RLHF without RL

Pustovalov Iurii 212

Plan

- What is RLHF for
- How RLHF works
- Problems
- What is DPO
- What is CoH

What is RLHF for?

Basically, usually we have a big pretrained language model, and we want to tune it to produce more human-like answers

Answers should be safe, coherent and helpful

How does RLHF work?

- SFT(Supervised Fine-Tuning)
- Reward Modeling Phase
- RL Fine-Tuning Phase

SFT

Simply finetune the model to well-known tasks on good datasets

Get model $\boldsymbol{\pi}_{\text{sft}}(x)$

Reward Modeling Phase

- $y1, y2 \sim \boldsymbol{\pi}_{\text{sft}}(x)$ for every x
- People determine if y1 better than y2

  - actually a sigmoid of difference

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

- Train model to predict r*

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

Intuition is clear: better answer - bigger reward

Reward Modeling Phase

$r_{\Phi}(x)$ - $\pi_{sft}(x)$ with linear head

Some weights are frozen to spend less resources

After all, normalize reward to nullify the expected value

RL Fine-Tuning Phase

Now we optimize the following:

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

where $\boldsymbol{\pi}_{\boldsymbol{\theta}}$ is initialized from $\boldsymbol{\pi}_{\text{sft}}$ (works better)

First term - optimize reward, second - stay close to original model

solve with RL(PPO)

Problems

- Complex training pipeline
- Need to train multiple LM's
- Need to sample from LM - costly

DPO(scary formulas)

Only RL Fine-Tuning Phase is changed

With some algebra, optimal policy is

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

Z(x) - just to normalize probabilities

But, can’t estimate Z(x) => can’t sample

DPO(scary formulas)

By using some more algebra, we get

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

Now, substitute this thing into $p^*(y_1 \succ y_2 \mid x) = \frac{\exp\left(r^*(x, y_1)\right)}{\exp\left(r^*(x, y_1)\right) + \exp\left(r^*(x, y_2)\right)}$., get

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

DPO(scary formulas)

Finally, we can maximize log-likelihood of our human dataset

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

Now model is very differentiable, can solve with DL, very cool

DPO(scary formulas)

Try to understand loss through its' gradient(r_θ is from 2 slides ago)

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

Weighing with beta is actually important, otherwise model degenerates

DPO(scary formulas)

Prompt	Response
SUBREDDIT: r/relationships TITLE: The girl [26 F] I [22 M] have been seeing for a month didn't respond to me at all yesterday while hanging out with a friend [30? M]. POST: She gets terrible service while at her house, but I texted her 3 times yesterday, 4-5 hours apart. She didn't call me until early this morning and left a voicemail that she was busy all day with a friend who showed up out of the blue. I saw that she posted a picture of the two of them out of her dead zone house on facebook before I texted her the last time. I don't mind that she hangs out with friends, and I know it's pretty early [. . .] TL;DR:	girl whenUB when when when when whenUB when when when whenUB
SUBREDDIT: r/tifu TITLE: TIFU by accidently kicking an old woman POST: So this didn't happen today but actually about a year or two ago. I was at my granddads funeral so of course it was all very sad and full of lots of crying old people. After the ceremony everyone walks outside the building and onto the other side of the small road the hearses drive down. Now the road is important because obviously if there's a road, there's a curb onto the sidewalk, so most of us are on the other side of the road, besides a few older people walking a lot slower. As one of the old woman goes to walk up the curb [. . .] TL;DR:	when an old woman was tripping the when

DPO(scary formulas)

Final pipeline is:

- Do everything the same as in RLHF without last stage
- initialize new policy with fine-tuned model and train it on the new loss on human-labeled dataset
- Actually, can use other good data, not only from this model

Experiments

- Controlled sentiment generation - IMDB - classifier
- Summarization - Reddit - GPT4
- Single-turn dialogue - Anthropic-HH - GPT4

