Washington University in St. Louis
James McKelvey School of Engineering

CSE 561A: Large Language Models

Spring 2024

Lecture 4: Reinforcement Learning from Human Feedback

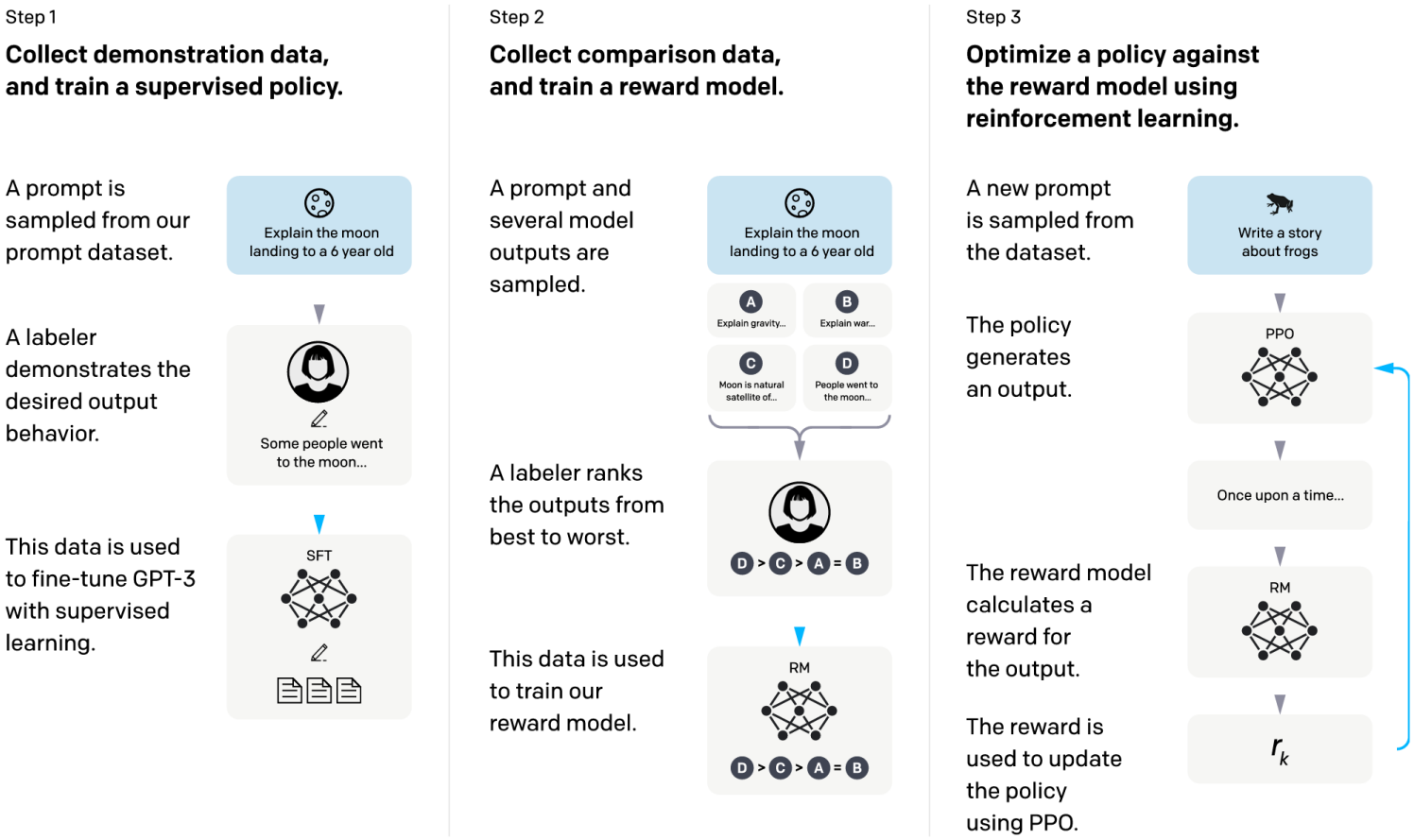
Jiaxin Huang

Course Announcements

- All the students in the waitlist have been enrolled!
- The sign-up sheet is out: https://docs.google.com/spreadsheets/d/1xSCaIOjiri17V7IjP2di-kFwbOgInPb_azBfZKeTgmc/edit
- The first student presentation lecture is on next Thursday (Feb.1st)
- Presentation Duration: 30-35 min
- Presenters (on Feb.1st) please send your slides to me (cc the TAs) before Monday 12:00PM (Jan. 29th)

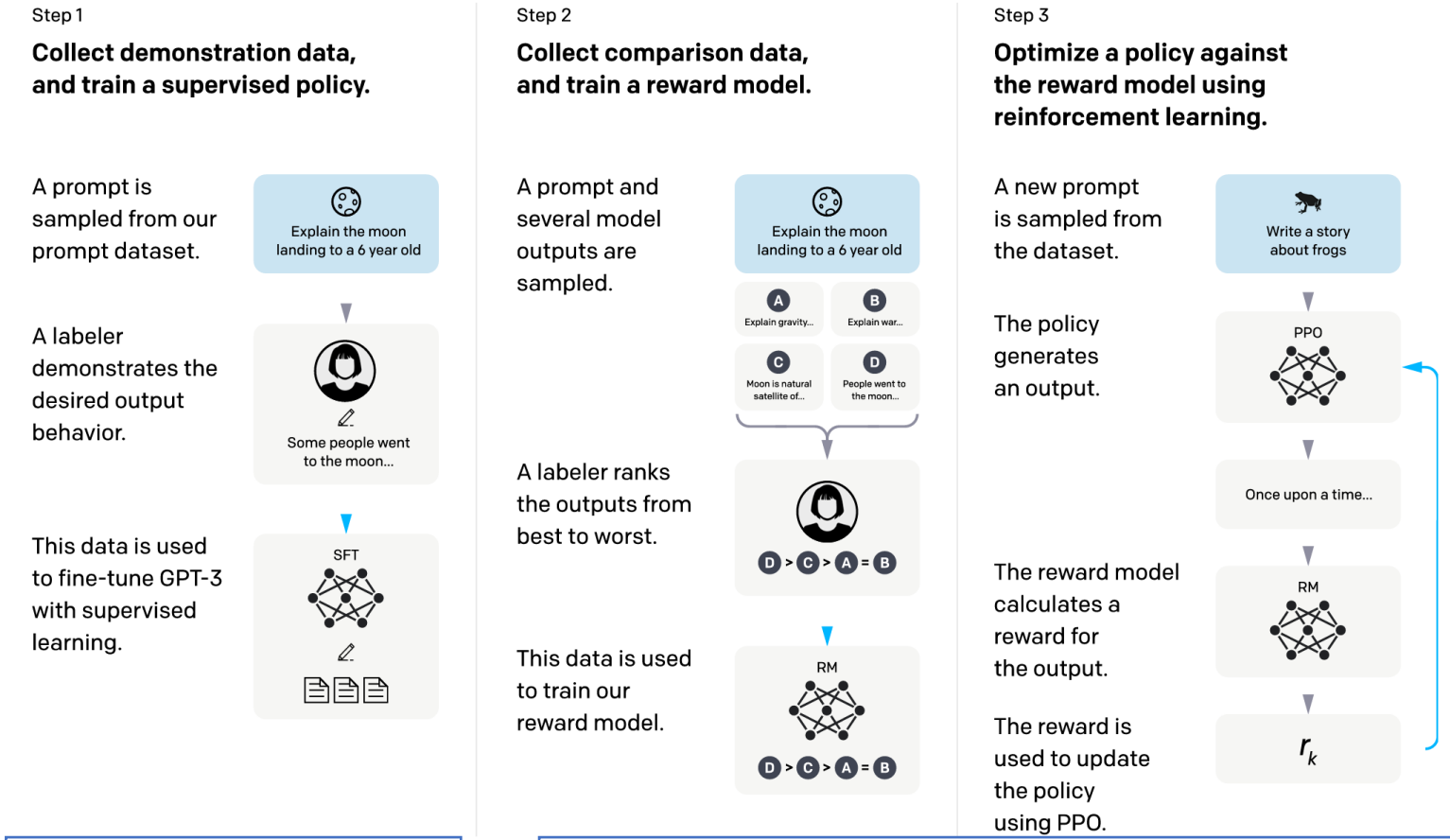
Large Language Model Pre-training Framework

- ChatGPT training procedure
  - Self-supervised pre-training
  - Supervised training on pairs of human-written data (Step 1)
  - Model generate multiple outputs for a prompt, train a reward model on human-labeled ranking list (Step 2)
  - Optimize the language model with the trained reward model (Step 3)



Training language models to follow instructions with human feedback. Ouyang et al. 2022.

