Lecture 9: RLHF and Guest Lecture on Values and Alignment

Emma Brunskill.[3]

CS234 Reinforcement Learning.

Winter 2025

[2]Slides with Red Text Font from Rafailov, Sharma, Mitchell 2024 guest lecture
[3]Slides with Red Text Font from Rafailov, Sharma, Mitchell 2024 guest lecture

Midterm

- In class on Thursday
- You are allowed 1 side of 1 8.5" x 11" sheet of notes
- All material through today's lecture is eligible for the exam
- See Ed post for past related practice materials
- Good luck!

[5] Slides with Red Text Font from Rafailov, Sharma, Mitchell 2024 guest lecture

Refresh Your Understanding L9N1

Select all that are true

(a) The Bradley Terry model expresses the probability that someone will select option b_i over b_j

(b) Using preference tuples and the Bradley Terry model, one can learn a model of the reward function

(c) The resulting reward function can be shifted by any constant and will not change the resulting preferences

(d) The resulting reward function can be multiplied by any constant and will not change the resulting preferences

(e) In RLHF we update the reward model after each PPO roll out

(f) Not sure

[7] Slides with Red Text Font from Rafailov, Sharma, Mitchell 2024 guest lecture[8]

Refresh Your Understanding L9N1 Solutions

Select all that are true

(a) The Bradley Terry model expresses the probability that someone will select option b_i over b_j

(b) Using preference tuples and the Bradley Terry model, one can learn a model of the reward function

(c) The resulting reward function can be shifted by any constant and will not change the resulting preferences

(d) The resulting reward function can be multiplied by any constant and will not change the resulting preferences

(e) In RLHF we update the reward model after each PPO roll out

(f) Not sure

1,2,3 are true. 4 is false: for example, we cannot multiply the rewards by -1 and preserve the ordering.

[9] Slides with Red Text Font from Rafailov, Sharma, Mitchell 2024 guest lecture

Class Structure

- Last time: Imitation Learning (Max Entropy IRL) and RLHF
- This time: RLHF and Direct Preference Optimization and value alignment
- Next time: Midterm

[11] Slides with Red Text Font from Rafailov, Sharma, Mitchell 2024 guest lecture[12]

Today

- RLHF for LLM
- Direct Preference Optimization

[13] Slides with Red Text Font from Rafailov, Sharma, Mitchell 2024 guest lecture

Pairwise Comparisons

$a \succ b \quad b \succ c$

$a \succ c$

requires some assumption on how pref are generated

- Often easier for people to make than hand writing a reward function
- Often easier than providing scalar reward (how much do you like this ad?)

[15] Slides with Red Text Font from Rafailov, Sharma, Mitchell 2024 guest lecture

Fitting the Parameters of a Bradley-Terry Model

- Consider k-armed bandits[18]: K actions $b_1, b_2, \ldots b_k$. No state/context.
- Assume a human makes noisy pairwise comparisons, where the probability she prefers $b_i \succ b_j$ is

$$P(b_i \succ b_j) = \frac{\exp(r(b_i))}{\exp(r(b_i)) + \exp(r(b_j))} = p_{ij} \tag{1}$$

- Assume have N tuples of form (b_i, b_j, μ) where $\mu(1) = 1$ if the human marked $b_i \succ b_j$, $\mu(1) = 0.5$ if the human marked $b_i = b_j$, else 0 if $b_j \succ b_i$
- Maximize likelihood with cross entropy

learn parameters of r_ϕ

$$loss = -\sum_{(b_i, b_j, \mu) \in \mathcal{D}} \mu(1) \log P(b_i \succ b_j) + \mu(2) \log P(b_j \succ b_j) \tag{2}$$

- Use learned reward model, and do PPO with this model
- See prior lecture for notes on doing this over trajectories

(assume dynamics model)

[17] Slides with Red Text Font from Rafailov, Sharma, Mitchell 2024 guest lecture