Experiments

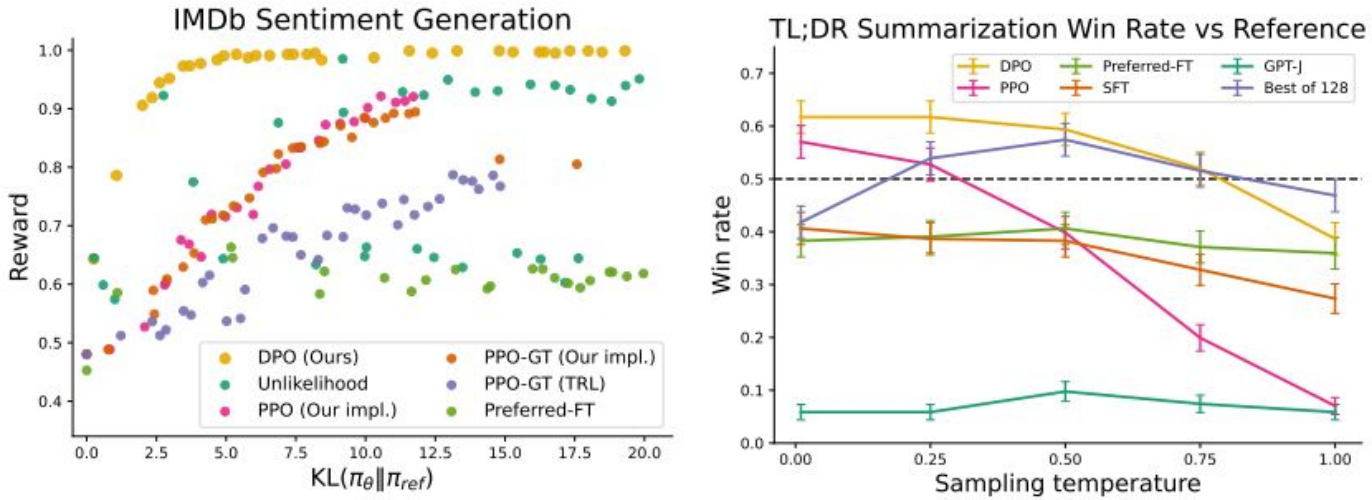

Figure 2: **Left.** The frontier of expected reward vs KL to the reference policy. DPO provides the highest expected reward for all KL values, demonstrating the quality of the optimization. **Right.** TL;DR summarization win rates vs. human-written summaries, using GPT-4 as evaluator. DPO exceeds PPO's best-case performance on summarization, while being more robust to changes in the sampling temperature.