Large Language Model Pre-training Framework



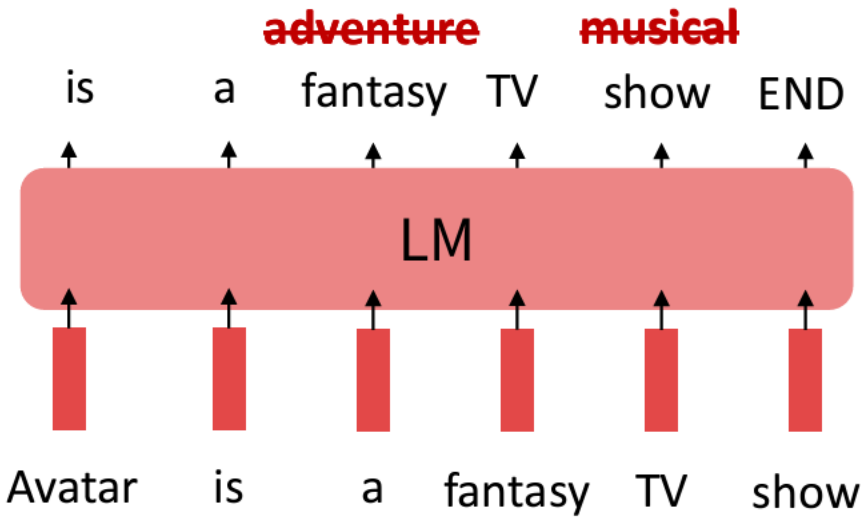
Instruction-Tuning (Supervised Fine-Tuning, SFT)

Reinforcement Learning from Human Feedback (RLHF) (covered in this course)

Training language models to follow instructions with human feedback. Ouyang et al. 2022.

Limitations of Instruction-Tuning

- Human-written pairs are very expensive
- Mismatch between LM objectives and human preferences
  - factual error vs. imprecise adjectives



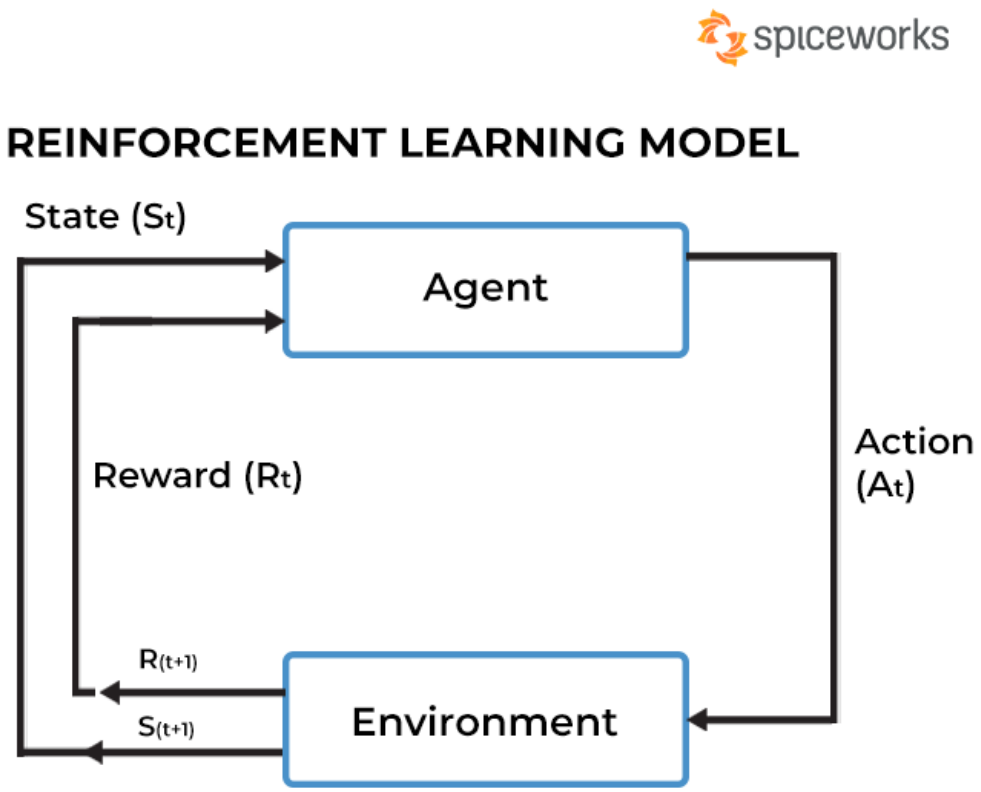
Common Objectives of Learning from Human Feedback

- Align model output with our values
- Trustworthy and robust on factualness
- Fairness on social values
- Explainable with logical rationales

Content

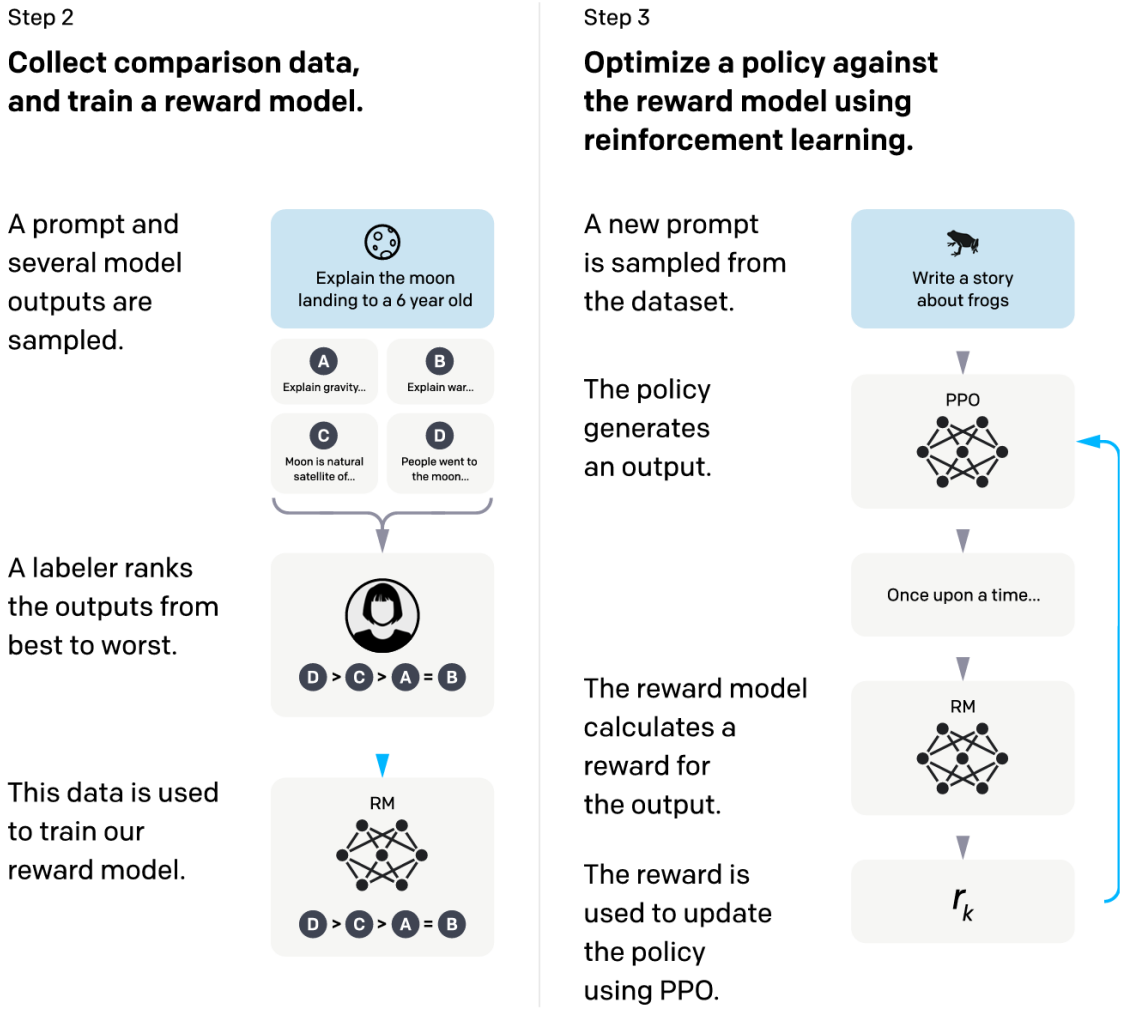
- **InstructGPT (Proximal Policy Optimization)**
- Direct Preference Optimization
- Fine-Grained Human Feedback
- Open problems for RLHF

Reinforcement Learning Model



- An agent has a policy function, which can take action A_t according to the current state S_t.
- As a result of the action, the agent receives a reward R_t from the environment and transit to the next state S_{t+1}.

