RL from Human Feedback

Sham Kakade and Kianté Brantley

CS 2824: Foundations of Reinforcement Learning

Recap

We have covered a few RL algorithms REINFORCE, PPO;

Recap

We have covered a few RL algorithms REINFORCE, PPO;

They all rely on a key and strong assumption: reward function/signal is given

Motivation

Modern chatbots are pre-trained via next-token prediction on web data, followed by fine-tuning using human feedback (post-training)

Outline

1. LLM as a policy

2. Learning reward functions from preference data

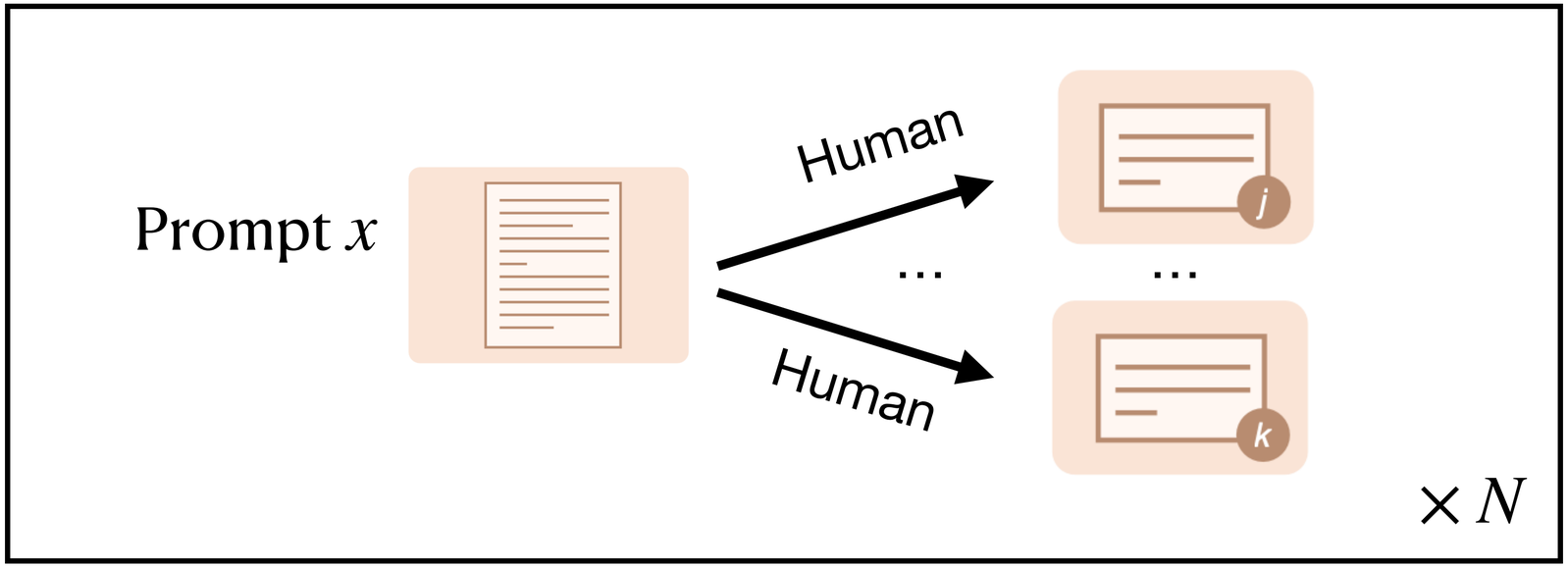
3. KL-regularized RL

4. DPO

5. REBEL

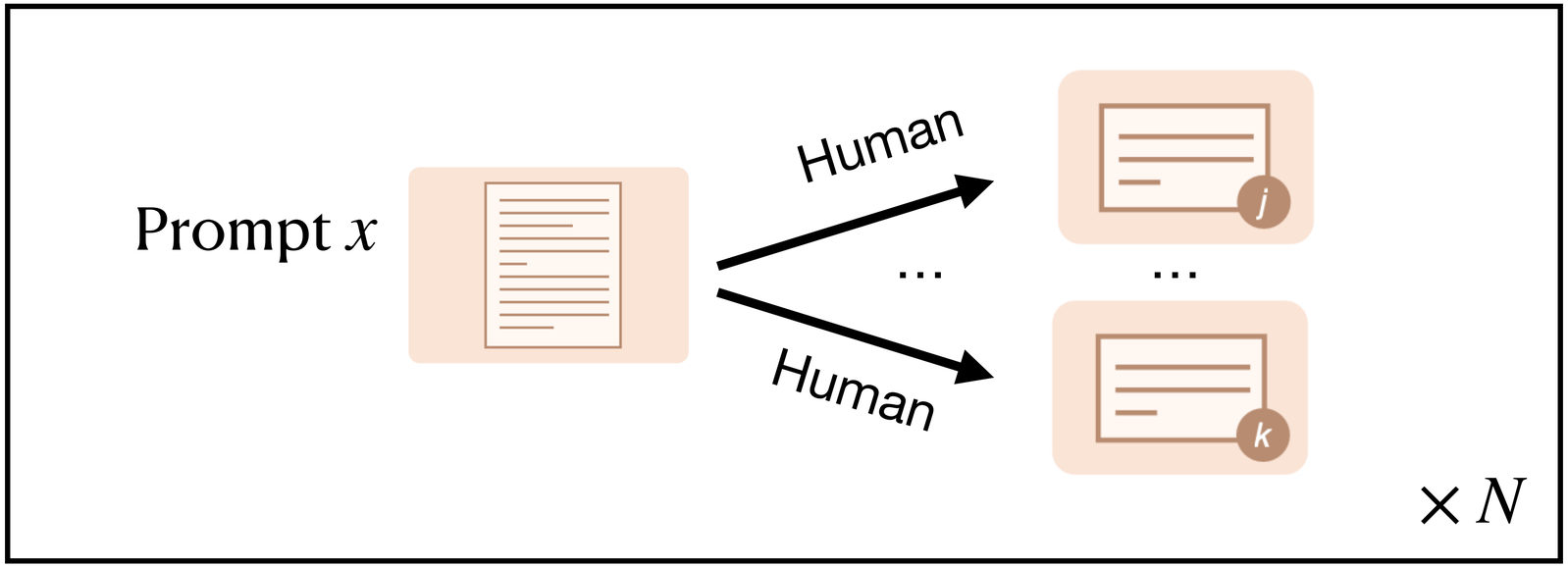
The post-training Pipeline: Supervised Fine-tuning (SFT)

Collect instruction-response data



The post-training Pipeline: Supervised Fine-tuning (SFT)

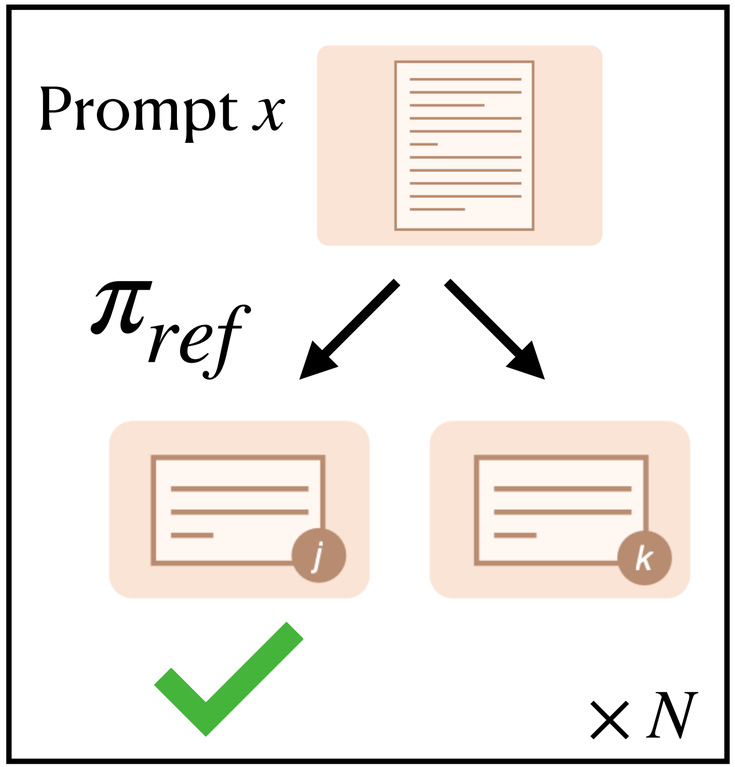
Collect instruction-response data



SFT: given prompts, train LLM to predict tokens in human responses

The post-training Pipeline: RLHF

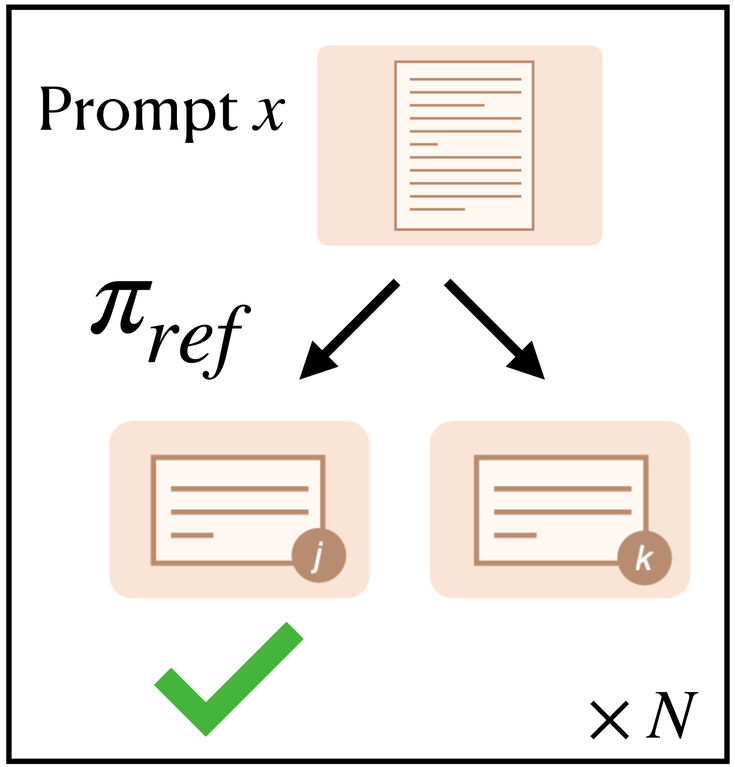
1. Collect preference dataset



[ChatGPT blog post: https://openai.com/index/chatgpt/]