Experiments

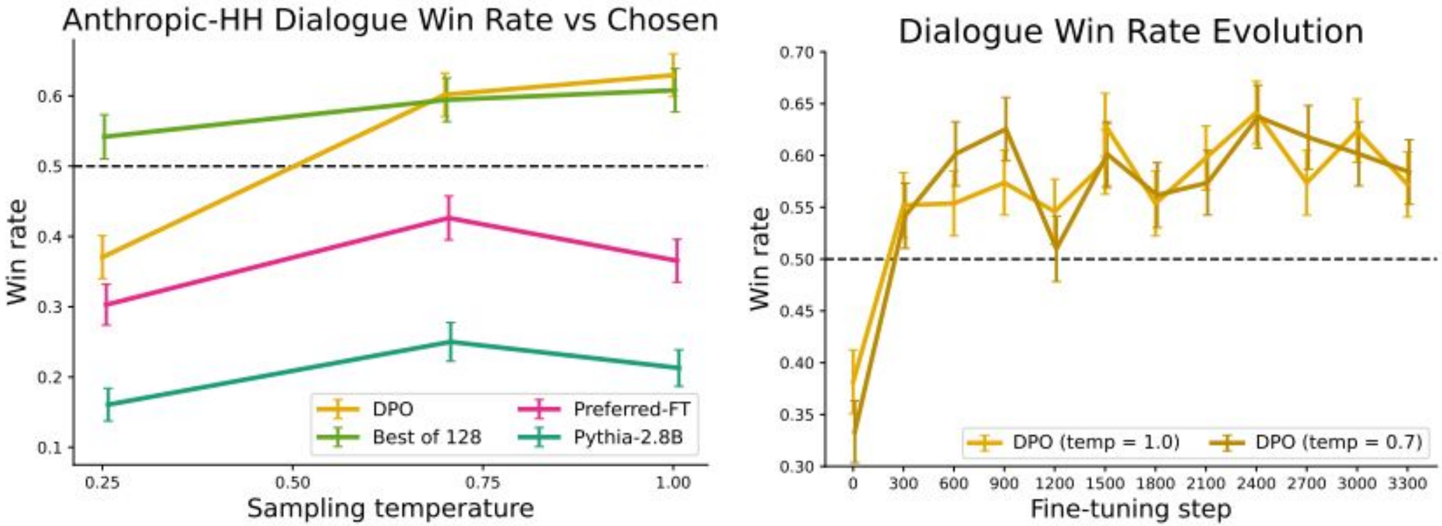

Figure 3: **Left.** Win rates computed by GPT-4 for Anthropic-HH one-step dialogue; DPO is the only method that improves over chosen summaries in the Anthropic-HH test set. **Right.** Win rates for different sampling temperatures over the course of training. DPO's improvement over the dataset labels is fairly stable over the course of training for different sampling temperatures.

Experiments

	DPO	SFT	PPO-1
N respondents	272	122	199
GPT-4 (S) win %	47	27	13
GPT-4 (C) win %	54	32	12
Human win %	58	43	17
GPT-4 (S)-H agree	70	77	86
GPT-4 (C)-H agree	67	79	85
H-H agree	65	-	87

Table 2: Comparing human and GPT-4 win rates and per-judgment agreement on TL;DR summarization samples. **Humans agree with GPT-4 about as much as they agree with each other.** Each experiment compares a summary from the stated method with a summary from PPO with temperature 0.

GPT is good, because people agree with it more than with each other

CoH(Chain of Hindsight)

Key idea - let model see other answers during training(with feedback)

Feedback of any form can be used, but authors stick to templated, based on rating

Natural language feedback examples

A good summary: {positive}, a worse summary: {negative}
You are a helpful assistant: {positive}, you are an unhelpful assistant: {negative}
A bad answer is {negative}, a good answer is {positive}

CoH

- Loss is not applied on feedback tokens, because it works worse

Algorithm 1 Aligning language models from feedback with Chain of Hindsight.

Required: Pretrained Language Model M, Human Feedback Dataset D
Required: Maximum training iterations n
Initialize
for $iter = 1$ **to** n **do**
  Randomly sample a minibatch of model outputs and their associated ratings from dataset D.
  Construct training sequences by combining sampled model outputs with feedback based on ratings.
  Instruct finetune model M on the training sequences.
end for

- Outputs are sampled before fine-tune cycle
- Mask 0%-5% previous tokens, so model doesn't remember answers
- Add log-likelihood on pretrain dataset

Experiments

- Summarization
- Single-turn dialogue
- 75 experts proficient in English

Experiments

Table 2: Pairwise human evaluation on dialogue task.

	Human evaluation win rate (%)			
	Base	Tie	CoH	Δ
Helpful	15.8	34.8	49.4	33.6
Harmless	14.5	35.9	49.6	35.1
Average	15.2	35.3	**49.5**	**34.4**
	SFT	Tie	CoH	Δ
Helpful	19.6	45.7	34.7	15.1
Harmless	18.6	37.4	44.0	25.4
Average	19.1	41.5	**39.4**	**20.3**
	C-SFT	Tie	CoH	Δ
Helpful	21.8	46.9	31.3	9.5
Harmless	22.4	35.2	42.4	20.0
Average	22.1	41.0	**36.8**	**14.7**
	SFT-U	Tie	CoH	Δ
Helpful	13.4	31.3	55.3	41.9
Harmless	14.5	28.7	56.8	42.3
Average	13.9	30.0	**56.0**	**42.1**
	RLHF	Tie	CoH	Δ
Helpful	25.8	40.8	33.4	7.6
Harmless	20.9	38.8	40.3	19.4
Average	23.4	39.8	**36.9**	**13.5**

Table 1: Pairwise human evaluation on summarization task.