InstructGPT: Training language models to follow instructions with human feedback. (Ouyang et. al, 2022)



- Agent: language model
- Action: predict the next token
- Policy π_θ : the output distribution of the next token

- Reward: a reward model r_ϕ trained by human evaluations on model responses, so no more human-in-the-loop is needed

Reward Model Training

- Prompt supervised fine-tuned language model with to produce pairs of answers

$$(y_1, y_2) \sim \pi^{\text{SFT}}(y \mid x)$$

- Human annotators decide which one wins / is preferred

$$y_w \succ y_l \mid x$$

- A reward model is trained to score y_w higher than y_l

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

- A reward model is often initialized from π^{SFT} with a linear layer to produce a scalar reward value

Fine-Tuning with RL: PPO[1]

- Optimize the language model π_θ with feedback from the reward model r_ϕ

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

prefer responses with high rewards

control the deviation from the reference policy, the π^{SFT} model

[1] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov. Proximal policy optimization algorithms, 2017.

Fine-Tuning with RL: PPO[1]

- Optimize the language model π_θ with feedback from the reward model r_ϕ

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

prefer responses with high rewards

control the deviation from the reference policy, the π^{SFT} model

- prevent mode-collapse to single high reward answers
- prevent the model deviating too far from the distribution where the reward model is accurate

[1] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov. Proximal policy optimization algorithms, 2017.

Fine-Tuning with RL: PPO[1]

- Optimize the language model π_θ with feedback from the reward model r_ϕ

$$\max_{\pi_\theta} \mathbb{E}_{x\sim\mathcal{D}} \underbrace{_{y\sim\pi_\theta(y|x)}}_{\text{sample y from the current policy}} \underbrace{\left[r_\phi(x,y)\right]}_{\text{prefer responses with high rewards}} - \underbrace{\beta\mathbb{D}_{\mathrm{KL}}\left[\pi_\theta(y \mid x) \,||\, \pi_{\mathrm{ref}}(y \mid x)\right]}_{\text{control the deviation from the reference policy, the } \pi^{\mathrm{SFT}} \text{ model}}$$

- prevent mode-collapse to single high reward answers
- prevent the model deviating too far from the distribution where the reward model is accurate

[1] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov. Proximal policy optimization algorithms, 2017.

Fine-Tuning with RL: PPO-ptx[1]

- Training objective

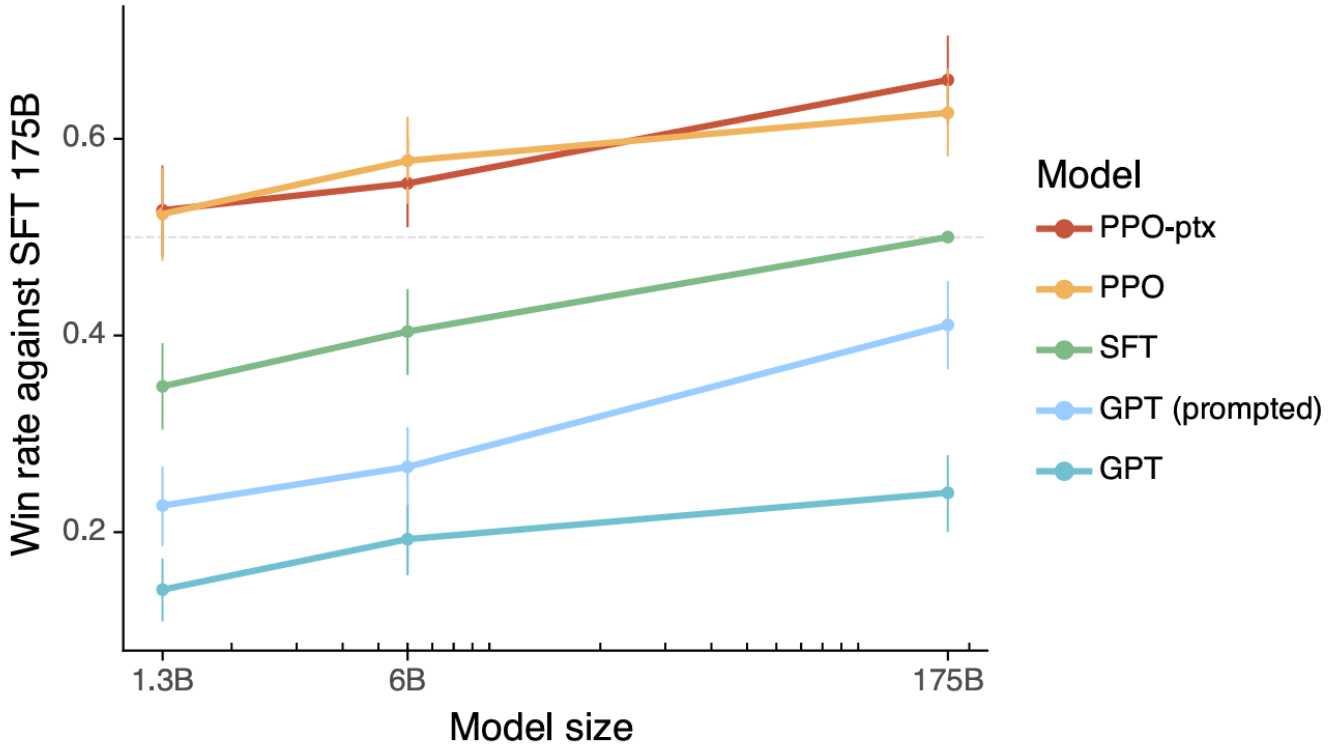
$$E_{(x,y)\sim D_{\pi_\phi^{\mathrm{RL}}}}\left[r_\theta(x,y) - \beta \log\left(\pi_\theta\ (y \mid x)/\pi_{\mathrm{ref}}(y \mid x)\right)\right] +$$
$$\gamma E_{x\sim D_{\mathrm{pretrain}}}\left[\log(\pi_\theta\ (x))\right]$$

- Add pre-training gradients to fix the performance regressions on public NLP tasks
- For PPO models, γ is set to 0

