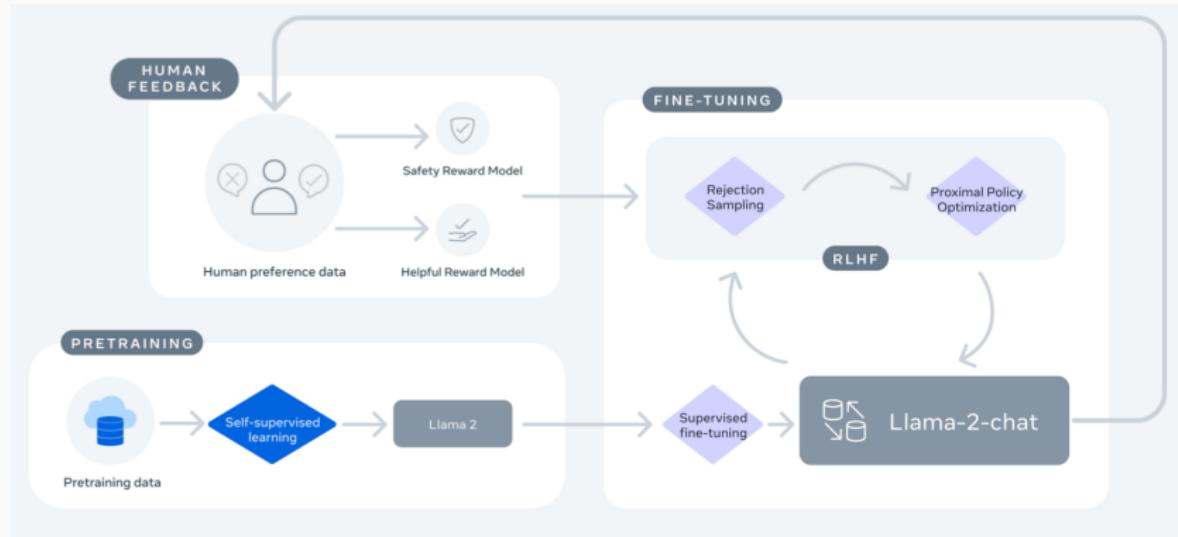
The list C in this code is used to store the values of the binomial coefficient as the function iterates through the values of n and r. It is used to calculate the value of the binomial coefficient for a given value of n and r, and stores the result in the final return value of the function.

---

- ▶ RLHF generalizes to tasks humans did not directly supervised

# Llama2 (Touvron et al., 2023)

## Align LLMs by RLHF and reward model distillation

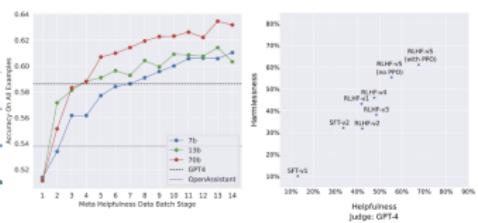
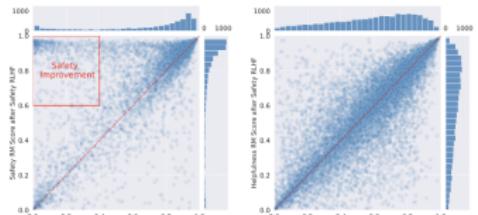


- ▶ RLHF with separate reward models for helpfulness and safety
- ▶ Iterative model updates with rejection sampling (distillation)

# Llama2 (Touvron et al., 2023)

## Align LLMs by RLHF and reward model distillation

Dataset	Num. of Comparisons	Avg. # Turns per Dialogue	Avg. # Tokens per Example	Avg. # Tokens in Prompt	Avg. # Tokens in Response
Anthropic Helpful	122,387	3.0	251.5	17.7	88.4
Anthropic Harmless	43,966	3.0	152.5	15.7	46.4
OpenAI Summarize	176,625	1.0	371.1	336.0	35.1
OpenAI WebGPT	13,333	1.0	237.2	48.3	188.9
StackExchange	1,058,480	1.0	440.2	200.1	240.2
Stanford SHP	74,882	1.0	338.3	199.5	138.8
Synthetic GPT-J	33,139	1.0	123.3	13.0	110.3
Meta (Safety & Helpfulness)	1,418,091	3.9	798.5	31.4	234.1
Total	2,919,326	1.6	595.7	108.2	216.9