	Human evaluation win rate (%)			
	Base	Tie	CoH	Δ
Accuracy	24.5	26.8	48.7	24.2
Coherence	15.6	18.5	65.9	50.3
Coverage	19.6	22.4	58.0	38.4
Average	19.9	22.6	**57.5**	**37.6**
	SFT	Tie	CoH	Δ
Accuracy	25.5	32.6	41.9	16.4
Coherence	30.5	25.6	43.9	13.4
Coverage	28.5	25.4	46.1	17.6
Average	28.2	27.9	**44.0**	**15.8**
	C-SFT	Tie	CoH	Δ
Accuracy	26.7	34.9	38.4	11.7
Coherence	32.5	22.9	44.6	12.1
Coverage	29.5	26.7	43.8	14.3
Average	29.6	28.2	**42.3**	**12.7**
	SFT-U	Tie	CoH	Δ
Accuracy	18.7	17.9	63.4	44.7
Coherence	21.8	15.8	62.4	40.6
Coverage	23.6	17.2	59.2	35.6
Average	21.4	17.0	**61.7**	**40.3**
	RLHF	Tie	CoH	Δ
Accuracy	31.8	29.5	38.7	6.9
Coherence	31.6	20.5	47.9	16.4
Coverage	28.9	21.9	49.2	20.3
Average	30.8	24.0	**45.3**	**14.5**

Experiments

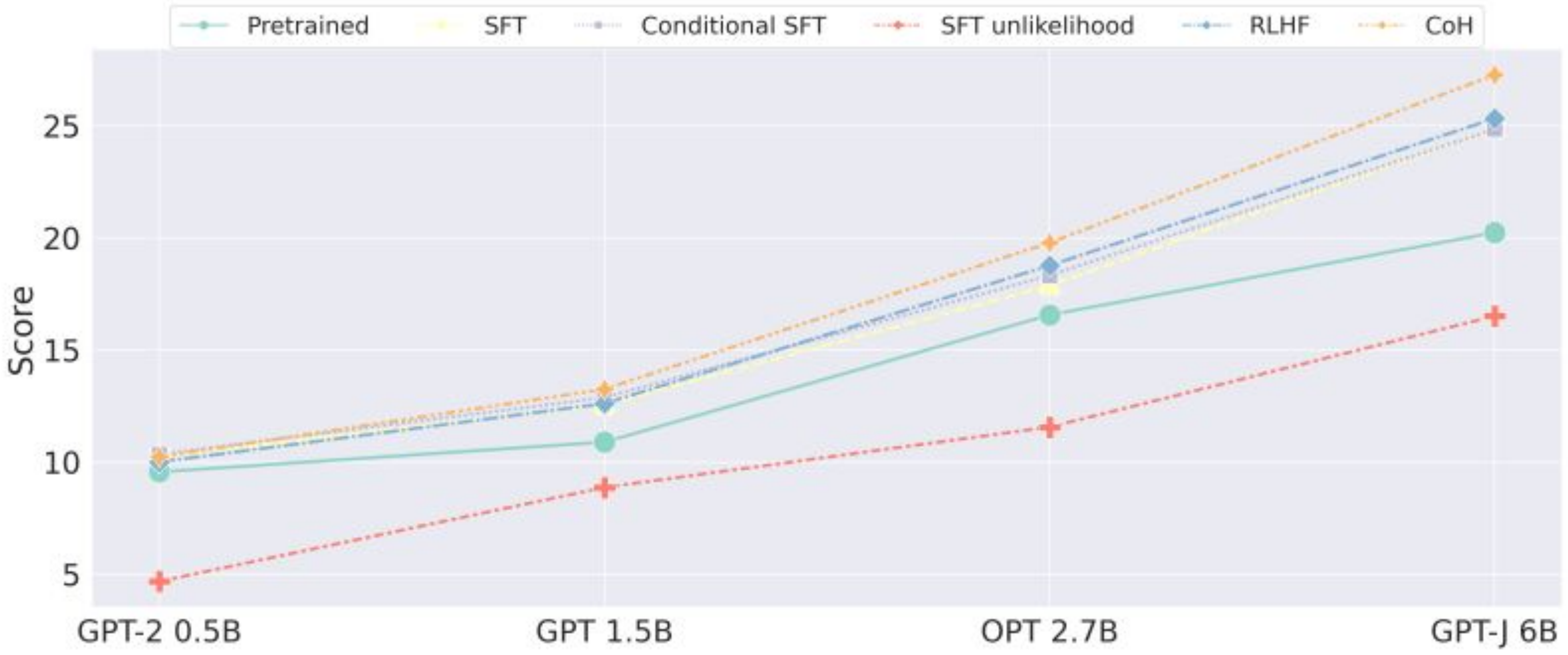

Figure 5: **Model scaling trend**. Comparing CoH with RLHF and SFT baselines on summarization benchmark with different model sizes. CoH outperforms RLHF, showing strong scaling capabilities.