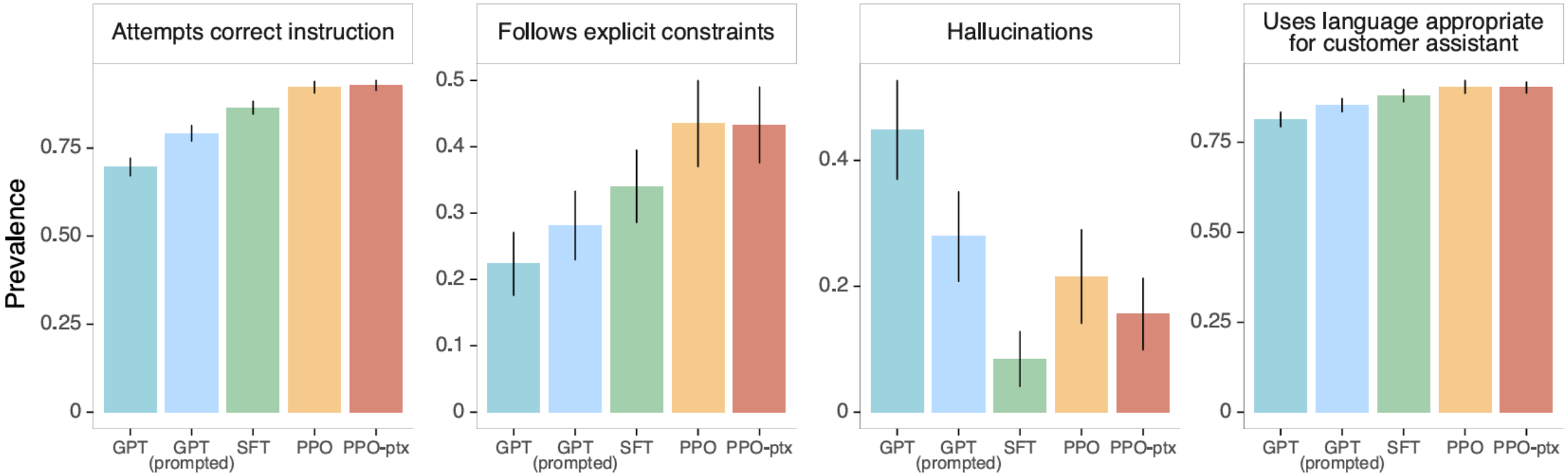
[1] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov. Proximal policy optimization algorithms, 2017.

Comparison with Baselines

- RLHF models are more preferred by human labelers



Evaluations on Different Aspects



Content

- InstructGPT (Proximal Policy Optimization)
- **Direct Preference Optimization**
- Fine-Grained Human Feedback
- Open problems for RLHF

Limitation of PPO methods

- Need to train multiple models: a reward model and a policy model
- Need sampling from LM during fine-tuning
- The RL training process is too complicated!
- Is it possible to directly train a language model from the human preference annotations?

DPO: Direct Preference Optimization: Your Language Model is Secretly a Reward Model (Rafailov et. al, 2023)

- Looking into the PPO objective

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

- Deriving optimal closed-form solution

$$\begin{aligned}
&\max_{\pi} \mathbb{E}_{x\sim\mathcal{D},y\sim\pi}\left[r(x,y)\right] - \beta\mathbb{D}_{\text{KL}}\left[\pi(y|x) \,||\, \pi_{\text{ref}}(y|x)\right] \\
&\quad= \max_{\pi} \mathbb{E}_{x\sim\mathcal{D}}\mathbb{E}_{y\sim\pi(y|x)}\left[r(x,y) - \beta\log\frac{\pi(y|x)}{\pi_{\text{ref}}(y|x)}\right] \\
&\quad= \min_{\pi} \mathbb{E}_{x\sim\mathcal{D}}\mathbb{E}_{y\sim\pi(y|x)}\left[\log\frac{\pi(y|x)}{\pi_{\text{ref}}(y|x)} - \frac{1}{\beta}r(x,y)\right] \\
&\quad= \min_{\pi} \mathbb{E}_{x\sim\mathcal{D}}\mathbb{E}_{y\sim\pi(y|x)}\left[\log\frac{\pi(y|x)}{\frac{1}{Z(x)}\pi_{\text{ref}}(y|x)\exp\left(\frac{1}{\beta}r(x,y)\right)} - \log Z(x)\right]
\end{aligned}$$

partition function:

$$Z(x) = \sum_y \pi_{\text{ref}}(y|x)\exp\left(\frac{1}{\beta}r(x,y)\right)$$

Direct Preference Optimization

- PPO Objective

$$\min_{\pi} \mathbb{E}_{x\sim\mathcal{D}}\mathbb{E}_{y\sim\pi(y|x)}\left[\log \frac{\pi(y|x)}{\frac{1}{Z(x)}\pi_{\text{ref}}(y|x)\exp\left(\frac{1}{\beta}r(x,y)\right)} - \log Z(x)\right]$$

partition function:

$$Z(x) = \sum_{y} \pi_{\text{ref}}(y|x)\exp\left(\frac{1}{\beta}r(x,y)\right)$$

- Partition function is a function of only x and π_{ref} , but does not depend on the policy π
- Therefore we can define $\pi^*(y|x) = \frac{1}{Z(x)}\pi_{\text{ref}}(y|x)\exp\left(\frac{1}{\beta}r(x,y)\right)$
- $\pi^*(y|x)$ is a valid probability, therefore the objective can be seen as a KL divergence between two probability distribution
- The optimal solution of the objective

$$\pi(y|x) = \pi^*(y|x) = \frac{1}{Z(x)}\pi_{\text{ref}}(y|x)\exp\left(\frac{1}{\beta}r(x,y)\right)$$

Direct Preference Optimization

- Every reward function induce an optimal policy

$$\pi_r(y \mid x) = \frac{1}{Z(x)} \pi_{\text{ref}}(y \mid x) \exp\left(\frac{1}{\beta} r(x, y)\right)$$

- Every policy is the optimal policy of some reward function

$$r(x, y) = \beta \log \frac{\pi_r(y \mid x)}{\pi_{\text{ref}}(y \mid x)} + \beta \log Z(x)$$

This term is intractable!

- Key idea: train the policy model so that r(x,y) fits the human preference data!

Direct Preference Optimization

- Recall the reward model training loss

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

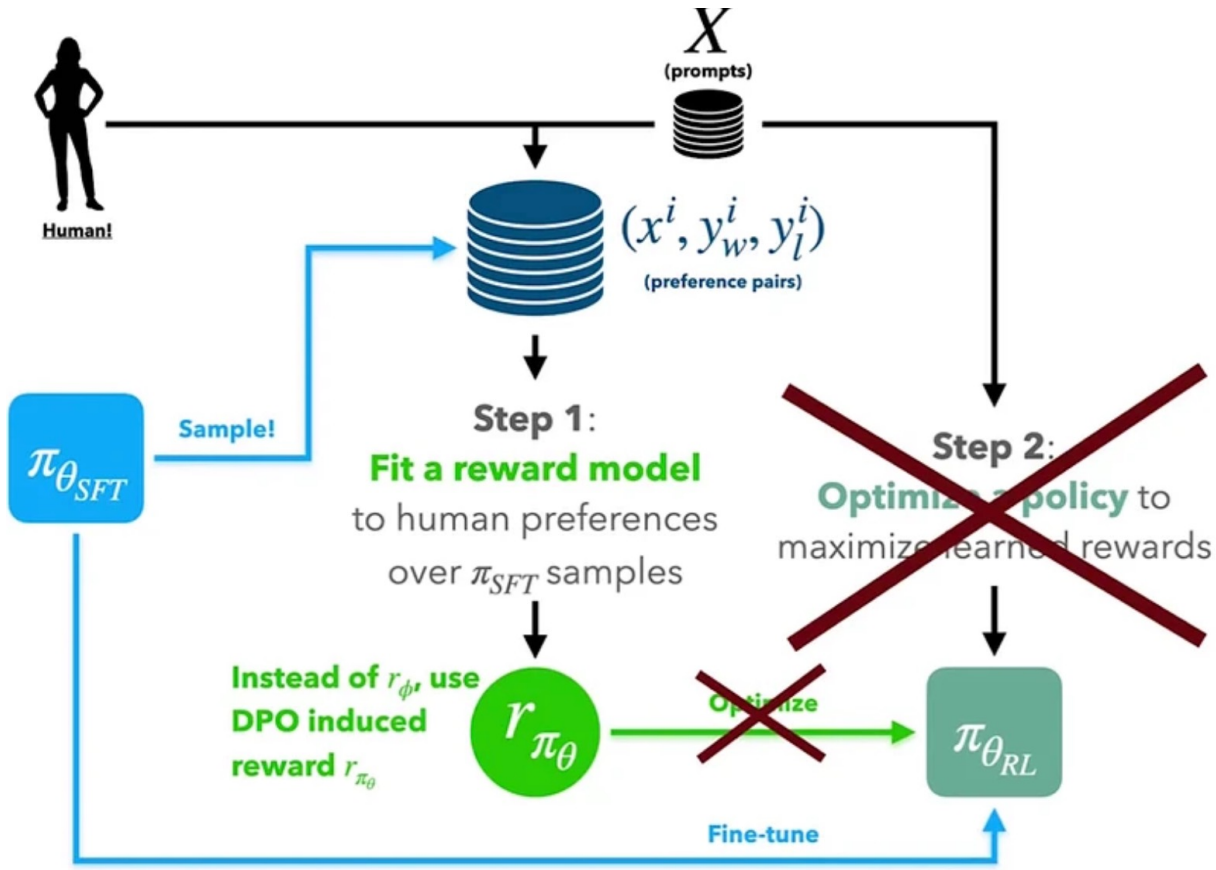
- The partition function cancels out when we take the difference between the reward of a pair of responses!
- DPO training objective:

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

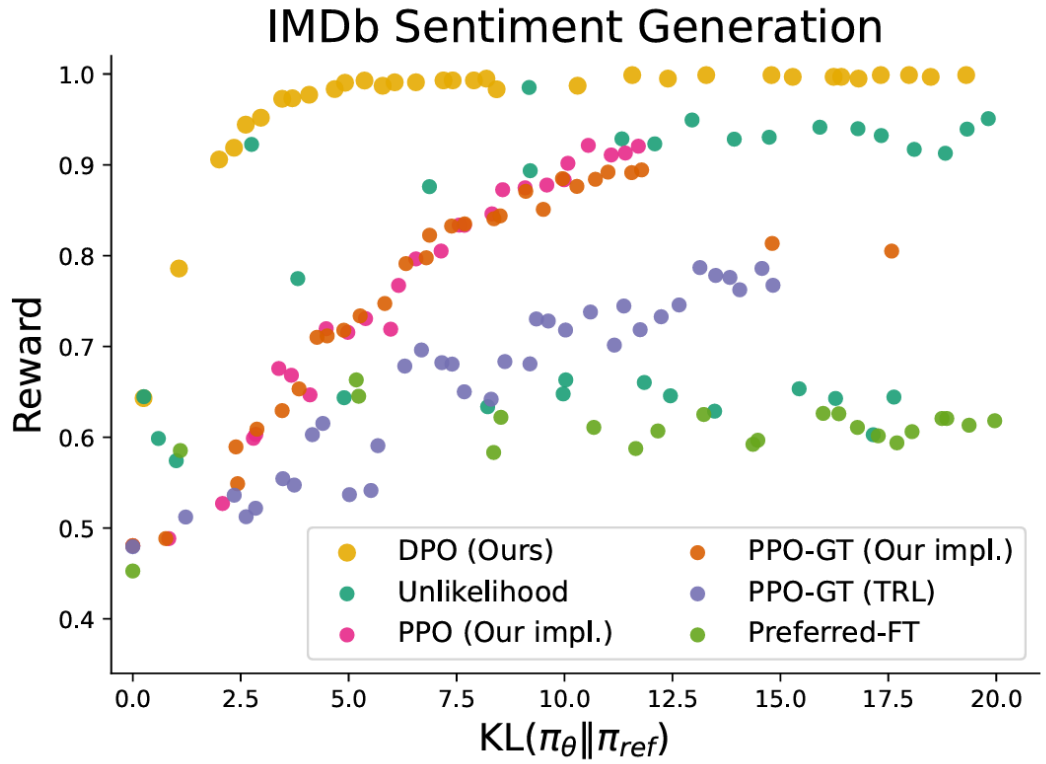
- A simple classification loss!

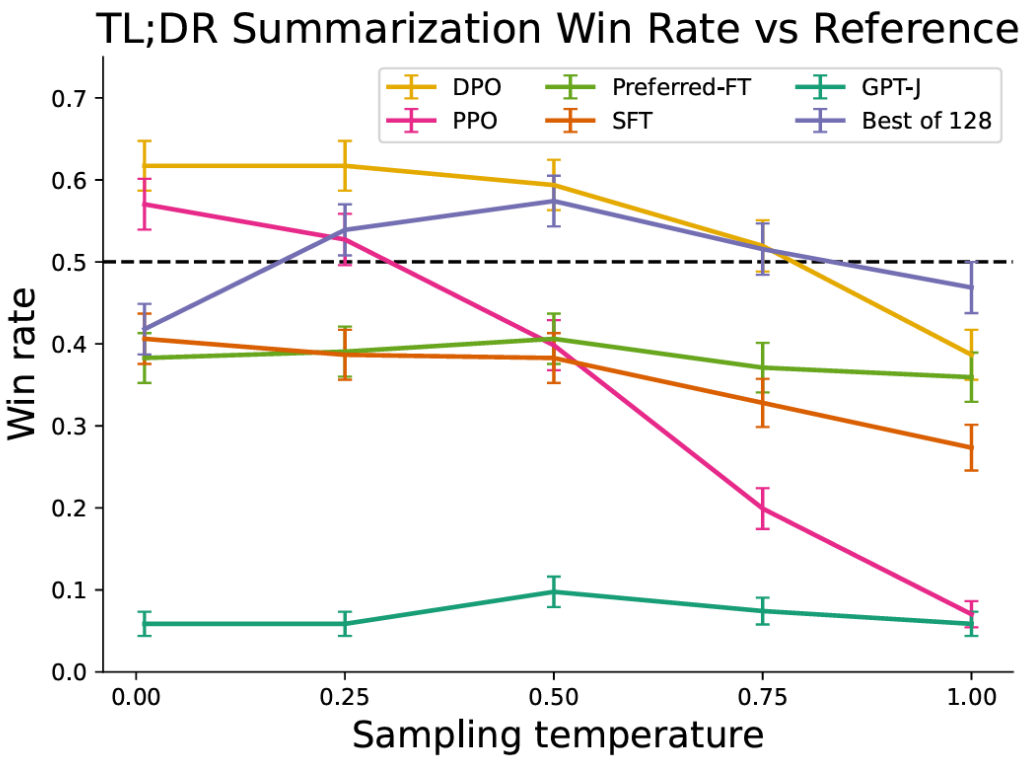
What does DPO do?

- DPO eliminates the need to train a reward model, sample from the LLM during fine-tuning, or perform significant hyperparameter search.



Comparison with Baseline Models





- Preferred-FT: Fine-tune the model on y_w
- PPO-GT: reward model is the ground truth of the sentiment
- Unlikelihood: optimize the policy model to maximize $P(y_w)$ and minimize $P(y_l)$
- Best of N: sampling N responses from the SFT model (very inefficient)