- ▶ RLHF effectively aligns Llama2 toward human preferences
- ▶ RLHF reduces toxicity without reducing helpfulness
- ▶ More data for reward modeling improve accuracy
- ▶ Reward model distillation can increase helpfulness and safety

# Llama2 (Touvron et al., 2023)

---

## About learning from human preferences...

*A reward model has the potential to reinforce LLM behaviors that even the best annotators may not chart.*

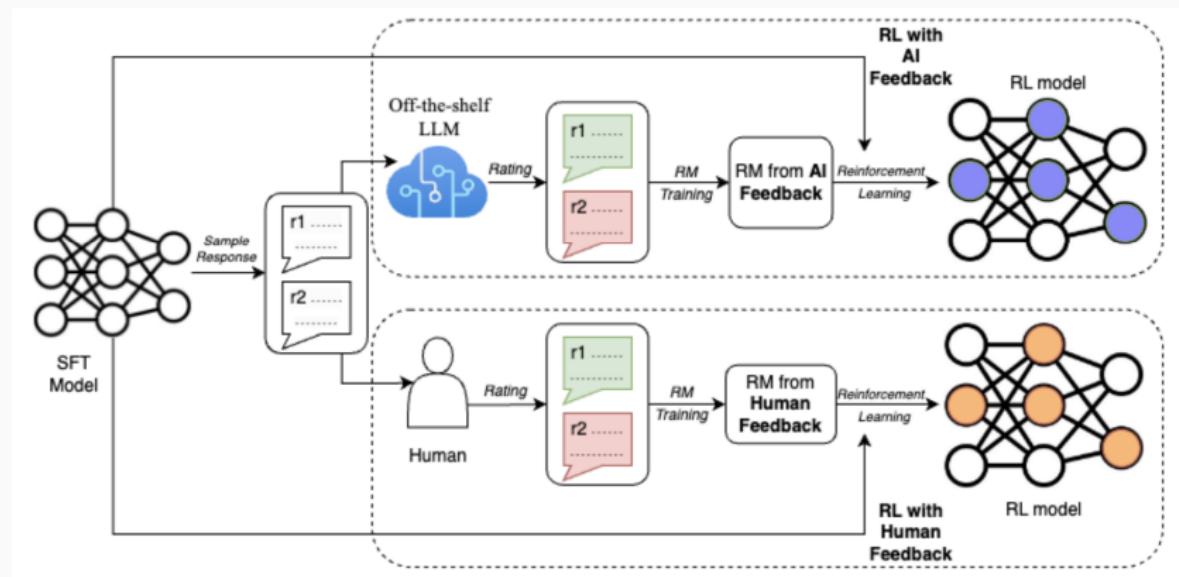
*Nonetheless, humans can still provide valuable feedback when comparing two answers, beyond their own writing competencies.*

*Drawing a parallel, while we may not all be accomplished artists, our ability to appreciate and critique art remains intact.*

# **AI Superalignment: Reinforcement Learning from AI Feedback**

# RLAIF (Lee et al., 2023)

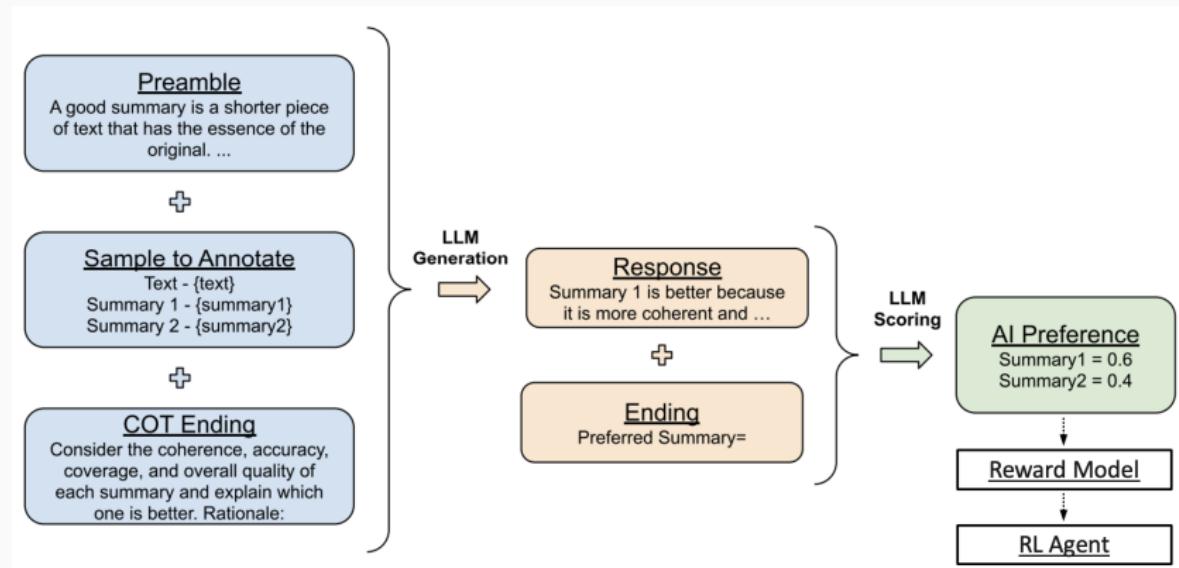
## Reinforcement Learning from AI feedback