The post-training Pipeline: RLHF

1. Collect preference dataset

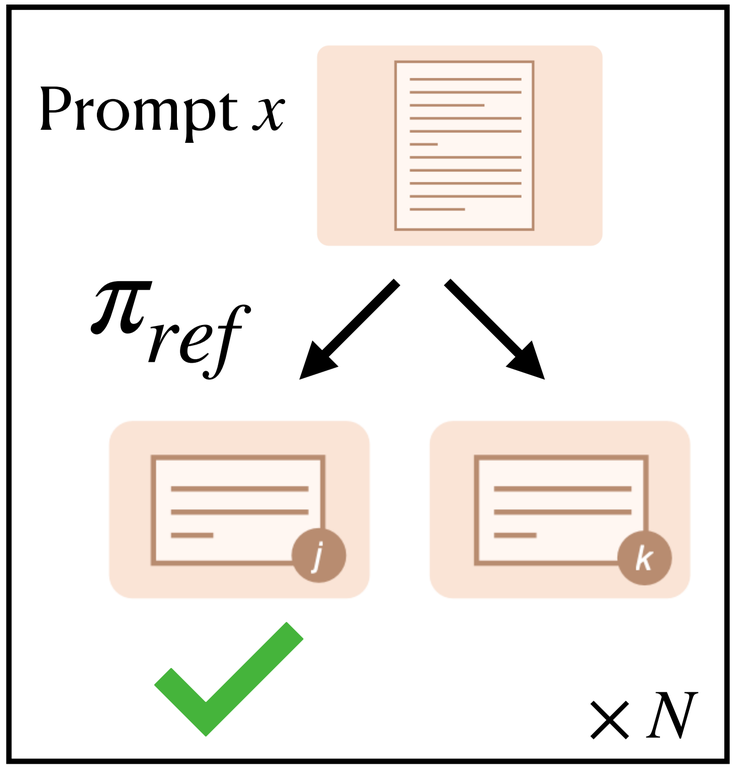


$$\mathscr{D}_{off} = \{x, \tau, \tau', z\}$$

[ChatGPT blog post: https://openai.com/index/chatgpt/]

The post-training Pipeline: RLHF

1. Collect preference dataset



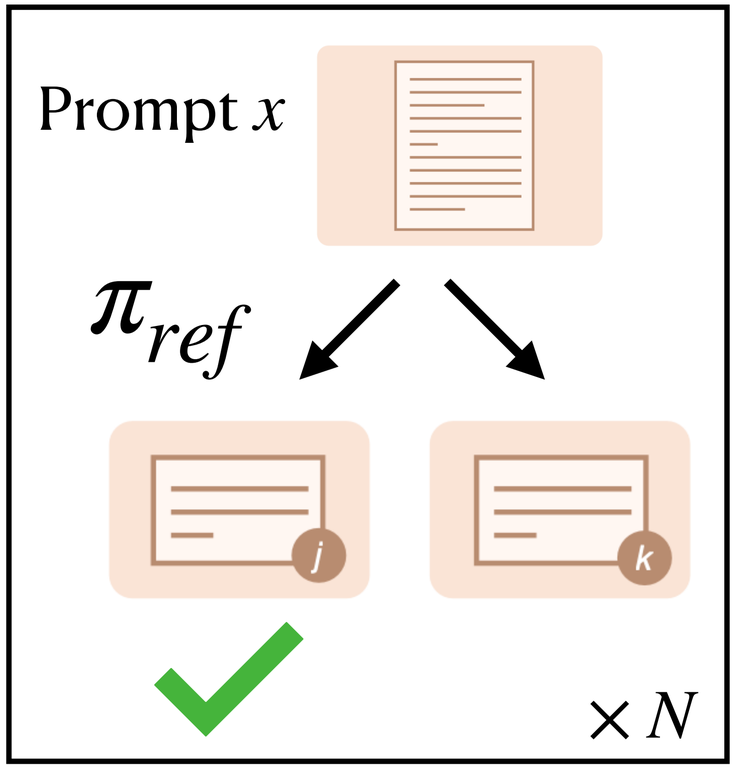
$\mathcal{D}_{off} = \{x, \tau, \tau', z\}$

2. Learn a reward model $\hat{r}$ using the data from step 1

[ChatGPT blog post: https://openai.com/index/chatgpt/]

The post-training Pipeline: RLHF

1. Collect preference dataset



$\mathcal{D}_{off} = \{x, \tau, \tau', z\}$

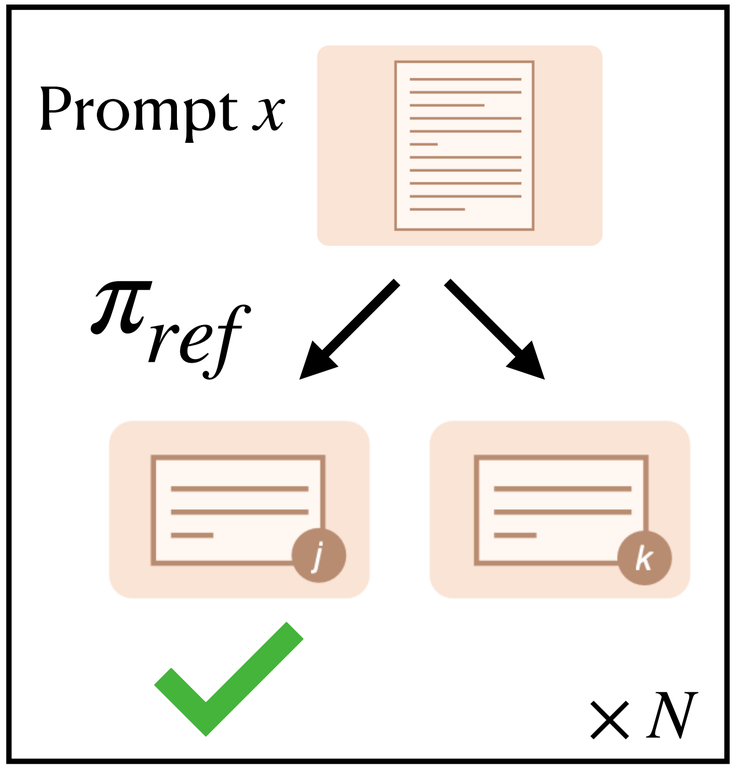
2. Learn a reward model $\hat{r}$ using the data from step 1

3. train policy via RL (e.g., PPO)

[ChatGPT blog post: https://openai.com/index/chatgpt/]

The post-training Pipeline: RLHF

1. Collect preference dataset



$\mathcal{D}_{off} = \{x, \tau, \tau', z\}$

2. Learn a reward model $\hat{r}$ using the data from step 1

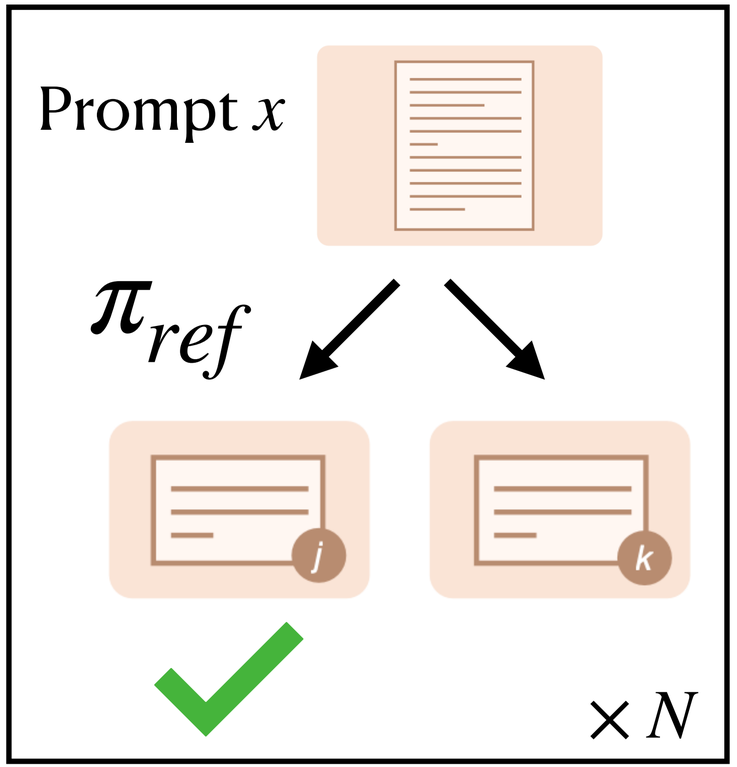
3. train policy via RL (e.g., PPO)

Prompt x

[ChatGPT blog post: https://openai.com/index/chatgpt/]

The post-training Pipeline: RLHF

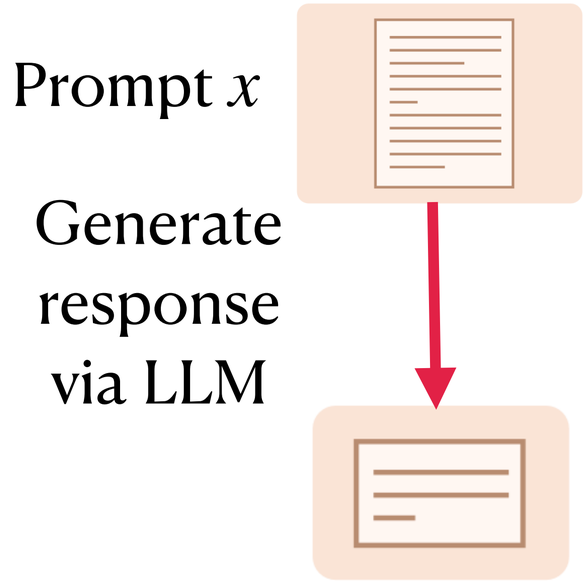
1. Collect preference dataset



$$\mathcal{D}_{off} = \{x, \tau, \tau', z\}$$

2. Learn a reward model $\hat{r}$ using the data from step 1

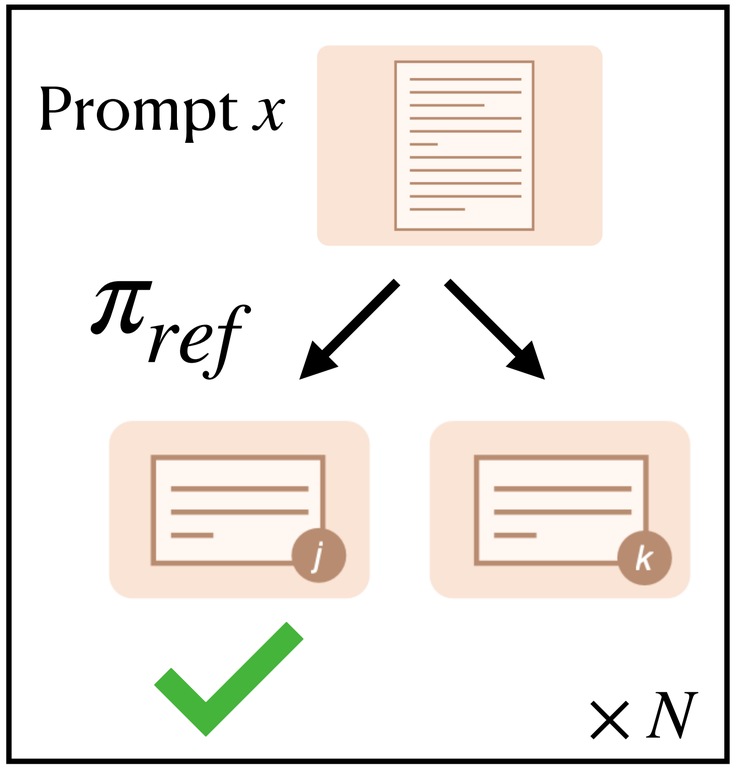
3. train policy via RL (e.g., PPO)