Comparison between PPO and DPO

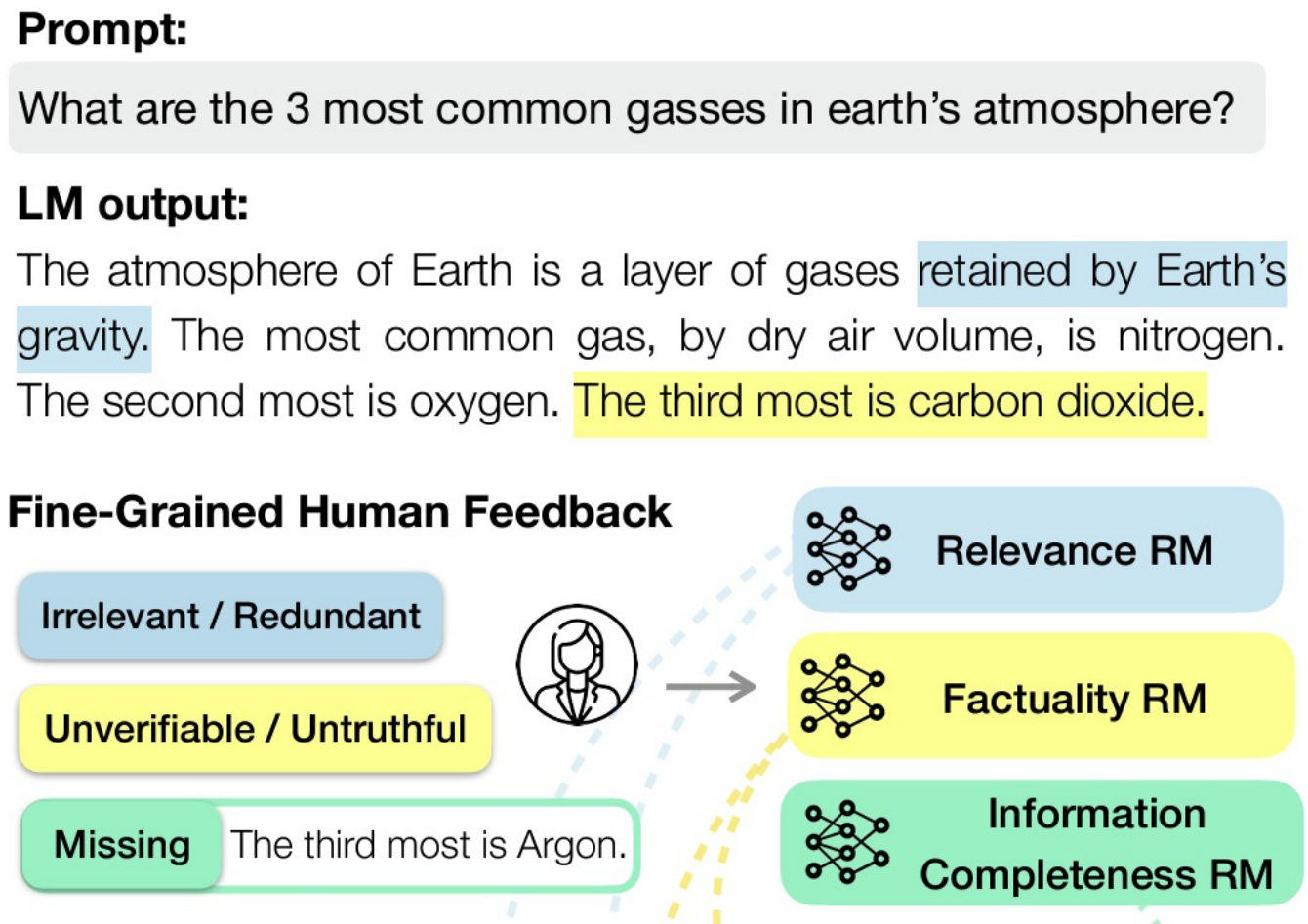
- DPO training is cheaper and more stable than PPO training
- PPO can handle more informative human feedback (e.g., numerical ratings) while DPO can only handle binary signals

Content

- InstructGPT (Proximal Policy Optimization)
- Direct Preference Optimization
- **Fine-Grained Human Feedback**
- Open problems for RLHF

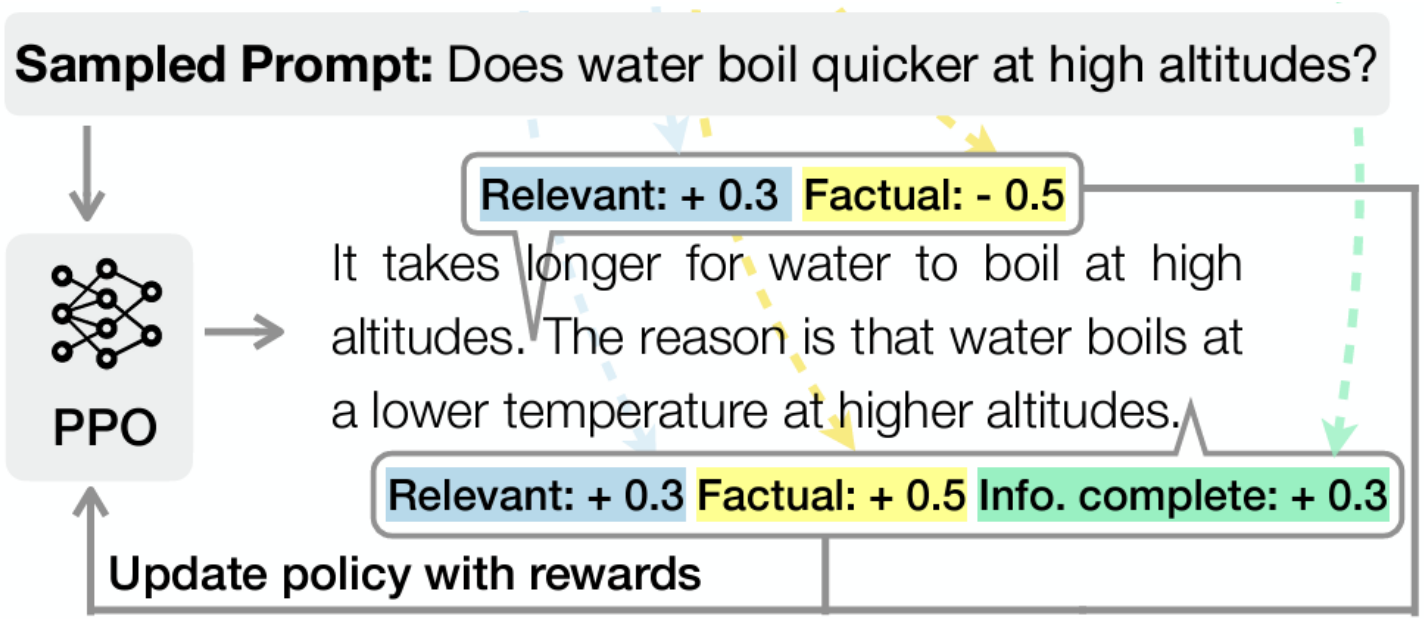
Fine-Grained Human Feedback Gives Better Rewards for Language Model Training (Wu et. al, 2023)

- Assigning a single score to the model output may not be informative enough



Multiple Fine-Grained Reward Functions

- (1) Provide a reward after every segment (e.g., a sentence) is generated
- (2) Different feedback types: factual incorrectness, irrelevance, and information incompleteness