[18] We will see more on bandits later in the course

RLHF: Learning a **reward model** from human feedback

Feedback comes as **preferences over model samples:** $\mathcal{D} = \{x^i, y_w^i, y_l^i\}$

Prompt

Preferred response

Dispreferred response

Bradley-Terry Model connects rewards to preferences:

Reward assigned to **preferred** and **dispreferred** responses

$$p(y_w \succ y_l \mid x) = \sigma(r(x, y_w) - r(x, y_l))$$

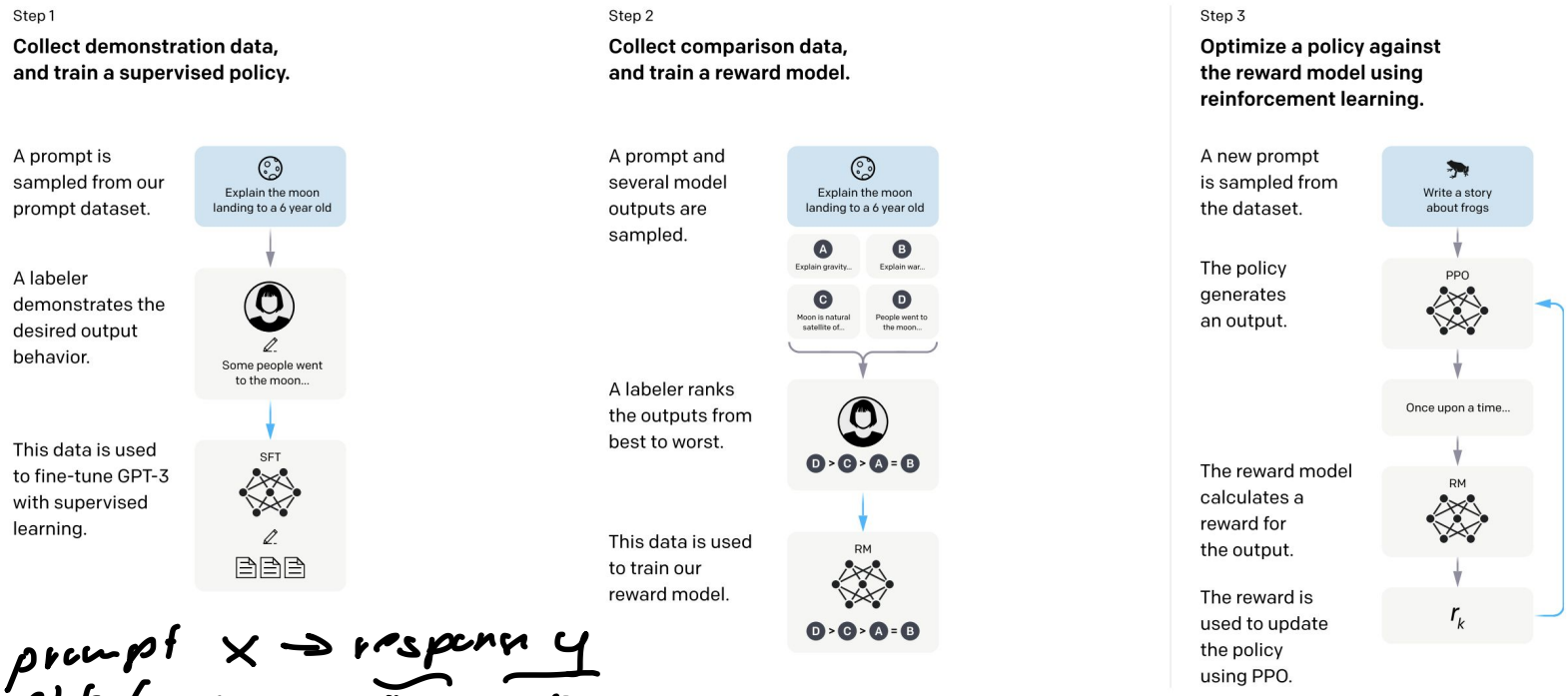
logistic

$$\frac{1}{1+\exp(\cdots)}$$

Train the reward model by **minimizing negative log likelihood:**

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

RLHF: Reinforcement Learning From Human Feedback



prompt x → response y
sbh / context
"action"

