LLM Alignment Week 4 Progression

Yuanqi Zhao

University of Oxford

12.Aug.2024

- **DPO review and the code implementation.**
- The whole procedure for a DPO Project.

DPO Review

Take TRL library DPO_trainer as an example;
The Objective Function to be optimized:

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

where:

y_w : The preferred response.

y_l : The rejected response.

x : The prompt.

π_θ : The Policy waited to be tuned.

π_{ref} : The Reference Policy (Frozen LLM).

β : Hyper-parameter.

σ : Sigmoid.

From Bradley–Terry model to Reward Modelling

The Bradley-Terry (BT) model mentioned is used to estimate the human preference distribution $P(y_w > y_l)$:

$$P(y_w > y_l) = \frac{e^{r\phi(x,y_w)}}{e^{r\phi(x,y_w)} + e^{r\phi(x,y_l)}}$$

By the property of sigmoid function:

$$\frac{e^A}{e^A + e^B} \to \sigma(A - B)$$

And meanwhile apply the MLE technique, we get the reward loss function:

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

DPO Derivation: From PPO to DPO

Recall the PPO objective function:

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

By KL divergence formula:

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

Since $Max(f(x)) = Max(\beta * f(x))$, we multiply a $\frac{1}{\beta}$

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

which is equivalent as:

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

DPO Derivation: From PPO to DPO

By a = log(exp(a)) :

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

Followed by a simple simplification:

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

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

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

DPO Derivation: From PPO to DPO

We can define π^* as a distribution, taking Z(x) as a normalizing constant.

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

Note, π^* is a function of x, and it's independent of π. Then we can rewrite the objective function as:

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

The inner Expectation is in the form of KL-Divergency:

$$\min_{\pi} \mathbb{E}_{x \sim \mathcal{D}} \left[\mathbb{D}_{\text{KL}}(\pi(y|x) \,||\, \pi^*(y|x)) - \log Z(x)\right]$$

DPO Derivation: From PPO to DPO

Now, since $Z(x)$ does not depend on π, the minimum is achieved by the policy that minimizes the first KL term. Gibbs' inequality tells us that the KL-divergence is minimized at 0 if and only if the two distributions are identical. Hence we have the optimal solution:

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

Followed by a simple simplification:

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

DPO Derivation: From PPO to DPO

Recall the Bradley-Terry Model:

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

Using the same argument as previous:

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

which is :

$$\sigma\left(\beta \log \frac{\pi^*(y_1 \mid x)}{\pi_{\text{ref}}(y_1 \mid x)} - \beta \log \frac{\pi^*(y_2 \mid x)}{\pi_{\text{ref}}(y_2 \mid x)}\right)$$

DPO Derivation: From PPO to DPO

Using MLE on $p^*(y_1 \succ y_2 \mid x)$, we finally get the loss function:

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

Before consider the derivative, recall the chain rule at first:

$$\frac{d}{dx} f(g(h(x))) = f'(g(h(x))) \cdot g'(h(x)) \cdot h'(x)$$

Meanwhile, consider the sigmoid property:

$$\sigma^{'}(x) = \sigma(x) * (1 - \sigma(x)) \Leftrightarrow (1 - \sigma(x)) = \frac{\sigma^{'}(x)}{\sigma(x)}$$

DPO Derivation: From PPO to DPO

Therefore, the derivative is:

$$\nabla_\theta \mathcal{L}_{\text{DPO}}(\pi_\theta; \pi_{\text{ref}}) =$$

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

The corresponding code

To get $log(\frac{\pi_\theta(y_l|x)}{\pi_{\text{ref}}(y_l|x)})$, $log(\frac{\pi_\theta(y_w|x)}{\pi_{\text{ref}}(y_w|x)})$:

```
chosen_logratios = policy_chosen_logps.to(self.accelerator.device) - (
    not self.reference_free
) * reference_chosen_logps.to(self.accelerator.device)

rejected_logratios = policy_rejected_logps.to(self.accelerator.device) - (
    not self.reference_free
) * reference_rejected_logps.to(self.accelerator.device)
```

Figure 1

The corresponding code

To get $log(\frac{\pi_\theta(y_l|x)}{\pi_{\text{ref}}(y_l|x)}) - log(\frac{\pi_\theta(y_w|x)}{\pi_{\text{ref}}(y_w|x)})$:

```python
else:
    pi_logratios = policy_chosen_logps - policy_rejected_logps
    if self.reference_free:
        ref_logratios = torch.tensor([0], dtype=pi_logratios.dtype, device=pi_logratios.device)
    else:
        ref_logratios = reference_chosen_logps - reference_rejected_logps

    pi_logratios = pi_logratios.to(self.accelerator.device)
    ref_logratios = ref_logratios.to(self.accelerator.device)
    logits = pi_logratios - ref_logratios
```

Figure 2

The corresponding code

Since PPO using the LogSigmoid (we dont consider other variants like IPO, SSO)

```
if self.loss_type == "sigmoid":
    losses = (
        -F.logsigmoid(self.beta * logits) * (1 - self.label_smoothing)
        - F.logsigmoid(-self.beta * logits) * self.label_smoothing
    )
```

Figure 3: Enter Caption

Reference

- HuggingFace TRL library: https://huggingface.co/docs/trl/en/index