[ChatGPT blog post: https://openai.com/index/chatgpt/]

The post-training Pipeline: RLHF

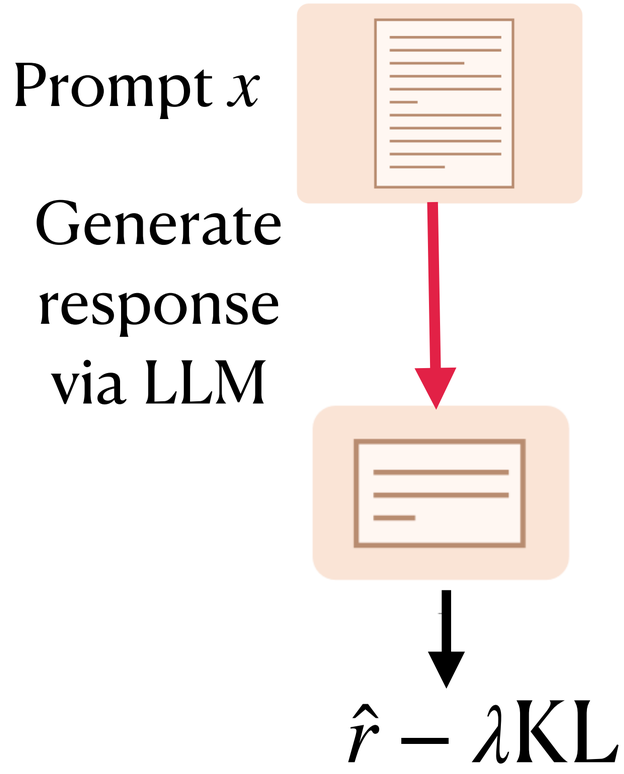
1. Collect preference dataset



$\mathcal{D}_{off} = \{x, \tau, \tau', z\}$

2. Learn a reward model $\hat{r}$ using the data from step 1

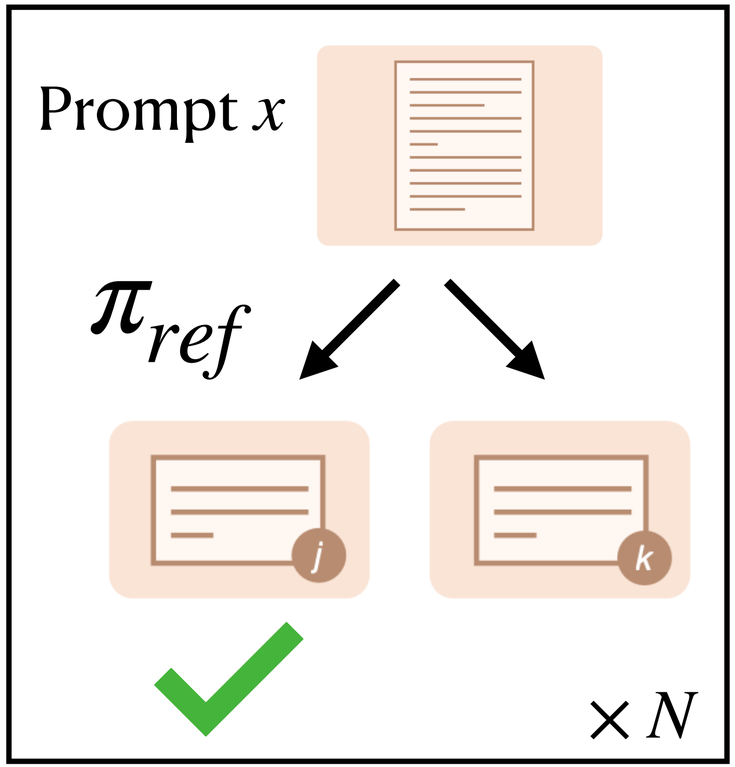
3. train policy via RL (e.g., PPO)



[ChatGPT blog post: https://openai.com/index/chatgpt/]

The post-training Pipeline: RLHF

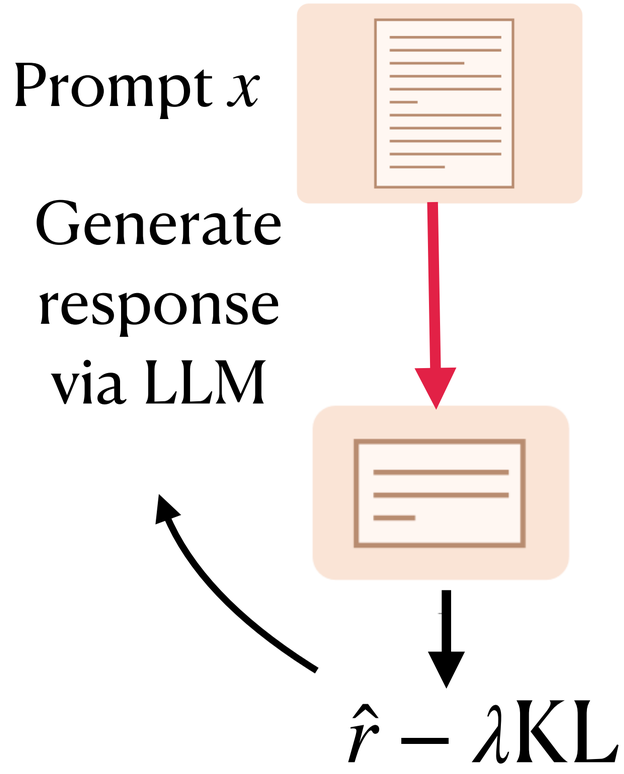
1. Collect preference dataset



$\mathcal{D}_{off} = \{x, \tau, \tau', z\}$

2. Learn a reward model $\hat{r}$ using the data from step 1

3. train policy via RL (e.g., PPO)



[ChatGPT blog post: https://openai.com/index/chatgpt/]

What’s the benefit of RLHF over SFT?

Evaluation is often easier than generation

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

What’s the benefit of RLHF over SFT?

Evaluation is often easier than generation

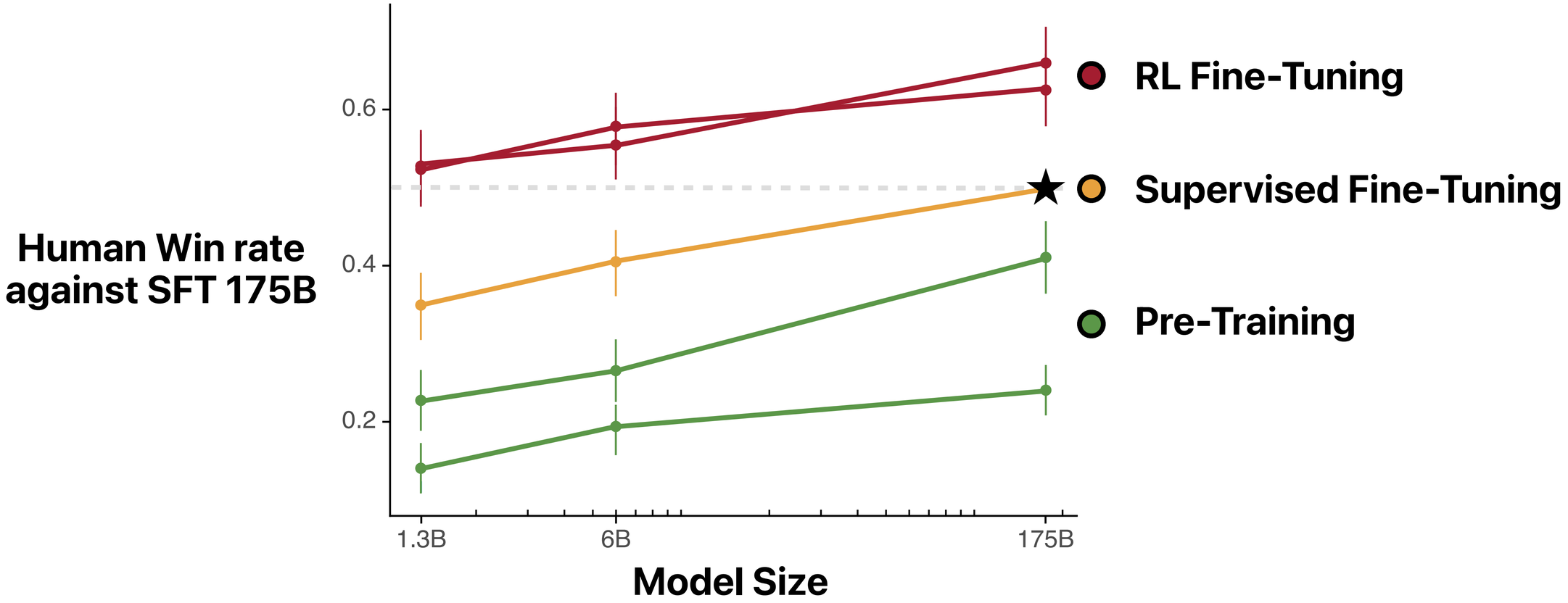
Given a high quality reward, RLHF can often make model outperform humans:

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

What's the benefit of RLHF over SFT?

Evaluation is often easier than generation

Given a high quality reward, RLHF can often make model outperform humans:



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

The MDP formulation of text generation

Initial state s_0: prompt x

Action: token y; action space: all possible tokens

State: prompt + generated tokens, e.g., $s_h = (x, y_0, y_1, \ldots, y_{h-1})$

Transition: concatenation, i.e., given s_h and y_h, $s_{h+1} = (s_h, y_h)$

Terminate: either hits the maximum content length or hits the special EOS token

Outline

1. LLM as a policy

2. Learning reward functions from preference data

3. KL-regularized RL

4. DPO

5. REBEL

Learning reward from human data

Reward design can be challenging in RL

Bradley-Terry Model

Assume there is a ground truth reward $r^\star(x, \tau)$ (i.e., high reward means response is good)

Bradley-Terry Model

Assume there is a ground truth reward $r^\star(x, \tau)$ (i.e., high reward means response is good)

The BT model assumes that humans generate labels based on the following probablistic model:

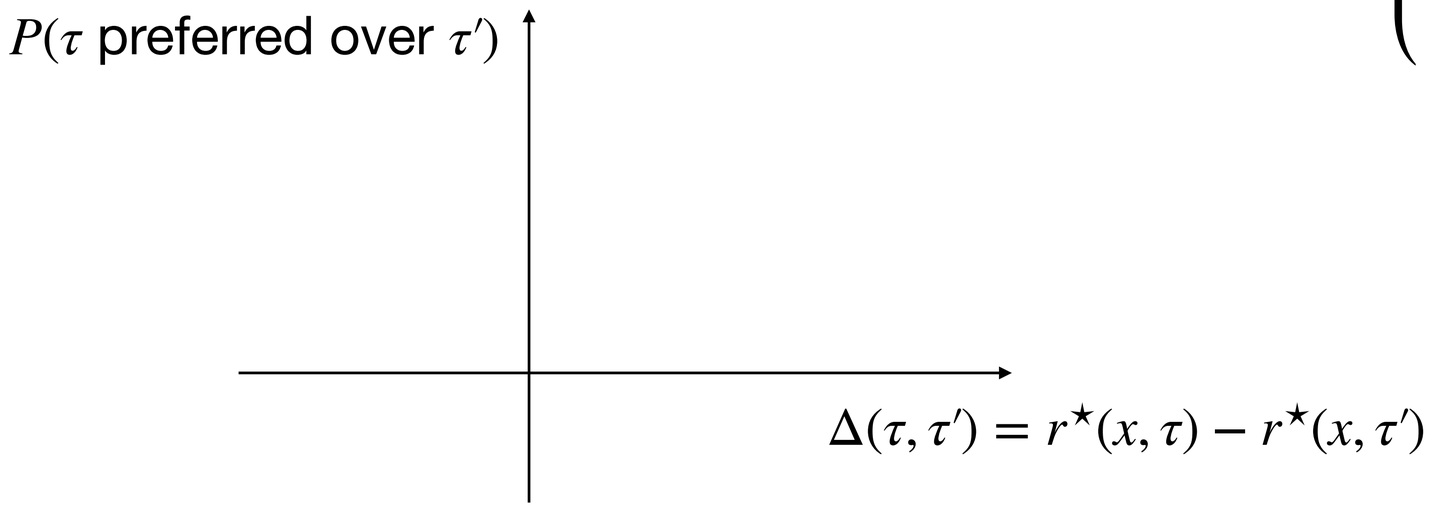
$$P(\tau \text{ is prefered over } \tau' \text{ given } x) = \frac{1}{1 + \exp\left(-\left(\underbrace{r^\star(x, \tau) - r^\star(x, \tau')}_{\Delta(\tau, \tau')}\right)\right)}$$

Bradley-Terry Model

Assume there is a ground truth reward $r^\star(x, \tau)$ (i.e., high reward means response is good)

The BT model assumes that humans generate labels based on the following probablistic model:

$$P(\tau \text{ is prefered over } \tau' \text{ given } x) = \frac{1}{1 + \exp\left(-\left(\underbrace{r^\star(x, \tau) - r^\star(x, \tau')}_{\Delta(\tau, \tau')}\right)\right)}$$



Learning reward based on the Bradley-Terry assumption

Given a preference dataset $\mathscr{D} = \{x, \tau, \tau', z\}$, where label $z \in \{1, -1\}$ is generated via BT on $r^\star$

(1 indicates τ is prefered over τ'; -1 otherwise)

Learn a reward model via MLE:

$$\max_r \sum_{x,\tau,\tau',z} \ln \frac{1}{1 + \exp\left(-z \cdot \left(r(x,\tau) - r(x,\tau')\right)\right)}$$

Outline

1. LLM as a policy

2. Learning reward functions from preference data

3. KL-regularized RL

4. DPO

5. REBEL

RL is very good at reward hacking

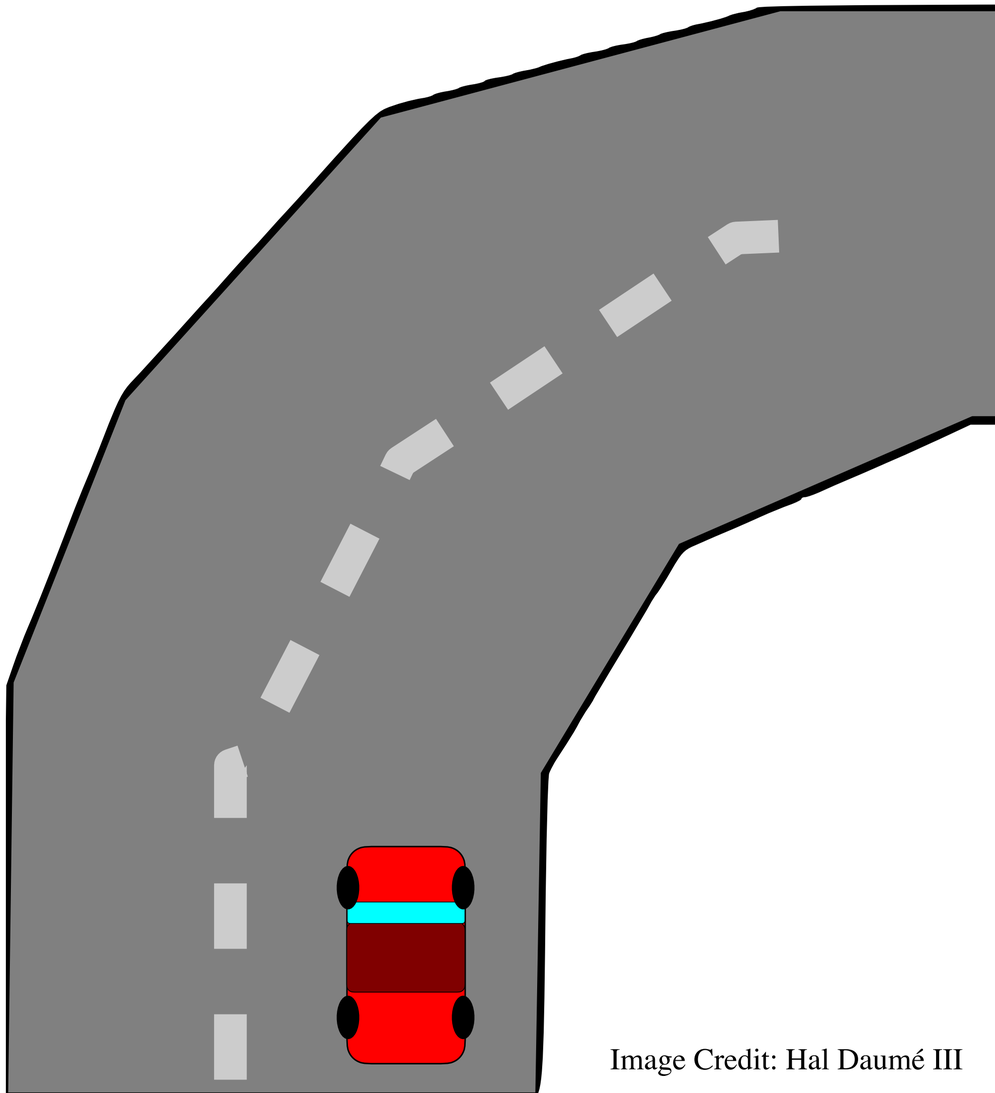


Image Credit: Hal Daumé III

To avoid reward hacking

We form the following KL regularized RL objective

$$J(\pi_\theta) = \mathbb{E}_{x\sim\nu}\left[\mathbb{E}_{\tau\sim\pi(\cdot|x)}\hat{r}(x,\tau) - \beta\mathsf{KL}\left(\pi(\,\cdot\,|\,x)\,\middle|\,\pi_{ref}(\,\cdot\,|\,x)\right)\right]$$

To avoid reward hacking

We form the following KL regularized RL objective

β : controls the strength of KL-reg;

$$J(\pi_\theta) = \mathbb{E}_{x\sim\nu}\left[\mathbb{E}_{\tau\sim\pi(\cdot|x)}\hat{r}(x,\tau) - \beta \mathsf{KL}\left(\pi(\cdot\,|\,x)\,\middle|\,\pi_{ref}(\cdot\,|\,x)\right)\right]$$

To avoid reward hacking

We form the following KL regularized RL objective

β : controls the strength of KL-reg;

$$J(\pi_\theta) = \mathbb{E}_{x\sim\nu}\left[\mathbb{E}_{\tau\sim\pi(\cdot|x)}\hat{r}(x,\tau) - \beta \mathrm{KL}\left(\pi(\cdot\,|\,x)\,\middle|\,\pi_{ref}(\cdot\,|\,x)\right)\right]$$

"stay close" to the SFT policy π_{ref}.

To avoid reward hacking

We form the following KL regularized RL objective

β : controls the strength of KL-reg;

$$J(\pi_\theta) = \mathbb{E}_{x\sim\nu}\left[\mathbb{E}_{\tau\sim\pi(\cdot|x)}\hat{r}(x,\tau) - \beta \mathrm{KL}\left(\pi(\cdot\,|\,x)\,\middle|\,\pi_{ref}(\cdot\,|\,x)\right)\right]$$

"stay close" to the SFT policy π_{ref}.

Q: Why this can help avoid reward hacking?

How to optimize the KL-reg RL objective

A simple heuristic is to add KL to reward

$$J(\pi_\theta) = \mathbb{E}_{x\sim\nu}\left[\mathbb{E}_{\tau\sim\pi(\cdot|x)}\hat{r}(x,\tau) - \beta \mathsf{KL}\left(\pi(\cdot\,|\,x)\,\middle|\,\pi_{ref}(\cdot\,|\,x)\right)\right]$$

How to optimize the KL-reg RL objective

A simple heuristic is to add KL to reward

$$J(\pi_\theta) = \mathbb{E}_{x\sim\nu}\left[\mathbb{E}_{\tau\sim\pi(\cdot|x)}\hat{r}(x,\tau) - \beta \mathsf{KL}\left(\pi(\cdot\,|\,x)\,\middle|\,\pi_{ref}(\cdot\,|\,x)\right)\right]$$

$$= \mathbb{E}_{x\sim\nu}\left[\mathbb{E}_{\tau\sim\pi(\cdot|x)}\underbrace{\left(\hat{r}(x,\tau) - \beta \ln \frac{\pi(\tau\,|\,x)}{\pi_{ref}(\tau\,|\,x)}\right)}_{:=r_{new}(x,\tau)}\right]$$

How to optimize the KL-reg RL objective

A simple heuristic is to add KL to reward

$$J(\pi_\theta) = \mathbb{E}_{x\sim\nu}\left[\mathbb{E}_{\tau\sim\pi(\cdot|x)}\hat{r}(x,\tau) - \beta \mathsf{KL}\left(\pi(\cdot\,|\,x)\,\middle|\,\pi_{ref}(\cdot\,|\,x)\right)\right]$$

$$= \mathbb{E}_{x\sim\nu}\left[\mathbb{E}_{\tau\sim\pi(\cdot|x)}\underbrace{\left(\hat{r}(x,\tau) - \beta \ln \frac{\pi(\tau\,|\,x)}{\pi_{ref}(\tau\,|\,x)}\right)}_{:=r_{new}(x,\tau)}\right]$$

Run PG (reinforce or PPO) w/ $r_{new}(x,\tau)$ as the reward signal

How to optimize the KL-reg RL objective

A simple heuristic is to add KL to reward

$$J(\pi_\theta) = \mathbb{E}_{x\sim\nu}\left[\mathbb{E}_{\tau\sim\pi(\cdot|x)}\hat{r}(x,\tau) - \beta \mathsf{KL}\left(\pi(\cdot\,|\,x)\,\middle|\,\pi_{ref}(\cdot\,|\,x)\right)\right]$$

$$= \mathbb{E}_{x\sim\nu}\left[\mathbb{E}_{\tau\sim\pi(\cdot|x)}\underbrace{\left(\hat{r}(x,\tau) - \beta \ln \frac{\pi(\tau\,|\,x)}{\pi_{ref}(\tau\,|\,x)}\right)}_{:=r_{new}(x,\tau)}\right]$$

Run PG (reinforce or PPO) w/ $r_{new}(x,\tau)$ as the reward signal

Remark: it works, but it is not the exact gradient

Outline

1. LLM as a policy

2. Learning reward functions from preference data

3. KL-regularized RL

4. DPO

5. REBEL

When models are large…

RM + PPO can be hard to optimize…

At least need to maintain 4 big models in GPU RAM (RM, π, V, π_{ref}…)

KL-reg RL objective

$$J(\pi) = \mathbb{E}_{x\sim\nu}\left[\mathbb{E}_{\tau\sim\pi(\cdot|x)}\hat{r}(x,\tau) - \beta\mathsf{KL}\left(\pi(\,\cdot\,|x)\,\middle|\,\pi_{ref}(\,\cdot\,|x)\right)\right]$$

What's the $\arg\max_{\pi} J(\pi)$?