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

RLHF: Learning a **policy** that optimizes the **reward**

Now we have a **reward model** r_ϕ that represents* **goodness according to humans**

Now, learn a policy π_θ achieving **high reward** while **staying close** to original model π_{ref}

plugged in estimate of true reward

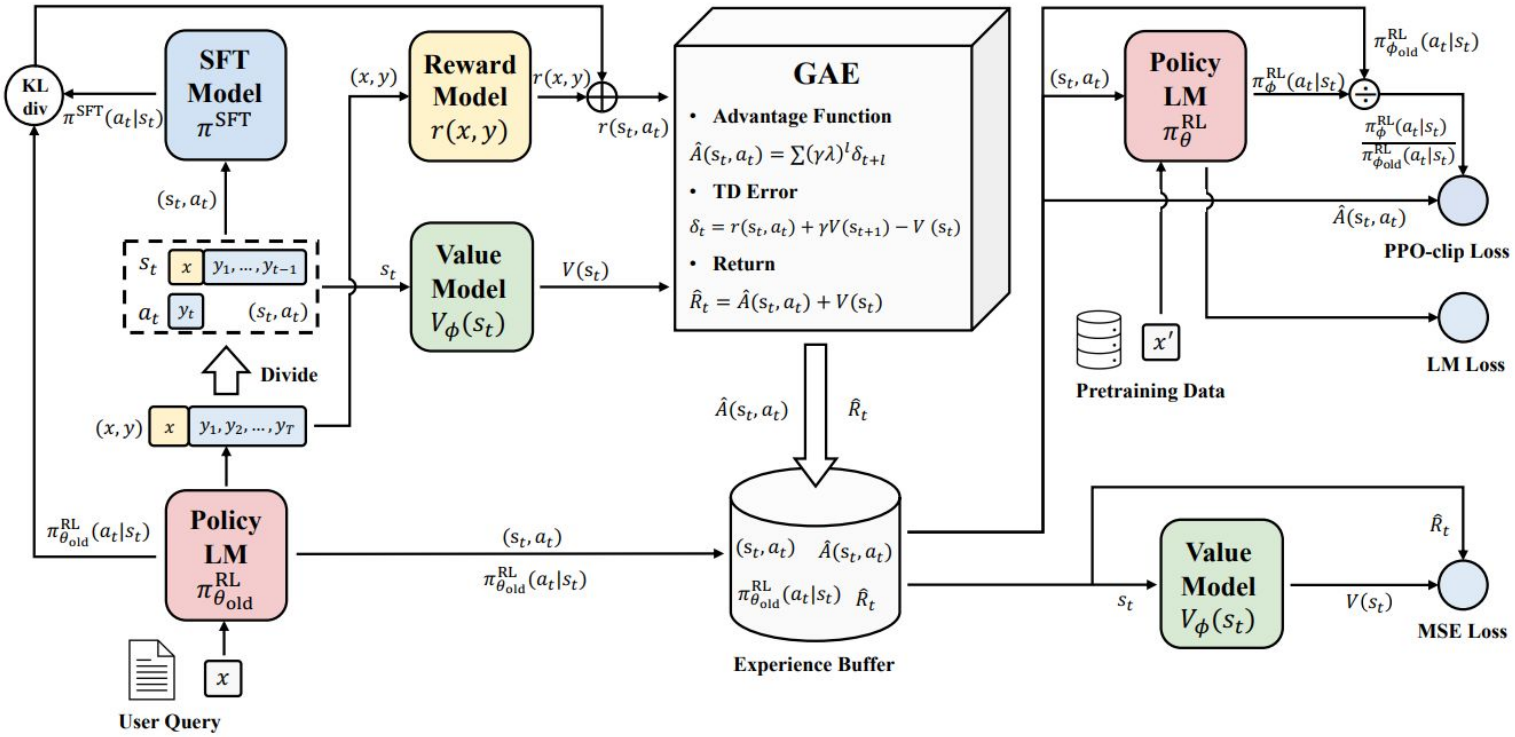
$$\max_{\pi_\theta} \mathbb{E}_{x\sim\mathcal{D}, y\sim\pi_\theta(y|x)} \left[r_\phi(x,y) \right] - \beta \mathbb{D}_{\text{KL}} \left[\pi_\theta(y|x) || \pi_{\text{ref}}(y|x) \right]$$