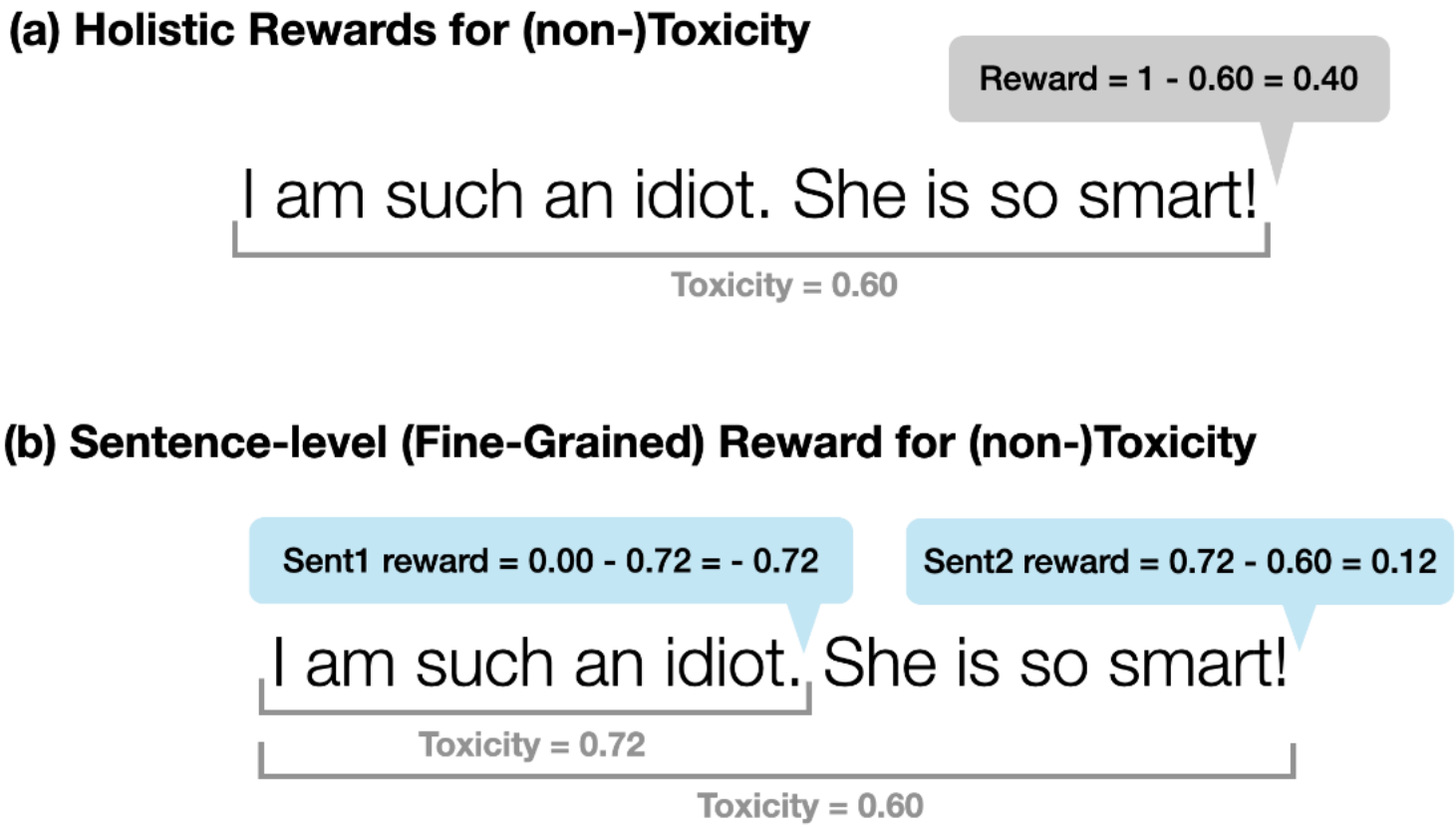
Combined Reward Function

$$r_t = \sum_{k=1}^{K} \sum_{j=1}^{L_k} \left(\mathbb{1}(t = T_j^k)\, w_k\, R_{\phi_k}(x, y, j) \right) - \beta \log \frac{P_\theta(a_t \mid s_t)}{P_{\theta_{\text{init}}}(a_t \mid s_t)}$$

- w_k is a weight assigned to each reward function

Use Case I: Detoxification

- Perspective API: measures toxicity (0: non-toxic, 1: toxic)



Use Case I: Detoxification

- Learning from **denser** fine-grained reward is more sample efficient than holistic reward.
- Fine-grained reward locates where the toxic content is, which is a stronger training signal compared with a scalar reward for the whole text.

	Toxicity	Fluency	Diversity	
	avg max (↓)	PPL (↓)	dist-2 (↑)	dist-3 (↑)
GPT-2	0.192	9.58	0.947	0.931
Controlled Generation				
GeDi	0.154	24.78	0.938	**0.938**
DEXPERTS	0.136	22.83	0.932	0.922
Hol. RLHF	0.130	11.75	0.943	0.926
F.G. RLHF	**0.081**	**9.77**	**0.949**	0.932

Table 1: Results on the REALTOXICITYPROMPTS test set.

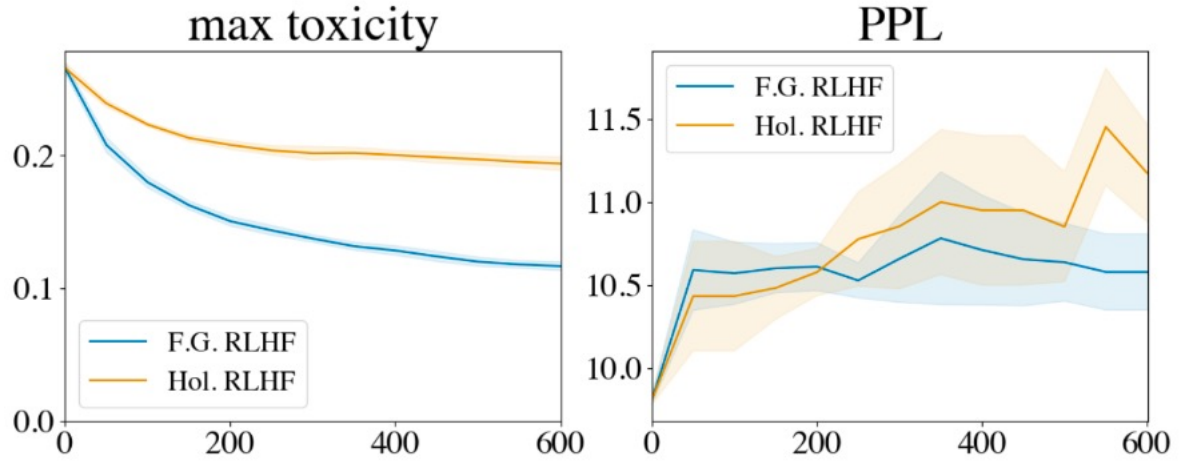
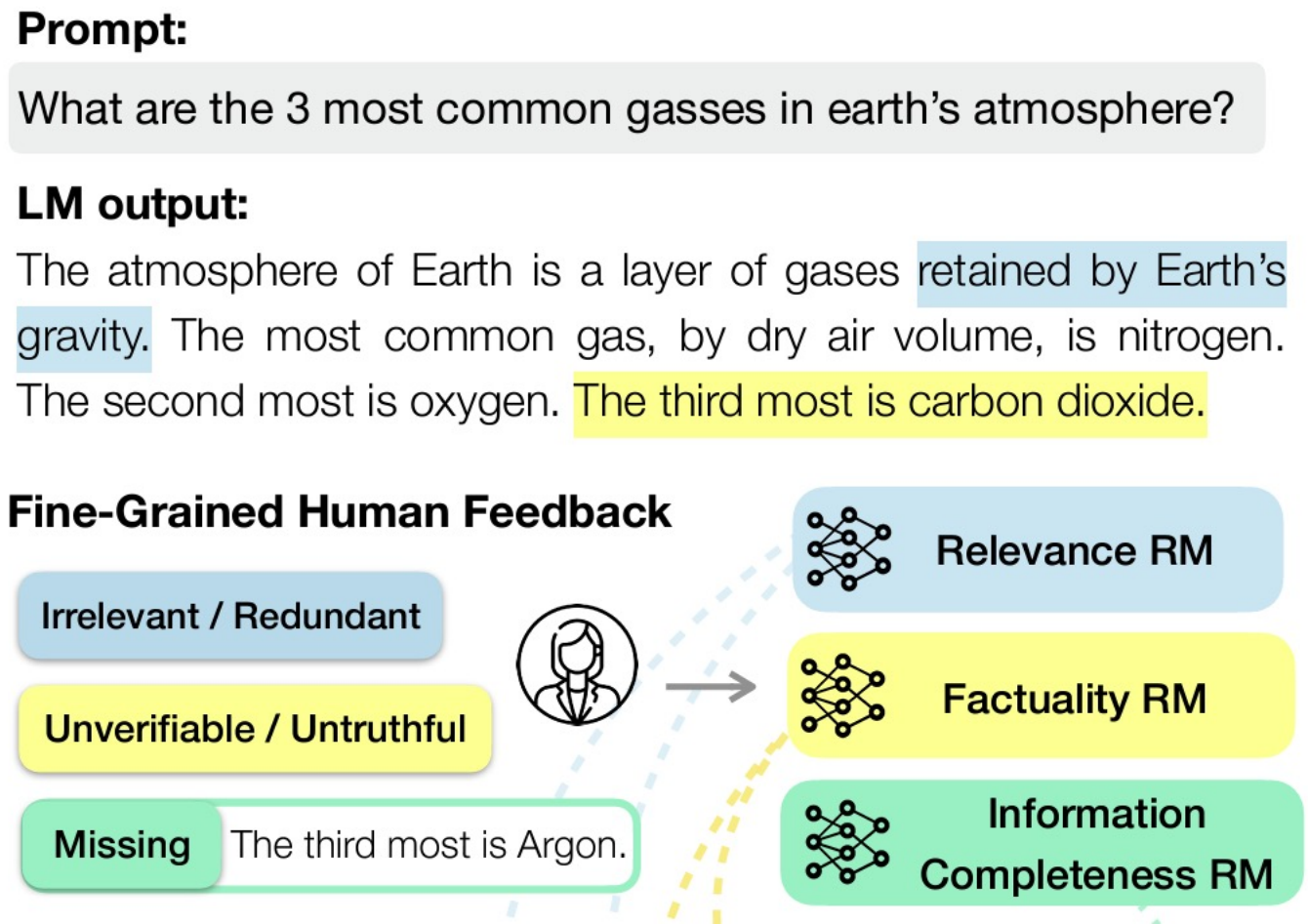


Figure 2: Curves of toxicity and perplexity on the dev set v.s. training steps.