KL-reg RL objective

$$J(\pi) = \mathbb{E}_{x\sim\nu}\left[\mathbb{E}_{\tau\sim\pi(\cdot|x)}\hat{r}(x,\tau) - \beta\mathsf{KL}\left(\pi(\,\cdot\,|x)\,\middle|\,\pi_{ref}(\,\cdot\,|x)\right)\right]$$

What's the $\arg\max_{\pi} J(\pi)$?

Consider on a (x, τ) pair, what is $\partial J(\pi)/\partial\pi(\tau\,|\,x)$?

KL-reg RL objective

$$J(\pi) = \mathbb{E}_{x\sim\nu}\left[\mathbb{E}_{\tau\sim\pi(\cdot|x)}\hat{r}(x,\tau) - \beta\mathsf{KL}\left(\pi(\,\cdot\,|x)\,\middle|\,\pi_{ref}(\,\cdot\,|x)\right)\right]$$

What's the $\arg\max_{\pi} J(\pi)$?

Consider on a (x,τ) pair, what is $\partial J(\pi)/\partial\pi(\tau\,|\,x)$?

$$\frac{\partial J(\pi)}{\partial\pi(\tau\,|\,x)} = \hat{r}(x,\tau) - \beta\left(\ln\pi(\tau\,|\,x) - \ln\pi_{ref}(\tau\,|\,x) + 1\right)$$

KL-reg RL objective

$$J(\pi) = \mathbb{E}_{x\sim\nu}\left[\mathbb{E}_{\tau\sim\pi(\cdot|x)}\hat{r}(x,\tau) - \beta\mathsf{KL}\left(\pi(\cdot\,|x)\,\middle|\,\pi_{ref}(\cdot\,|x)\right)\right]$$

What's the $\arg\max_{\pi} J(\pi)$?

Consider on a (x,τ) pair, what is $\partial J(\pi)/\partial\pi(\tau\,|\,x)$?

$$\frac{\partial J(\pi)}{\partial\pi(\tau\,|\,x)} = \hat{r}(x,\tau) - \beta\left(\ln\pi(\tau\,|\,x) - \ln\pi_{ref}(\tau\,|\,x) + 1\right)$$

$$\pi(\tau\,|\,x) \propto \pi_{ref}(\tau\,|\,x)\exp\left(\hat{r}(x,\tau)/\beta\right)$$

KL-reg RL objective

$$J(\pi) = \mathbb{E}_{x\sim\nu}\left[\mathbb{E}_{\tau\sim\pi(\cdot|x)}\hat{r}(x,\tau) - \beta \mathsf{KL}\left(\pi(\cdot\,|x)\,\middle|\,\pi_{ref}(\cdot\,|x)\right)\right]$$

What's the $\arg\max_{\pi} J(\pi)$?

Consider on a (x,τ) pair, what is $\partial J(\pi)/\partial\pi(\tau\,|\,x)$?

$$\frac{\partial J(\pi)}{\partial \pi(\tau\,|\,x)} = \hat{r}(x,\tau) - \beta\left(\ln\pi(\tau\,|\,x) - \ln\pi_{ref}(\tau\,|\,x) + 1\right)$$

$$\pi(\tau\,|\,x) \propto \pi_{ref}(\tau\,|\,x)\exp\left(\hat{r}(x,\tau)/\beta\right)$$

$$\pi(\tau\,|\,x) = \pi_{ref}(\tau\,|\,x)\exp\left(\frac{\hat{r}(x,\tau)}{\beta}\right)/Z(x),\ \text{where } Z(x) = \mathbb{E}_{\tau\sim\pi_{ref}(\cdot|x)}\exp(\hat{r}(x,\tau)/\beta)$$

Can we parameterize RM using policies?

In sum, the optimal policy of the KL-reg RL objective is:

$$\hat{\pi}(\tau \mid x) = \frac{\pi_{ref}(\tau \mid x) \cdot \exp\left(\frac{\hat{r}(x, \tau)}{\beta}\right)}{Z(x)}$$

Can we parameterize RM using policies?

In sum, the optimal policy of the KL-reg RL objective is:

$$\hat{\pi}(\tau \mid x) = \frac{\pi_{ref}(\tau \mid x) \cdot \exp\left(\frac{\hat{r}(x,\tau)}{\beta}\right)}{Z(x)}$$

$$\ln \hat{\pi}(\tau \mid x) = \ln \pi_{ref}(\tau \mid x) - \ln Z(x) + \frac{\hat{r}(x,\tau)}{\beta}$$

Can we parameterize RM using policies?

In sum, the optimal policy of the KL-reg RL objective is:

$$\hat{\pi}(\tau \mid x) = \frac{\pi_{ref}(\tau \mid x) \cdot \exp\left(\frac{\hat{r}(x,\tau)}{\beta}\right)}{Z(x)}$$

$$\ln \hat{\pi}(\tau \mid x) = \ln \pi_{ref}(\tau \mid x) - \ln Z(x) + \frac{\hat{r}(x,\tau)}{\beta}$$

$$\hat{r}(x,\tau) = \beta\left(\ln \frac{\hat{\pi}(\tau \mid x)}{\pi_{ref}(\tau \mid x)} + \ln Z(x)\right)$$

Can we parameterize RM using policies?

In sum, the optimal policy of the KL-reg RL objective is:

$$\hat{\pi}(\tau \mid x) = \frac{\pi_{ref}(\tau \mid x) \cdot \exp\left(\frac{\hat{r}(x,\tau)}{\beta}\right)}{Z(x)}$$

$$\ln \hat{\pi}(\tau \mid x) = \ln \pi_{ref}(\tau \mid x) - \ln Z(x) + \frac{\hat{r}(x,\tau)}{\beta}$$

$$\hat{r}(x,\tau) = \beta\left(\ln \frac{\hat{\pi}(\tau \mid x)}{\pi_{ref}(\tau \mid x)} + \ln Z(x)\right)$$

Not done yet, this $Z(x)$ technically contains $\hat{r}$!

Can we parameterize RM using policies?

In sum, the optimal policy of the KL-reg RL objective is:

$$\hat{\pi}(\tau \mid x) = \frac{\pi_{ref}(\tau \mid x) \cdot \exp\left(\frac{\hat{r}(x,\tau)}{\beta}\right)}{Z(x)}$$

$$\ln \hat{\pi}(\tau \mid x) = \ln \pi_{ref}(\tau \mid x) - \ln Z(x) + \frac{\hat{r}(x,\tau)}{\beta}$$

$$\hat{r}(x,\tau) = \beta\left(\ln \frac{\hat{\pi}(\tau \mid x)}{\pi_{ref}(\tau \mid x)} + \ln Z(x)\right)$$

Not done yet, this $Z(x)$ technically contains $\hat{r}$!

But $\ln Z(x)$ is a shift that is independent of τ…

DPO

1. Take any policy π_θ, we can use it to model the reward difference:

$$r_\theta(\tau \mid x) - r_\theta(\tau' \mid x) := \beta \left(\ln \frac{\pi_\theta(\tau \mid x)}{\pi_{ref}(\tau \mid x)} - \ln \frac{\pi_\theta(\tau' \mid x)}{\pi_{ref}(\tau' \mid x)} \right)$$

DPO

1. Take any policy π_θ, we can use it to model the reward difference:

$$r_\theta(\tau \mid x) - r_\theta(\tau' \mid x) := \beta \left(\ln \frac{\pi_\theta(\tau \mid x)}{\pi_{ref}(\tau \mid x)} - \ln \frac{\pi_\theta(\tau' \mid x)}{\pi_{ref}(\tau' \mid x)} \right)$$

2. Now plug this into the MLE loss we had for learning the reward difference:

DPO

1. Take any policy π_θ, we can use it to model the reward difference:

$$r_\theta(\tau \mid x) - r_\theta(\tau' \mid x) := \beta \left(\ln \frac{\pi_\theta(\tau \mid x)}{\pi_{ref}(\tau \mid x)} - \ln \frac{\pi_\theta(\tau' \mid x)}{\pi_{ref}(\tau' \mid x)} \right)$$

2. Now plug this into the MLE loss we had for learning the reward difference:

$$\arg\max_\theta \sum_{x,\tau,\tau',z} \ln \frac{1}{1 + \exp\left(-z \cdot \left(r_\theta(x,\tau) - r_\theta(x,\tau') \right) \right)}$$

DPO

DPO optimizes policy π_θ directly using the following loss:

$$\arg\max_{\theta} \sum_{x,\tau,\tau',z} \ln \frac{1}{1+\exp\left(-z\cdot\beta\left(\ln\frac{\pi_\theta(\tau\,|\,x)}{\pi_{ref}(\tau\,|\,x)}-\ln\frac{\pi_\theta(\tau'\,|\,x)}{\pi_{ref}(\tau'\,|\,x)}\right)\right)}$$

Outline

1. LLM as a policy

2. Learning reward functions from preference data

3. KL-regularized RL

4. DPO

5. REBEL

Reparameterization

Mirror descent indicates the following ideal update:

$$\pi_{t+1}(\tau \mid x) = \pi_t(\tau \mid x) \exp\left(r(x, \tau)/\beta\right)/Z(x)$$

Reparameterization

Mirror descent indicates the following ideal update:

$$\pi_{t+1}(\tau \mid x) = \pi_t(\tau \mid x)\exp\left(r(x,\tau)/\beta\right)/Z(x)$$

---

1. Take log on both sides and rearrange terms, we get

$$r(x,\tau) = \beta\left(\ln\frac{\pi_{t+1}(\tau \mid x)}{\pi_t(\tau \mid x)} + \ln Z(x)\right)$$

Reparameterization

Mirror descent indicates the following ideal update:

$$\pi_{t+1}(\tau \mid x) = \pi_t(\tau \mid x) \exp\left(r(x,\tau)/\beta\right)/Z(x)$$

---

1. Take log on both sides and rearrange terms, we get

$$r(x,\tau) = \beta\left(\ln \frac{\pi_{t+1}(\tau \mid x)}{\pi_t(\tau \mid x)} + \ln Z(x)\right)$$