Sample from policy

Want high reward…

…but keep KL to original model small!

RLHF: Learning a **policy** that optimizes the **reward**



Secrets of RLHF in Large Language Models Part I: PPO, Zheng et.al. 2023

Direct Preference Optimization

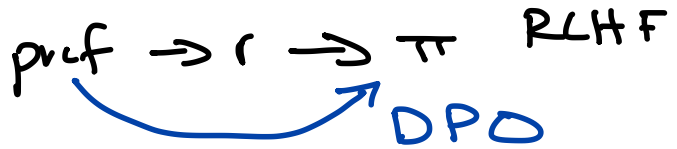
RLHF Objective

(get **high reward**, stay **close** to reference model)

any reward function

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

Direct Preference Optimization



RLHF Objective

(get **high reward**, stay **close** to reference model)

any reward function

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

Closed-form Optimal Policy

(write **optimal policy** as function of **reward function**; from prior work)

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

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

Closed-Form Optimal Policy

$$\max_{\pi} E_{x\sim D,\, y\sim\pi}\left[r(x,y) - \beta D_{KL}\left(\pi(y|x)\,\|\,\pi_{ref}(y|x)\right)\right]$$

$$= \text{''} \quad \left[r(x,y) - \beta \log \pi(y|x)/\pi_{ref}(y|x)\right]$$

$$= \text{''} \quad -\frac{1}{\beta}\left[\log \pi(y|x)/\pi_{ref}(y|x) - \frac{1}{\beta} r(x,y)\right]$$

$$= \min_{\pi} E_{x\sim D,\, y\sim\pi}\left[\log \pi(y|x)/\pi_{ref}(y|x) - \frac{1}{\beta} r(x,y)\right]$$

$$= \text{''} \quad \left[\log \pi(y|x)/\pi_{ref}(y|x) - \log \exp \frac{1}{\beta} r(x,y)\right]$$

$$= \text{''} \quad \left[\log \frac{\pi(y|x)}{\pi_{ref}(y|x)\exp\left(\frac{1}{\beta} r(x,y)\right)}\right]$$

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

$$= \text{''} \quad \left[\log\left[\quad \text{''} \quad \frac{Z(x)}{Z(x)}\right]\right]$$

$$= \min_{\pi} \left[\log \frac{\pi(y|x)}{\frac{1}{Z(x)}\pi_{ref}(y|x)\exp\left(\frac{1}{\beta} r(x,y)\right)} - \log Z(x)\right]$$

[20] Slides with Red Text Font from Rafailov, Sharma, Mitchell 2024 guest lecture[21]

Direct Preference Optimization

RLHF Objective

(get **high reward**, stay **close** to reference model)

any reward function

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

Closed-form Optimal Policy

(write **optimal policy** as function of **reward function**; from prior work)

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

with $Z(x) = \sum_y \pi_{\mathrm{ref}}(y \mid x) \exp\left(\frac{1}{\beta} r(x,y)\right)$ ← Note **intractable sum** over possible responses; can't immediately use this

$$\min_{\pi} \mathbb{E}_{x\sim\mathcal{D}} \mathbb{E}_{y\sim\pi} \log \frac{\pi(y|x)}{\pi^*(y|x)} - \log Z(x)$$

$$\equiv \min_{\pi} \mathbb{E}_{x\sim\mathcal{D}} D_{KL}(\pi(y|x) \| \pi^*(y|x))$$