Use Case II: Long-Form Question Answering

- Train a fine-grained reward model for each of the three aspects.



Use Case II: Long-Form Question Answering

- Fine-Grained RLHF outperforms SFT and Preference RLHF on all error types.
- RLHF (both preference-based and fine-grained) are particularly effective in reducing factual errors.

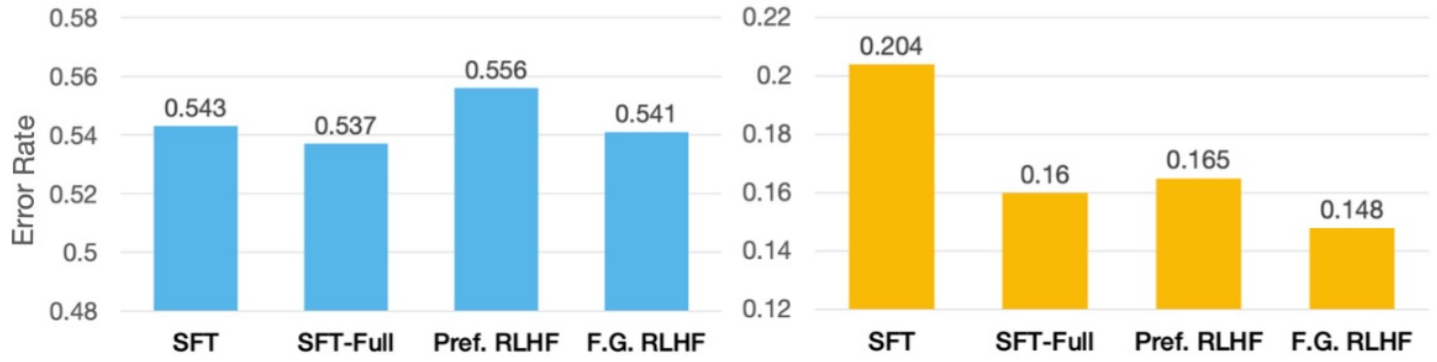


Figure 3: Human evaluation on *rel.* (left) and *fact.* (right) error, measured by % of sub-sentences that contain the error type (↓).

Ours v.s.	Win	Tie	Lose
SFT	**23.0%**	65.5%	11.5%
SFT-Full	**22.0%**	61.0%	17.0%
Pref. RLHF	**19.5%**	71.0%	9.5%

Table 2: Human pairwise comparison on information completeness (*comp.*), where win/lose refers to FINE-GRAINED RLHF.

Fine-Grained RLHF for Customizing LLM Behavior

- Relevance reward weight: 0.4/0.3/0.2
- Keep factualness/completeness reward weight fixed
- Relevance reward penalize referencing passages and auxiliary information

Question: | When did the French join revolution on colonists' side?

LM outputs with different reward model configurations:

Short: French involvement in the American Revolutionary War began in 1775, when France, a rival of the British Empire, secretly shipped supplies to the Continental Army. A Treaty of Alliance in 1778 soon followed, which led to shipments of money and matériel to the United States. Subsequently, the Spanish Empire and the Dutch Republic also began to send assistance, leaving the British Empire with no allies.

Medium: French involvement in the American Revolutionary War began in 1775, when France, a rival of the British Empire, secretly shipped supplies to the Continental Army. A Treaty of Alliance in 1778 soon followed, which led to shipments of money and matériel to the United States. Subsequently, the Spanish Empire and the Dutch Republic also began to send assistance, leaving the British Empire with no allies. On March 17, 1778, four days after a French ambassador informed the British government that France had officially recognized the United States as an independent nation, Great Britain declared war on France, thereby engaging that nation in the American Revolutionary War.

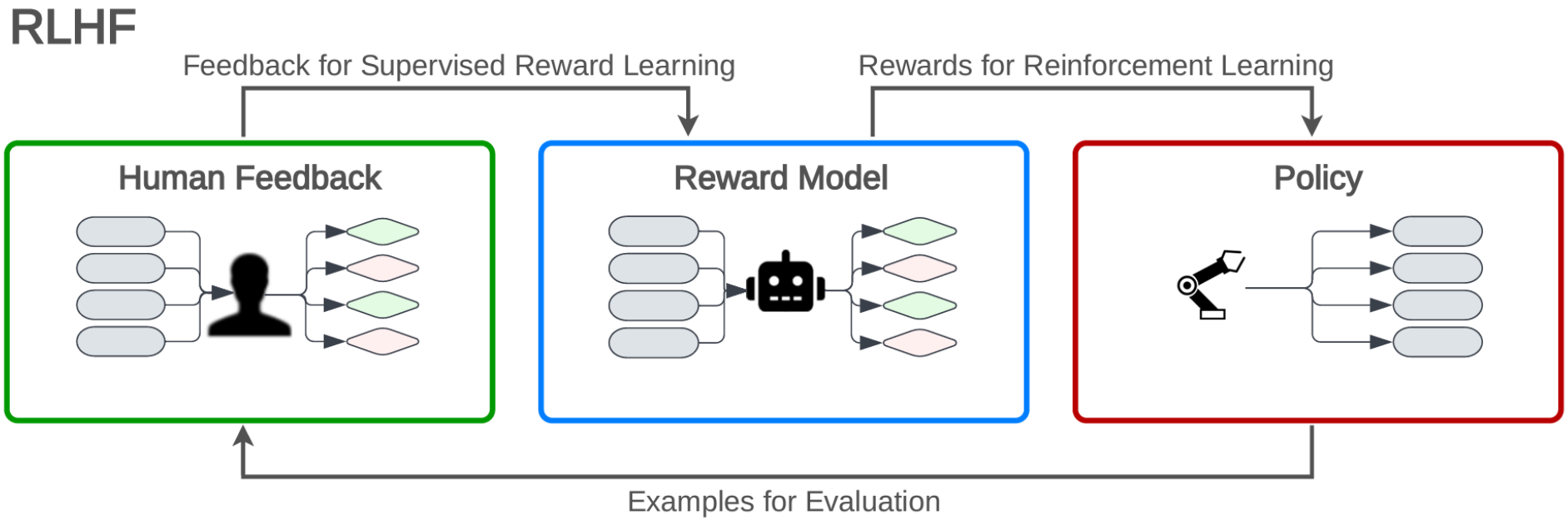
Long: French involvement in the American Revolutionary War began in 1775, when France, a rival of the British Empire, secretly shipped supplies to the Continental Army. A Treaty of Alliance in 1778 soon followed, which led to shipments of money and matériel to the United States. Subsequently, the Spanish Empire and the Dutch Republic also began to send assistance, leaving the British Empire with no allies. On March 17, 1778, four days after a French ambassador informed the British government that France had officially recognized the United States as an independent nation with the signing of the Treaty of Amity and Commerce and the Treaty of Alliance, Great Britain declared war on France, thereby engaging that nation in the American Revolutionary War.

Content

- InstructGPT (Proximal Policy Optimization)
- Direct Preference Optimization
- Fine-Grained Human Feedback
- **Open problems for RLHF**

Open Problems and Fundamental Limitations of Reinforcement Learning from Human Feedback (Casper et. al, 2023)

- Challenges within each step: human feedback, reward model and policy



Challenges with Obtaining Human Feedback

- Human evaluators may have biases
  - Studies found that ChatGPT models became politically biased post RLHF.
- Good oversight is difficult
  - Evaluators are paid per example and may make mistakes given time constraints; poor feedback on evaluating difficult tasks
- Data quality
  - cost / quality tradeoff
- Tradeoff between richness and efficiency of feedback types
  - comparison-based feedback, scalar feedback, correction feedback, language feedback, …

Challenges with the Reward Model

- A single reward function cannot represent a diverse society of humans
- Reward misgeneralization: reward models may fit with human preference data with unexpected features
- Evaluation of a reward model is difficult and expensive

Challenges with the Policy

- Robust reinforcement learning is difficult
  - balance between exploring new actions and exploiting known rewards
  - the challenge intensifies in high-dimensional or sparse reward settings
- Policy misgeneralization: training and deployment environment is difference

Next Course: Parameter-Efficient Fine-tuning of LLMs