2. Instead of modeling reward, we model reward difference to cancel $Z(x)$:

$$r(x,\tau) - r(x,\tau') = \beta\left(\ln \frac{\pi_{t+1}(\tau \mid x)}{\pi_t(\tau \mid x)} - \ln \frac{\pi_{t+1}(\tau' \mid x)}{\pi_t(\tau' \mid x)}\right)$$

Reparameterization

We obtained the following relationship between r and π_{t+1} & π_t:

Reparameterization

We obtained the following relationship between r and π_{t+1} & π_t:

$$\forall(x, \tau, \tau') : \quad r(x, \tau) - r(x, \tau') = \beta \left(\ln \frac{\pi_{t+1}(\tau \,|\, x)}{\pi_t(\tau \,|\, x)} - \ln \frac{\pi_{t+1}(\tau' \,|\, x)}{\pi_t(\tau' \,|\, x)} \right)$$

Reparameterization

We obtained the following relationship between r and π_{t+1} & π_t:

$$\forall(x,\tau,\tau'):\ \ r(x,\tau)-r(x,\tau')=\beta\left(\ln\frac{\pi_{t+1}(\tau\,|\,x)}{\pi_t(\tau\,|\,x)}-\ln\frac{\pi_{t+1}(\tau'\,|\,x)}{\pi_t(\tau'\,|\,x)}\right)$$

This indicates π_{t+1} is **the minimizer** of the following least square regression problem:

Reparameterization

We obtained the following relationship between r and π_{t+1} & π_t:

$$\forall(x,\tau,\tau'):\ \ r(x,\tau)-r(x,\tau')=\beta\left(\ln\frac{\pi_{t+1}(\tau\,|\,x)}{\pi_t(\tau\,|\,x)}-\ln\frac{\pi_{t+1}(\tau'\,|\,x)}{\pi_t(\tau'\,|\,x)}\right)$$

This indicates π_{t+1} is **the minimizer** of the following least square regression problem:

$$\mathbb{E}_{x,\tau,\tau'}\left(\beta\left(\ln\frac{\pi_{t+1}(\tau\,|\,x)}{\pi_t(\tau\,|\,x)}-\ln\frac{\pi_{t+1}(\tau'\,|\,x)}{\pi_t(\tau'\,|\,x)}\right)-\big(r(x,\tau)-r(x,\tau')\big)\right)^2$$

Reparameterization

We obtained the following relationship between r and π_{t+1} & π_t:

$$\forall(x,\tau,\tau'):\ \ r(x,\tau)-r(x,\tau')=\beta\left(\ln\frac{\pi_{t+1}(\tau\,|\,x)}{\pi_t(\tau\,|\,x)}-\ln\frac{\pi_{t+1}(\tau'\,|\,x)}{\pi_t(\tau'\,|\,x)}\right)$$

This indicates π_{t+1} is **the minimizer** of the following least square regression problem:

π_{t+1} should be the minimizer regardless of what the distribution is;

$$\mathbb{E}_{x,\tau,\tau'}\left(\beta\left(\ln\frac{\pi_{t+1}(\tau\,|\,x)}{\pi_t(\tau\,|\,x)}-\ln\frac{\pi_{t+1}(\tau'\,|\,x)}{\pi_t(\tau'\,|\,x)}\right)-\Big(r(x,\tau)-r(x,\tau')\Big)\right)^2$$

Reparameterization

We obtained the following relationship between r and π_{t+1} & π_t:

$$\forall(x,\tau,\tau'):\ \ r(x,\tau)-r(x,\tau')=\beta\left(\ln\frac{\pi_{t+1}(\tau\,|\,x)}{\pi_t(\tau\,|\,x)}-\ln\frac{\pi_{t+1}(\tau'\,|\,x)}{\pi_t(\tau'\,|\,x)}\right)$$

This indicates π_{t+1} is **the minimizer** of the following least square regression problem:

π_{t+1} should be the minimizer regardless of what the distribution is;

$$\mathbb{E}_{x,\tau,\tau'}\left(\beta\left(\ln\frac{\pi_{t+1}(\tau\,|\,x)}{\pi_t(\tau\,|\,x)}-\ln\frac{\pi_{t+1}(\tau'\,|\,x)}{\pi_t(\tau'\,|\,x)}\right)-\left(r(x,\tau)-r(x,\tau')\right)\right)^2$$

In pratice, we often use

$x,\tau\sim\pi_t(\,\cdot\,|\,x),\tau'\sim\pi_t(\,\cdot\,|\,x)$

Reparameterization

We obtained the following relationship between r and π_{t+1} & π_t:

$$\forall(x,\tau,\tau'):\ \ r(x,\tau)-r(x,\tau')=\beta\left(\ln\frac{\pi_{t+1}(\tau\,|\,x)}{\pi_t(\tau\,|\,x)}-\ln\frac{\pi_{t+1}(\tau'\,|\,x)}{\pi_t(\tau'\,|\,x)}\right)$$

This indicates π_{t+1} is **the minimizer** of the following least square regression problem:

π_{t+1} should be the minimizer regardless of what the distribution is;

$$\mathbb{E}_{x,\tau,\tau'}\left(\beta\left(\ln\frac{\pi_{t+1}(\tau\,|\,x)}{\pi_t(\tau\,|\,x)}-\ln\frac{\pi_{t+1}(\tau'\,|\,x)}{\pi_t(\tau'\,|\,x)}\right)-\underbrace{\big(r(x,\tau)-r(x,\tau')\big)}_{\text{Relative reward}}\right)^2$$

In pratice, we often use $x,\tau\sim\pi_t(\,\cdot\,|\,x),\tau'\sim\pi_t(\,\cdot\,|\,x)$

REBEL algorithm

Put things together, we arrive at the following iterative algorithm:

Given π_t, we compute π_{t+1} via least square regression:

REBEL algorithm

Put things together, we arrive at the following iterative algorithm:

Given π_t, we compute π_{t+1} via least square regression:

$$\pi_{t+1} = \arg\min_{\pi} \mathbb{E}_{x,(\tau,\tau')\sim\pi_t(\cdot|x)} \left(\beta \left(\ln \frac{\pi(\tau \,|\, x)}{\pi_t(\tau \,|\, x)} - \ln \frac{\pi(\tau' \,|\, x)}{\pi_t(\tau' \,|\, x)} \right) - \Big(r(x,\tau) - r(x,\tau') \Big) \right)^2$$

REBEL algorithm

Put things together, we arrive at the following iterative algorithm:

Given π_t, we compute π_{t+1} via least square regression:

$$\pi_{t+1} = \arg\min_{\pi} \mathbb{E}_{x,(\tau,\tau')\sim\pi_t(\cdot|x)} \left(\beta \left(\ln \frac{\pi(\tau \,|\, x)}{\pi_t(\tau \,|\, x)} - \ln \frac{\pi(\tau' \,|\, x)}{\pi_t(\tau' \,|\, x)} \right) - \big(r(x,\tau) - r(x,\tau') \big) \right)^2$$

sample τ, τ' from the latest policy $\pi_t(\,\cdot\,|\,x)$, independently;

REBEL algorithm

Put things together, we arrive at the following iterative algorithm:

Given π_t, we compute π_{t+1} via least square regression:

$$\pi_{t+1} = \arg\min_{\pi} \mathbb{E}_{x,(\tau,\tau')\sim\pi_t(\cdot|x)} \left(\beta \left(\ln \frac{\pi(\tau \,|\, x)}{\pi_t(\tau \,|\, x)} - \ln \frac{\pi(\tau' \,|\, x)}{\pi_t(\tau' \,|\, x)} \right) - \underbrace{\big(r(x,\tau) - r(x,\tau') \big)}_{\text{Relative reward}} \right)^2$$

sample τ, τ' from the latest policy $\pi_t(\,\cdot\,|\,x)$, independently;

REBEL algorithm

Put things together, we arrive at the following iterative algorithm:

Given π_t, we compute π_{t+1} via least square regression:

$$\pi_{t+1} = \arg\min_{\pi} \mathbb{E}_{x,(\tau,\tau')\sim\pi_t(\cdot|x)} \left(\underbrace{\beta \left(\ln \frac{\pi(\tau \,|\, x)}{\pi_t(\tau \,|\, x)} - \ln \frac{\pi(\tau' \,|\, x)}{\pi_t(\tau' \,|\, x)} \right)}_{\text{Regressor}} - \underbrace{\big(r(x,\tau) - r(x,\tau') \big)}_{\text{Relative reward}} \right)^2$$

sample τ, τ' from the latest policy $\pi_t(\,\cdot\,|\,x)$, independently;

REBEL Theoretical Analysis

Assumption 1: over T iterations, we have the following for some ϵ:

$$\mathbb{E}_{x,\, y\sim\pi_t(\cdot|x),\, y'\sim\pi_t(\cdot|x)}\left(\frac{1}{\eta}\left(\ln\frac{\pi_{t+1}(y\mid x)}{\pi_t(y\mid x)}-\ln\frac{\pi_{t+1}(y'\mid x)}{\pi_t(y'\mid x)}\right)-\big(r(x,y)-r(x,y')\big)\right)^2\leq\epsilon$$

REBEL Theoretical Analysis

Assumption 1: over T iterations, we have the following for some ϵ:

$$\mathbb{E}_{x,\, y\sim\pi_t(\cdot|x),\, y'\sim\pi_t(\cdot|x)}\left(\frac{1}{\eta}\left(\ln\frac{\pi_{t+1}(y\mid x)}{\pi_t(y\mid x)}-\ln\frac{\pi_{t+1}(y'\mid x)}{\pi_t(y'\mid x)}\right)-\big(r(x,y)-r(x,y')\big)\right)^2\leq\epsilon$$

Data Coverage: Given a test policy π, we denote the concentrability coefficient as:

$$C_{\pi_t\to\pi}=\max_{x,y}\frac{\pi(y\mid x)}{\pi_t(y\mid x)}$$

π_t covers π if $C_{\pi_t\to\pi}<+\infty$

REBEL Theorem

Under assumption 1, after T many iterations, among the learned policies $\pi_1, \ldots, \pi_T$, there must exist a policy π_t such that:

$$\forall \pi^* : \ \mathbb{E}_{x,\, y \sim \pi^*(\cdot|x)} r(x, y) - \mathbb{E}_{x,\, y \sim \pi_t(\cdot|x)} r(x, y) \leq O\left(\sqrt{\frac{1}{T}} + \sqrt{C_{\max} \epsilon} \right)$$

$$C_{\max} = \max_{\pi \in \{\pi_1, \ldots, \pi_T\}} C_{\pi \to \pi^*}$$

Theorem – Lemma 1

Assume $\max_{x,y,t} |A_t(x,y)| \leq A \in \mathbb{R}^+$**, and** $\pi_0(\cdot \mid x)$ **is uniform over** $\mathcal{Y}$**.**

Then, $\eta = \sqrt{\ln(|\mathcal{Y}|)/(A^2 T)}$**, for the sequence of policies computed by REBEL, we have:**

$$\forall \pi, x: \quad \sum_{t=0}^{T-1} \mathbb{E}_{y \sim \pi(\cdot|x)} A_t(x,y) \leq 2A\sqrt{\ln(|\mathcal{Y}|)\, T}\,.$$