Direct Preference Optimization

RLHF Objective

(get **high reward**, stay **close** to reference model)

any reward function

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

Closed-form Optimal Policy

(write **optimal policy** as function of **reward function**; from prior work)

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

with $Z(x) = \sum_y \pi_{\mathrm{ref}}(y \mid x)\exp\left(\frac{1}{\beta}r(x,y)\right)$ ← Note **intractable sum** over possible responses; can't immediately use this

Rearrange

(write **any reward function** as function of **optimal policy**)

DPO Derivation

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

solve for $r(x,y)$ in terms of π_r^*:

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

take log

$$\log \frac{\pi_r^*(y|x)}{\pi_{ref}(y|x)} + \log Z(x) = \frac{1}{\beta} r(x,y)$$

$$\beta \log \frac{\pi_r^*(y|x)}{\pi_{ref}(y|x)} + \beta \log Z(x) = r(x,y)$$

added post lecture

Now plug this into preference func & will show we can derive preferences directly in terms of π_{ref} & π^*

[22] Slides with Red Text Font from Rafailov, Sharma, Mitchell 2024 guest lecture

DPO Derivation

$$p^*(y_1 \succ y_2 | x) = \frac{\exp(r^*(x, y_1))}{\exp(r^*(x, y_1)) + \exp(r^*(x, y_2))}$$

$$r^*(x, y) = \beta \log \frac{\pi^*(y|x)}{\pi_{ref}(y|x)} + \beta \log Z(x)$$

$$p^*(y_1 \succ y_2 | x) = \frac{1}{1 + \exp(r^*(x, y_2) - r^*(x, y_1))}$$

$$= \frac{1}{1 + \exp\left(\beta \log \frac{\pi^*(y_2|x)}{\pi_{ref}(y_2|x)} - \beta \log \frac{\pi^*(y_1|x)}{\pi_{ref}(y_1|x)}\right)}$$

[22] Slides with Red Text Font from Rafailov, Sharma, Mitchell 2024 guest lecture

Direct Preference Optimization

RLHF Objective

(get **high reward**, stay **close** to reference model)

any reward function

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

Closed-form Optimal Policy

(write **optimal policy** as function of **reward function**; from prior work)

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

with $Z(x) = \sum_y \pi_{\mathrm{ref}}(y \mid x) \exp\left(\frac{1}{\beta} r(x,y)\right)$ ← Note **intractable sum** over possible responses; can't immediately use this

Rearrange

(write **any reward function** as function of **optimal policy**)

Ratio is **positive** if policy likes response more than reference model, **negative** if policy likes response less than ref. model

$$r(x,y) = \underbrace{\beta \log \frac{\pi^*(y \mid x)}{\pi_{\mathrm{ref}}(y \mid x)} + \beta \log Z(x)}_{\text{some parameterization of a reward function}}$$

Direct Preference Optimization: Putting it together

A loss function on reward functions

+

A transformation between reward functions and policies

=

A loss function on policies

Derived from the Bradley-Terry model of human preferences:

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

Direct Preference Optimization: Putting it together

A loss function on reward functions

+

A transformation between reward functions and policies

=

A loss function on policies

Derived from the Bradley-Terry model of human preferences:

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

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

Direct Preference Optimization: Putting it together

A loss function on reward functions

+

A transformation between reward functions and policies

=

A loss function on policies

Derived from the Bradley-Terry model of human preferences:

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

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

Reward of **preferred** response

Reward of **dispreferred** response

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

Direct Preference Optimization: Putting it together

A loss function on reward functions

+

A transformation between reward functions and policies

=

A loss function on policies

Derived from the Bradley-Terry model of human preferences:

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

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

When substituting, the **log Z term cancels**, because the loss only cares about **difference** in rewards

Reward of **preferred** response

Reward of **dispreferred** response

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

Direct Preference Optimization: Putting it together

A loss function on reward functions

+

A transformation between reward functions and policies

Derived from the Bradley-Terry model of human preferences:

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

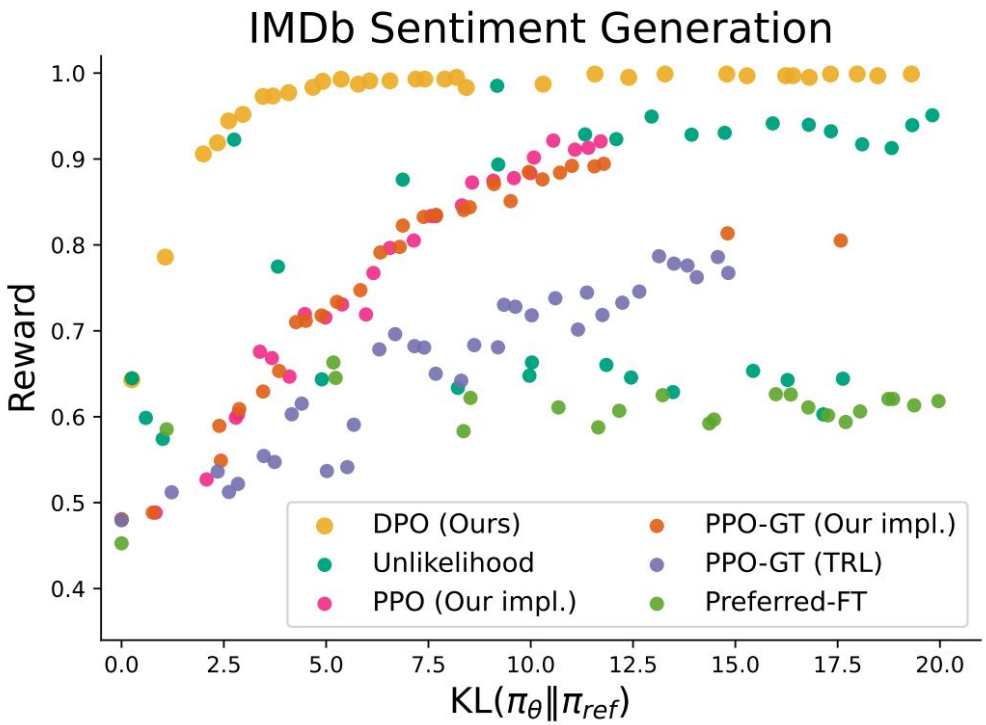
$$r_{\pi_\theta}(x, y) = \beta \log \frac{\pi_\theta(y \mid x)}{\pi_{\text{ref}}(y \mid x)} + \beta \log Z(x)$$