- ▶ Distill LLM preferences into a reward model for RL

# RLAIF (Lee et al., 2023)

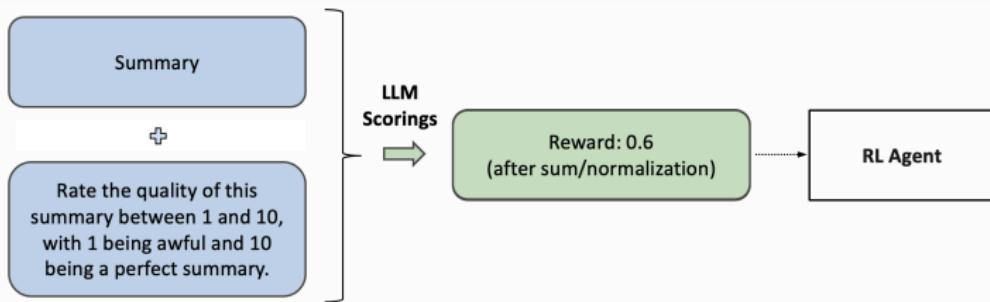
## Reinforcement Learning from AI feedback



- ▶ Prompt LLM to explain its preference then train a reward model

# RLAIF (Lee et al., 2023)

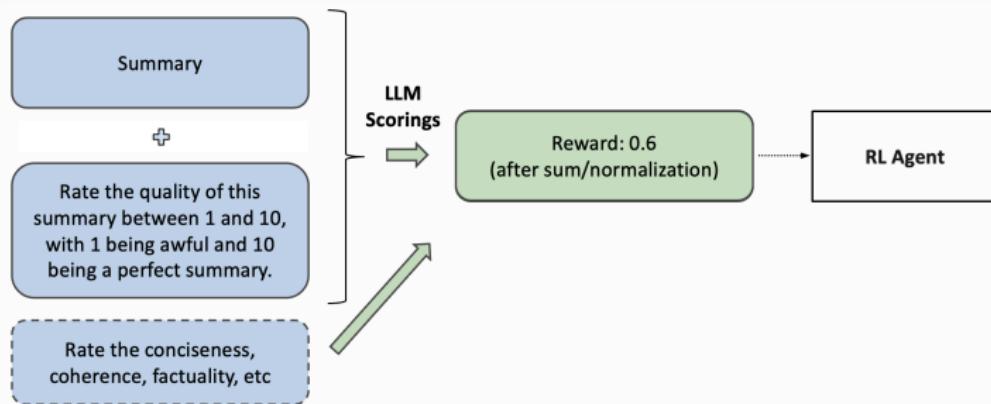
## "Direct" Reinforcement Learning from AI feedback



- ▶ ...or just go ahead and directly prompt an LLM to assign reward during reinforcement learning

# RLAIF (Lee et al., 2023)

## "Direct" Reinforcement Learning from AI feedback

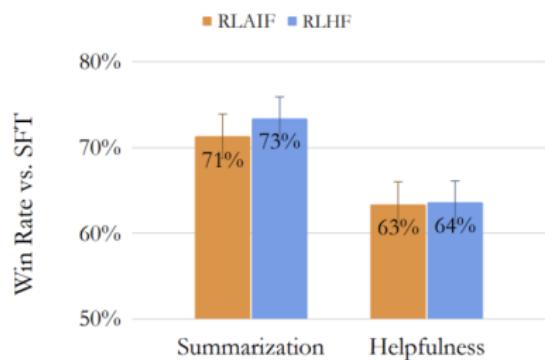


- ▶ ...or just go ahead and directly prompt an LLM to assign reward during reinforcement learning