Theorem – Proof Lemma 1

Start with $\pi_{t+1}(y \mid x) = \pi_t(y \mid x)\exp(\eta A_t(x, y))/Z_t(x),$

where $Z_t(x)$ **is the normalization constant**

Theorem – Proof Lemma 1

Start with $\pi_{t+1}(y \mid x) = \pi_t(y \mid x)\exp(\eta A_t(x, y))/Z_t(x)$,
where $Z_t(x)$ is the normalization constant

$$\pi_{t+1}(y \mid x) = \pi_t(y \mid x)\exp(\eta A_t(x, y))/Z_t(x),$$

Theorem – Proof Lemma 1

Start with $\pi_{t+1}(y \mid x) = \pi_t(y \mid x)\exp(\eta A_t(x, y))/Z_t(x)$,
where $Z_t(x)$ is the normalization constant

$$\pi_{t+1}(y \mid x) = \pi_t(y \mid x)\exp(\eta A_t(x, y))/Z_t(x),$$

$$\log \pi_{t+1}(y \mid x) = \log \pi_t(y \mid x) + \eta A_t(x, y) - \log Z_t(x),$$

Theorem – Proof Lemma 1

Start with $\pi_{t+1}(y \mid x) = \pi_t(y \mid x)\exp(\eta A_t(x, y))/Z_t(x)$,
where $Z_t(x)$ is the normalization constant

$$\pi_{t+1}(y \mid x) = \pi_t(y \mid x)\exp(\eta A_t(x, y))/Z_t(x),$$

$$\log \pi_{t+1}(y \mid x) = \log \pi_t(y \mid x) + \eta A_t(x, y) - \log Z_t(x),$$

$$E_{y\sim\pi}\Big[\log \pi_{t+1}(y \mid x)\Big] = E_{y\sim\pi}\Big[\log \pi_t(y \mid x)\Big] + E_{y\sim\pi}\Big[\eta A_t(x, y)\Big] - E_{y\sim\pi}\Big[\log Z_t(x)\Big]$$

Theorem – Proof Lemma 1

Start with $\pi_{t+1}(y \mid x) = \pi_t(y \mid x)\exp(\eta A_t(x, y))/Z_t(x)$,
where $Z_t(x)$ is the normalization constant

$$\pi_{t+1}(y \mid x) = \pi_t(y \mid x)\exp(\eta A_t(x, y))/Z_t(x),$$

$$\log \pi_{t+1}(y \mid x) = \log \pi_t(y \mid x) + \eta A_t(x, y) - \log Z_t(x),$$

$$E_{y\sim\pi}\Big[\log \pi_{t+1}(y \mid x)\Big] = E_{y\sim\pi}\Big[\log \pi_t(y \mid x)\Big] + E_{y\sim\pi}\Big[\eta A_t(x, y)\Big] - E_{y\sim\pi}\Big[\log Z_t(x)\Big]$$

$$E_{y\sim\pi}\Big[\log \pi_{t+1}(y \mid x)\Big] = E_{y\sim\pi}\Big[\log \pi_t(y \mid x)\Big] + E_{y\sim\pi}\Big[\log \pi(y \mid x) - \log \pi(y \mid x)\Big] + E_{y\sim\pi}\Big[\eta A_t(x, y)\Big] - E_{y\sim\pi}\Big[\log Z_t(x)\Big]$$

Theorem – Proof Lemma 1

Start with $\pi_{t+1}(y \mid x) = \pi_t(y \mid x)\exp(\eta A_t(x, y))/Z_t(x)$,
where $Z_t(x)$ is the normalization constant

$$\pi_{t+1}(y \mid x) = \pi_t(y \mid x)\exp(\eta A_t(x, y))/Z_t(x),$$

$$\log \pi_{t+1}(y \mid x) = \log \pi_t(y \mid x) + \eta A_t(x, y) - \log Z_t(x),$$

$$E_{y\sim\pi}\Big[\log \pi_{t+1}(y \mid x)\Big] = E_{y\sim\pi}\Big[\log \pi_t(y \mid x)\Big] + E_{y\sim\pi}\Big[\eta A_t(x, y)\Big] - E_{y\sim\pi}\Big[\log Z_t(x)\Big]$$

$$E_{y\sim\pi}\Big[\log \pi_{t+1}(y \mid x)\Big] = E_{y\sim\pi}\Big[\log \pi_t(y \mid x)\Big] + E_{y\sim\pi}\Big[\log \pi(y \mid x) - \log \pi(y \mid x)\Big] + E_{y\sim\pi}\Big[\eta A_t(x, y)\Big] - E_{y\sim\pi}\Big[\log Z_t(x)\Big]$$

$$-\mathrm{D}_{\mathrm{KL}}(\pi||\pi_{t+1}) = -\mathrm{D}_{\mathrm{KL}}(\pi||\pi_t) + E_{y\sim\pi}\Big[\eta A_t(x, y)\Big] - E_{y\sim\pi}\Big[\log Z_t(x)\Big]$$

Theorem – Proof Lemma 1

Start with $\pi_{t+1}(y \mid x) = \pi_t(y \mid x)\exp(\eta A_t(x, y))/Z_t(x)$,
where $Z_t(x)$ is the normalization constant

$$\pi_{t+1}(y \mid x) = \pi_t(y \mid x)\exp(\eta A_t(x, y))/Z_t(x),$$

$$\log \pi_{t+1}(y \mid x) = \log \pi_t(y \mid x) + \eta A_t(x, y) - \log Z_t(x),$$

$$E_{y\sim\pi}\Big[\log \pi_{t+1}(y \mid x)\Big] = E_{y\sim\pi}\Big[\log \pi_t(y \mid x)\Big] + E_{y\sim\pi}\Big[\eta A_t(x, y)\Big] - E_{y\sim\pi}\Big[\log Z_t(x)\Big]$$

$$E_{y\sim\pi}\Big[\log \pi_{t+1}(y \mid x)\Big] = E_{y\sim\pi}\Big[\log \pi_t(y \mid x)\Big] + E_{y\sim\pi}\Big[\log \pi(y \mid x) - \log \pi(y \mid x)\Big] + E_{y\sim\pi}\Big[\eta A_t(x, y)\Big] - E_{y\sim\pi}\Big[\log Z_t(x)\Big]$$

$$-\mathrm{D}_{\mathrm{KL}}(\pi||\pi_{t+1}) = -\mathrm{D}_{\mathrm{KL}}(\pi||\pi_t) + E_{y\sim\pi}\Big[\eta A_t(x, y)\Big] - \underbrace{E_{y\sim\pi}\Big[\log Z_t(x)\Big]}$$

term

Theorem – Proof Lemma 1

$$\log Z_t(x)$$

$$\log Z_t(x) = \ln \mathbb{E}_{y \sim \pi_t(\cdot|x)} \exp\Big(\eta A_t(x, y)\Big)$$

Theorem – Proof Lemma 1

$$\log Z_t(x)$$

$$\log Z_t(x) = \ln \mathbb{E}_{y \sim \pi_t(\cdot|x)} \exp\big(\eta A_t(x, y)\big)$$

Using $\exp(x) \leq 1 + x + x^2$ **for any** $x \leq 1$:

Theorem – Proof Lemma 1

$$\log Z_t(x)$$

$$\log Z_t(x) = \ln \mathbb{E}_{y \sim \pi_t(\cdot|x)} \exp\big(\eta A_t(x, y)\big)$$

Using $\exp(x) \leq 1 + x + x^2$ **for any** $x \leq 1$:

$$\log Z_t(x) \leq \ln \mathbb{E}_{y \sim \pi_t(\cdot|x)} \big(1 + \eta A_t(x, y) + \eta^2 A_t(x, y)^2\big)$$

Theorem – Proof Lemma 1

$$\log Z_t(x)$$

$$\log Z_t(x) = \ln \mathbb{E}_{y \sim \pi_t(\cdot|x)} \exp\big(\eta A_t(x, y)\big)$$

Using $\exp(x) \leq 1 + x + x^2$ **for any** $x \leq 1$:

$$\log Z_t(x) \leq \ln \mathbb{E}_{y \sim \pi_t(\cdot|x)} \big(1 + \eta A_t(x, y) + \eta^2 A_t(x, y)^2\big)$$

$$\leq \ln(1 + 0 + \eta^2 A^2)$$

Theorem – Proof Lemma 1

$$\log Z_t(x)$$

$$\log Z_t(x) = \ln \mathbb{E}_{y \sim \pi_t(\cdot|x)} \exp\big(\eta A_t(x, y)\big)$$

Using $\exp(x) \leq 1 + x + x^2$ **for any** $x \leq 1$:

$$\log Z_t(x) \leq \ln \mathbb{E}_{y \sim \pi_t(\cdot|x)} \big(1 + \eta A_t(x, y) + \eta^2 A_t(x, y)^2\big)$$

$$\leq \ln(1 + 0 + \eta^2 A^2)$$

$$\leq \eta^2 A^2$$

Theorem – Proof Lemma 1

$$-\mathrm{D}_{\mathrm{KL}}(\pi||\pi_{t+1}) = -\mathrm{D}_{\mathrm{KL}}(\pi||\pi_t) + E_{y\sim\pi}\Big[\eta A_t(x,y)\Big] - E_{y\sim\pi}\Big[\log Z_t(x)\Big]$$

$$E_{y\sim\pi}\Big[\eta A_t(x,y)\Big] = \mathrm{D}_{\mathrm{KL}}(\pi||\pi_{t+1}) - \mathrm{D}_{\mathrm{KL}}(\pi||\pi_t) - E_{y\sim\pi}\Big[\log Z_t(x)\Big]$$

Theorem – Proof Lemma 1

$$-\mathrm{D}_{\mathrm{KL}}(\pi||\pi_{t+1}) = -\mathrm{D}_{\mathrm{KL}}(\pi||\pi_t) + E_{y\sim\pi}\Big[\eta A_t(x,y)\Big] - E_{y\sim\pi}\Big[\log Z_t(x)\Big]$$

$$E_{y\sim\pi}\Big[\eta A_t(x,y)\Big] = \mathrm{D}_{\mathrm{KL}}(\pi||\pi_{t+1}) - \mathrm{D}_{\mathrm{KL}}(\pi||\pi_t) - E_{y\sim\pi}\Big[\log Z_t(x)\Big]$$

$$E_{y\sim\pi}\Big[\eta A_t(x,y)\Big] \leq \mathrm{D}_{\mathrm{KL}}(\pi||\pi_{t+1}) - \mathrm{D}_{\mathrm{KL}}(\pi||\pi_t) + E_{y\sim\pi} - \eta^2 A^2$$