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

Reward of **preferred** response — Reward of **dispreferred** response

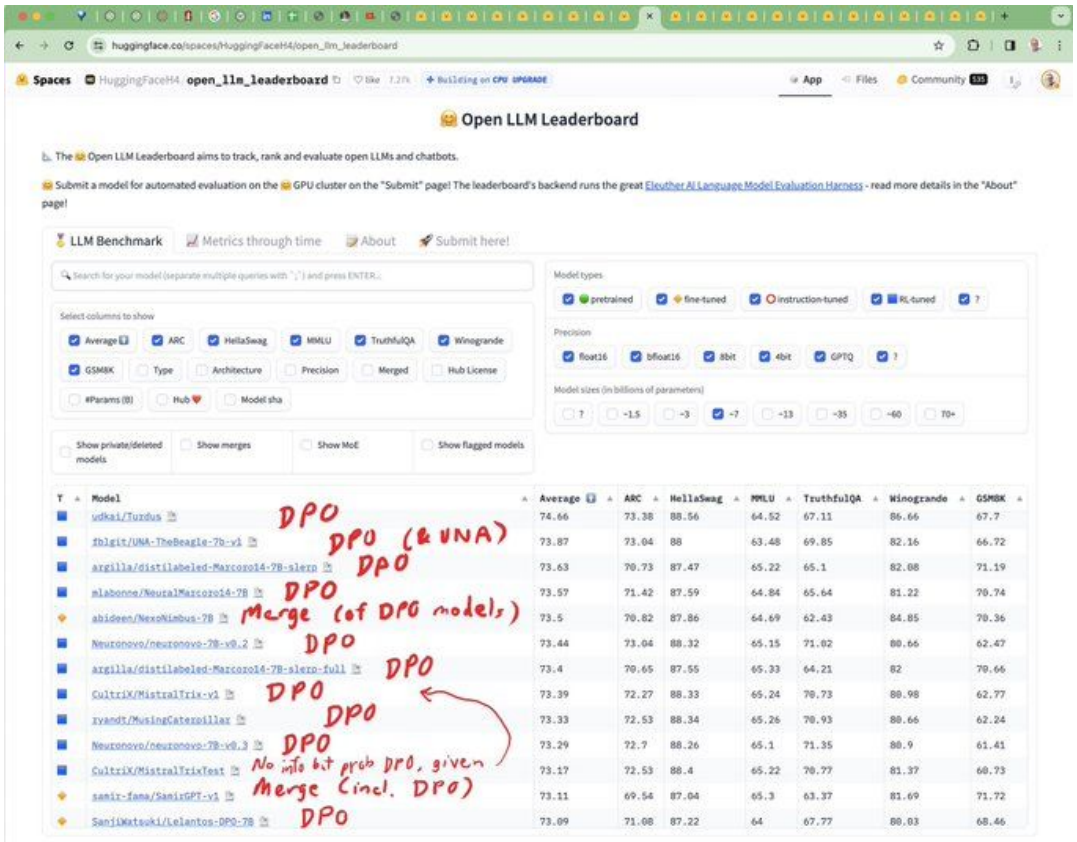
Results

How Efficiently does DPO Trade off Reward & KL?



1. Generate positive IMDB reviews from GPT2-XL
2. Use pre-trained sentiment classifier as Gold RM
3. Create preferences based on Gold RM
4. Optimize with PPO and DPO

Models Trained With DPO



Large-Scale DPO Training

Large-Scale DPO Training

Mistral

4 Instruction Fine-tuning

We train Mixtral – Instruct using supervised fine-tuning (SFT) on an instruction dataset followed by Direct Preference Optimization (DPO) [25] on a paired feedback dataset. Mixtral – Instruct reaches a score of 8.30 on MT-Bench [33] (see Table 2), making it the best open-weights model as of December 2023. Independent human evaluation conducted by LMSys is reported in Figure 6[3] and shows that Mixtral – Instruct outperforms GPT-3.5-Turbo, Gemini Pro, Claude-2.1, and Llama 2 70B chat.

Model	Arena Elo rating	MT-bench (score)	License
GPT-4-Turbo	1243	9.32	Proprietary
GPT-4-0314	1192	8.96	Proprietary
GPT-4-0613	1158	9.18	Proprietary
Claude-1	1149	7.9	Proprietary
Claude-2.0	1131	8.06	Proprietary
Mixtral-8x7b-Instruct-v0.1	1121	8.3	Apache 2.0
Claude-2.1	1117	8.18	Proprietary
GPT-3.5-Turbo-0613	1117	8.39	Proprietary
Gemini Pro	1111		Proprietary
Claude-Instant-1	1110	7.85	Proprietary
Tulu-2-DPO-70B	1110	7.89	AI2 ImpACT Low-risk
Yi-34B-Chat	1110		Yi License
GPT-3.5-Turbo-0314	1105	7.94	Proprietary
Llama-2-70b-chat	1077	6.86	Llama 2 Community

Figure 6: LMSys Leaderboard. (Screenshot from Dec 22, 2023) Mixtral 8x7B Instruct v0.1 achieves an Arena Elo rating of 1121 outperforming Claude-2.1 (1117), all versions of GPT-3.5-Turbo (1117 best), Gemini Pro (1111), and Llama-2-70b-chat (1077). Mixtral is currently the best open-weights model by a large margin.