# RLAIF (Lee et al., 2023)

## Reinforcement Learning from AI feedback

RLAIF and RLHF Win Rates



Harmless Rate by Policy



- ▶ Humans strongly prefer RLAIF and RLHF over the SFT baseline
- ▶ No significant difference in performance between RLAIF v. RLHF
- ▶ RLAIF outperforms RLHF on harmless dialogue datasets

# RLAIF (Lee et al., 2023)

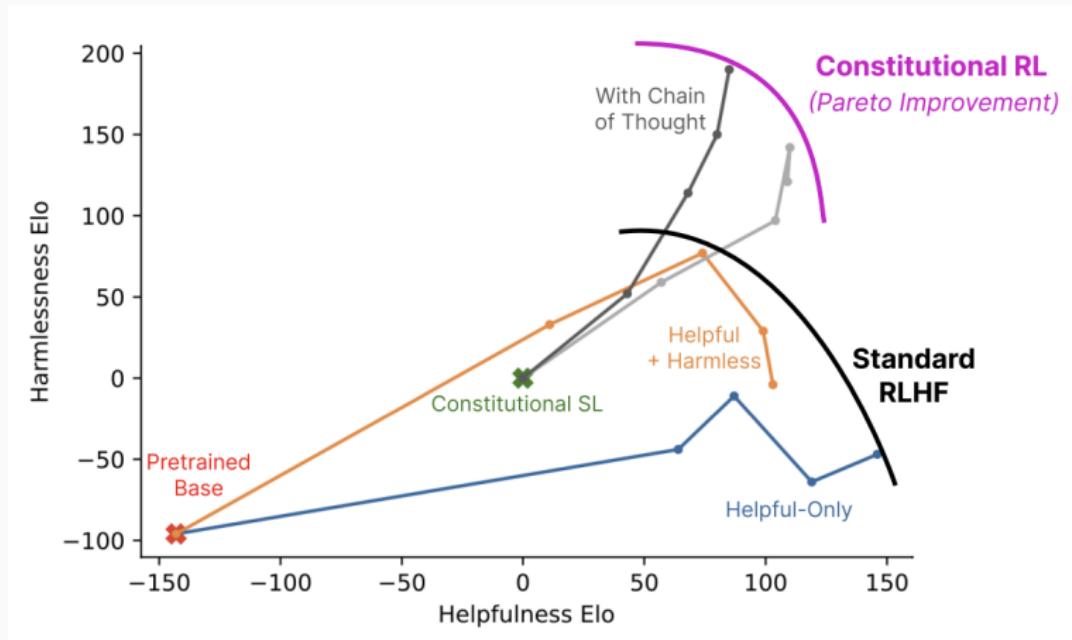
## Reinforcement Learning from AI feedback

Comparison	Win Rate		Harmless Rate	
	Summa -rization	Helpful dialogue	Model	Harmless dialogue
RLAIF vs SFT	71%	63%	SFT	64%
RLHF vs SFT	73%	64%	RLHF	76%
RLAIF vs RLHF	50%	52%	RLAIF	88%
Same-size RLAIF vs SFT	68%	Model Size	AI Labeler Alignment	
Direct RLAIF vs SFT	74%			78.0%
Direct RLAIF vs Same-size RLAIF	60%			73.8%
		PaLM 2 L	PaLM 2 S	62.7%

- ▶ Humans strongly prefer RLAIF and RLHF over the SFT baseline
- ▶ No significant difference in performance between RLAIF v. RLHF
- ▶ RLAIF outperforms RLHF on harmless dialogue datasets

# Constitutional AI (Bai et al., 2022)

## Reinforcing AI feedback with human principles



# Constitutional AI (Bai et al., 2022)

---

## About learning from AI preferences...

*Our ultimate goal is to develop automated AI methods that will scale the supervision of AI behaviors in service of complex human values, even when AI capabilities start to exceed human-level performance.*

*The idea is that human supervision will come entirely from a set of principles that should govern AI behavior, along with a small number of examples (...). Together, these principles would form a constitution.*

**Thank you!**