Theorem – Proof Lemma 1

$$-\mathrm{D}_{\mathrm{KL}}(\pi||\pi_{t+1}) = -\mathrm{D}_{\mathrm{KL}}(\pi||\pi_t) + E_{y\sim\pi}\Big[\eta A_t(x,y)\Big] - E_{y\sim\pi}\Big[\log Z_t(x)\Big]$$

$$E_{y\sim\pi}\Big[\eta A_t(x,y)\Big] = \mathrm{D}_{\mathrm{KL}}(\pi||\pi_{t+1}) - \mathrm{D}_{\mathrm{KL}}(\pi||\pi_t) - E_{y\sim\pi}\Big[\log Z_t(x)\Big]$$

$$E_{y\sim\pi}\Big[\eta A_t(x,y)\Big] \leq \mathrm{D}_{\mathrm{KL}}(\pi||\pi_{t+1}) - \mathrm{D}_{\mathrm{KL}}(\pi||\pi_t) + E_{y\sim\pi} - \eta^2 A^2$$

$$\sum_{t=0}^{T-1} E_{y\sim\pi}\Big[A_t(x,y)\Big] \leq \frac{1}{\eta}\sum_{t=0}^{T-1}\mathrm{D}_{\mathrm{KL}}(\pi||\pi_{t+1}) - \frac{1}{\eta}\sum_{t=0}^{T-1}\mathrm{D}_{\mathrm{KL}}(\pi||\pi_t) + E_{y\sim\pi} - \frac{1}{\eta}\sum_{t=0}^{T-1}\eta^2 A^2$$

Theorem – Proof Lemma 1

$$-\mathrm{D}_{\mathrm{KL}}(\pi||\pi_{t+1}) = -\mathrm{D}_{\mathrm{KL}}(\pi||\pi_t) + E_{y\sim\pi}\Big[\eta A_t(x,y)\Big] - E_{y\sim\pi}\Big[\log Z_t(x)\Big]$$

$$E_{y\sim\pi}\Big[\eta A_t(x,y)\Big] = \mathrm{D}_{\mathrm{KL}}(\pi||\pi_{t+1}) - \mathrm{D}_{\mathrm{KL}}(\pi||\pi_t) - E_{y\sim\pi}\Big[\log Z_t(x)\Big]$$

$$E_{y\sim\pi}\Big[\eta A_t(x,y)\Big] \leq \mathrm{D}_{\mathrm{KL}}(\pi||\pi_{t+1}) - \mathrm{D}_{\mathrm{KL}}(\pi||\pi_t) + E_{y\sim\pi} - \eta^2 A^2$$

$$\sum_{t=0}^{T-1} E_{y\sim\pi}\Big[A_t(x,y)\Big] \leq \frac{1}{\eta}\sum_{t=0}^{T-1}\mathrm{D}_{\mathrm{KL}}(\pi||\pi_{t+1}) - \frac{1}{\eta}\sum_{t=0}^{T-1}\mathrm{D}_{\mathrm{KL}}(\pi||\pi_t) + E_{y\sim\pi} - \frac{1}{\eta}\sum_{t=0}^{T-1}\eta^2 A^2$$

$$\sum_{t=0}^{T-1} E_{y\sim\pi}\Big[A_t(x,y)\Big] \leq \mathrm{D}_{\mathrm{KL}}(\pi||\pi_0) - \mathrm{D}_{\mathrm{KL}}(\pi||\pi_T) - T\eta A^2$$

Theorem – Proof Lemma 1

$$-\mathrm{D}_{\mathrm{KL}}(\pi||\pi_{t+1}) = -\mathrm{D}_{\mathrm{KL}}(\pi||\pi_t) + E_{y\sim\pi}\Big[\eta A_t(x,y)\Big] - E_{y\sim\pi}\Big[\log Z_t(x)\Big]$$

$$E_{y\sim\pi}\Big[\eta A_t(x,y)\Big] = \mathrm{D}_{\mathrm{KL}}(\pi||\pi_{t+1}) - \mathrm{D}_{\mathrm{KL}}(\pi||\pi_t) - E_{y\sim\pi}\Big[\log Z_t(x)\Big]$$

$$E_{y\sim\pi}\Big[\eta A_t(x,y)\Big] \leq \mathrm{D}_{\mathrm{KL}}(\pi||\pi_{t+1}) - \mathrm{D}_{\mathrm{KL}}(\pi||\pi_t) + E_{y\sim\pi} - \eta^2 A^2$$

$$\sum_{t=0}^{T-1} E_{y\sim\pi}\Big[A_t(x,y)\Big] \leq \frac{1}{\eta}\sum_{t=0}^{T-1}\mathrm{D}_{\mathrm{KL}}(\pi||\pi_{t+1}) - \frac{1}{\eta}\sum_{t=0}^{T-1}\mathrm{D}_{\mathrm{KL}}(\pi||\pi_t) + E_{y\sim\pi} - \frac{1}{\eta}\sum_{t=0}^{T-1}\eta^2 A^2$$

$$\sum_{t=0}^{T-1} E_{y\sim\pi}\Big[A_t(x,y)\Big] \leq \mathrm{D}_{\mathrm{KL}}(\pi||\pi_0) - \mathrm{D}_{\mathrm{KL}}(\pi||\pi_T) - T\eta A^2$$

$$\sum_{t=0}^{T-1} E_{y\sim\pi}\Big[A_t(x,y)\Big] \leq \mathrm{D}_{\mathrm{KL}}(\pi||\pi_0) - T\eta A^2 \leq \frac{\ln|y|}{\eta} + T\eta A^2$$

Theorem – Proof Lemma 1

Set $\eta = \sqrt{\frac{\ln(|y|)}{A^2 T}}$

$$\sum_{t=0}^{T-1} E_{y\sim\pi}\left[A_t(x,y)\right] \leq 2A\sqrt{\ln(|y|T}$$

Theorem – Proof

Under assumption 1, after T many iterations, among the learned policies $\pi_1, \ldots, \pi_T$, there must exist a policy π_t such that:

$$\forall \pi^* : \ \mathbb{E}_{x,\, y \sim \pi^*(\cdot|x)} r(x, y) - \mathbb{E}_{x,\, y \sim \pi_t(\cdot|x)} r(x, y) \leq O\left(\sqrt{\frac{1}{T}} + \sqrt{C_{\max} \epsilon} \right)$$

Theorem – Proof

Under assumption 1, after T many iterations, among the learned policies $\pi_1, \ldots, \pi_T$, there must exist a policy π_t such that:

$$\forall \pi^* : \ \mathbb{E}_{x,\, y \sim \pi^*(\cdot|x)} r(x, y) - \mathbb{E}_{x,\, y \sim \pi_t(\cdot|x)} r(x, y) \leq O\left(\sqrt{\frac{1}{T}} + \sqrt{C_{\max} \epsilon} \right)$$

Consider a comparator policy π^*. We start with the performance difference between π^* and the uniform mixture policy.

Theorem – Proof

Under assumption 1, after T many iterations, among the learned policies $\pi_1, \ldots, \pi_T$, there must exist a policy π_t such that:

$$\forall \pi^* : \ \mathbb{E}_{x,\, y \sim \pi^*(\cdot|x)} r(x, y) - \mathbb{E}_{x,\, y \sim \pi_t(\cdot|x)} r(x, y) \leq O\left(\sqrt{\frac{1}{T}} + \sqrt{C_{\max} \epsilon} \right)$$

Consider a comparator policy π^*. We start with the performance difference between π^* and the uniform mixture policy.

$$\frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E}_{x,\, y \sim \pi^*(\cdot|x)} \left(A^{\pi_t}(x, y) \right)$$

Theorem – Proof

Under assumption 1, after T many iterations, among the learned policies $\pi_1, \ldots, \pi_T$, there must exist a policy π_t such that:

$$\forall \pi^* : \ \mathbb{E}_{x,\, y\sim\pi^*(\cdot|x)} r(x,y) - \mathbb{E}_{x,\, y\sim\pi_t(\cdot|x)} r(x,y) \leq O\left(\sqrt{\frac{1}{T}} + \sqrt{C_{\max}\epsilon}\right)$$

Consider a comparator policy π^*. We start with the performance difference between π^* and the uniform mixture policy.

$$\frac{1}{T}\sum_{t=0}^{T-1} \mathbb{E}_{x,\, y\sim\pi^*(\cdot|x)} \left(A^{\pi_t}(x,y)\right)$$

$$= \frac{1}{T}\sum_{t=0}^{T-1} \mathbb{E}_{x,\, y\sim\pi^*(\cdot|x)} \left(A_t(x,y)\right) + \frac{1}{T}\sum_{t=0}^{T-1} \mathbb{E}_{x,\, y\sim\pi^*(\cdot|x)} \left(A^{\pi_t}(x,y) - A_t(x,y)\right)$$

Theorem – Proof

Under assumption 1, after T many iterations, among the learned policies $\pi_1, \ldots, \pi_T$, there must exist a policy π_t such that:

$$\forall \pi^* : \ \mathbb{E}_{x,\, y\sim\pi^*(\cdot|x)} r(x,y) - \mathbb{E}_{x,\, y\sim\pi_t(\cdot|x)} r(x,y) \leq O\left(\sqrt{\frac{1}{T}} + \sqrt{C_{\max}\epsilon}\right)$$

Consider a comparator policy π^*. We start with the performance difference between π^* and the uniform mixture policy.

$$\frac{1}{T}\sum_{t=0}^{T-1} \mathbb{E}_{x,\, y\sim\pi^*(\cdot|x)}\left(A^{\pi_t}(x,y)\right)$$

$$= \frac{1}{T}\sum_{t=0}^{T-1} \mathbb{E}_{x,\, y\sim\pi^*(\cdot|x)}\left(A_t(x,y)\right) + \frac{1}{T}\sum_{t=0}^{T-1} \mathbb{E}_{x,\, y\sim\pi^*(\cdot|x)}\left(A^{\pi_t}(x,y) - A_t(x,y)\right)$$

$$\leq 2A\sqrt{\frac{\ln(|\mathcal{Y}|)}{T}} + \frac{1}{T}\sum_{t=0}^{T-1}\sqrt{\mathbb{E}_x \mathbb{E}_{y\sim\pi^*(\cdot|x)}\left(A^{\pi_t}(x,y) - A_t(x,y)\right)^2}\,.$$

Summary

RLHF is a tool for post-training LLMs so that llms can understand and follow human instructions

Summary

RLHF is a tool for post-training LLMs so that llms can understand and follow human instructions

Reward Model (RM) is learned from human feedback (i.e., pair-wise preference)

Summary

RLHF is a tool for post-training LLMs so that llms can understand and follow human instructions

Reward Model (RM) is learned from human feedback (i.e., pair-wise preference)

RM learning is based on the Bradley-Terry model

Summary

RLHF is a tool for post-training LLMs so that llms can understand and follow human instructions

Reward Model (RM) is learned from human feedback (i.e., pair-wise preference)

RM learning is based on the Bradley-Terry model

KL regularization is important to avoid hacking the learned RM

Summary

RLHF is a tool for post-training LLMs so that llms can understand and follow human instructions

Reward Model (RM) is learned from human feedback (i.e., pair-wise preference)

RM learning is based on the Bradley-Terry model

KL regularization is important to avoid hacking the learned RM

DPO reparameterizes the reward difference via BT

Summary

RLHF is a tool for post-training LLMs so that llms can understand and follow human instructions

Reward Model (RM) is learned from human feedback (i.e., pair-wise preference)

RM learning is based on the Bradley-Terry model

KL regularization is important to avoid hacking the learned RM

DPO reparameterizes the reward difference via BT

REBEL reparameterizes the reward difference without BT