Large-Scale DPO Training

Mistral

4 Instruction Fine-tuning

We train Mixtral – Instruct using supervised fine-tuning (SFT) on an instruction dataset followed by Direct Preference Optimization (DPO) [25] on a paired feedback dataset. Mixtral – Instruct reaches a score of 8.30 on MT-Bench [33] (see Table 2), making it the best open-weights model as of December 2023. Independent human evaluation conducted by LMSys is reported in Figure 6[3] and shows that Mixtral – Instruct outperforms GPT-3.5-Turbo, Gemini Pro, Claude-2.1, and Llama 2 70B chat.

Model	Arena Elo rating	MT-bench (score)	License
GPT-4-Turbo	1243	9.32	Proprietary
GPT-4-0314	1192	8.96	Proprietary
GPT-4-0613	1158	9.18	Proprietary
Claude-1	1149	7.9	Proprietary
Claude-2.0	1131	8.06	Proprietary
Mixtral-8x7b-Instruct-v0.1	1121	8.3	Apache 2.0
Claude-2.1	1117	8.18	Proprietary
GPT-3.5-Turbo-0613	1117	8.39	Proprietary
Gemini Pro	1111		Proprietary
Claude-Instant-1	1110	7.85	Proprietary
Tulu-2-DPO-70B	1110	7.89	AI2 ImpACT Low-risk
Yi-34B-Chat	1110		Yi License
GPT-3.5-Turbo-0314	1105	7.94	Proprietary
Llama-2-70b-chat	1077	6.86	Llama 2 Community

Figure 6: LMSys Leaderboard. (Screenshot from Dec 22, 2023) Mixtral 8x7B Instruct v0.1 achieves an Arena Elo rating of 1121 outperforming Claude-2.1 (1117), all versions of GPT-3.5-Turbo (1117 best), Gemini Pro (1111), and Llama-2-70b-chat (1077). Mixtral is currently the best open-weights model by a large margin.

LLaMa3

Instruction fine-tuning

To fully unlock the potential of our pretrained models in chat use cases, we innovated on our approach to instruction-tuning as well. Our approach to post-training is a combination of supervised fine-tuning (SFT), rejection sampling, proximal policy optimization (PPO), and direct preference optimization (DPO). The quality of the prompts that are used in SFT and the preference rankings that are used in PPO and DPO has an outsized influence on the performance of aligned models. Some of our biggest improvements in model quality came from carefully curating this data and performing multiple rounds of quality assurance on annotations provided by human annotators.

Learning from preference rankings via PPO and DPO also greatly improved the performance of Llama 3 on reasoning and coding tasks. We found that if you ask a model a reasoning question that it struggles to answer, the model will sometimes produce the right reasoning trace: The model knows how to produce the right answer, but it does not know how to select it. Training on preference rankings enables the model to learn how to select it.

lecture finished here

CPL

[24] Slides with Red Text Font from Rafailov, Sharma, Mitchell 2024 guest lecture

[26] Slides with Red Text Font from Rafailov, Sharma, Mitchell 2024 guest lecture

Today

- RLHF for LLM
- **Direct Preference Optimization**

[28] Slides with Red Text Font from Rafailov, Sharma, Mitchell 2024 guest lecture

Class Structure

- Last time: Imitation Learning (Max Entropy IRL) and RLHF
- This time: RLHF and Direct Preference Optimization and guest lecture
- Next time: Midterm

[30] Slides with Red Text Font from Rafailov, Sharma, Mitchell 2024 guest lecture