TGDPO: Harnessing Token-Level Reward Guidance for Enhancing Direct Preference Optimization

Mingkang Zhu[1], Xi Chen[2], Zhongdao Wang[3], Bei Yu[1], Hengshuang Zhao[2], Jiaya Jia[4, 5]

[1]The Chinese University of Hong Kong, [2]The University of Hong Kong, [3]Huawei, [4]SmartMore, [5]The Hong Kong University of Science and Technology

ICML International Conference On Machine Learning

Motivation

- Fine-grained or token-level rewards have shown effectiveness in RLHF using PPO.
- Being a popular and efficient alternative in RLHF, DPO bypasses reward modeling. However, it cannot incorporate the fine-grained guidance like PPO.

How to harness dense reward guidance for DPO? 🤔

TGDPO

$$\mathcal{L}_{\text{TGDPO}}(\pi_\theta) = -\mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}}\left[\log\sigma\left(\sum_{t=0}^{T_w-1}\beta f_w(\hat{r}([x,y_w^{<t}],y_w^t))\log\frac{\pi_\theta(y_w^t|[x,y_w^{<t}])}{\pi_{\text{ref}}(y_w^t|[x,y_w^{<t}])} - \sum_{t=0}^{T_l-1}\beta f_l(\hat{r}([x,y_l^{<t}],y_l^t))\log\frac{\pi_\theta(y_l^t|[x,y_l^{<t}])}{\pi_{\text{ref}}(y_l^t|[x,y_l^{<t}])}\right)\right]$$

where $\hat{r}(s_t,a_t)$ is a token-level reward function, $f_w(\cdot)$ and $f_l(\cdot)$ are positive univariate functions

- TGDPO leverages $f\big(\hat{r}(s_t,a_t)\big)$ to shape the optimization of the policy on the tokens of chosen and rejected responses.
- With an appropriate choice of $f(\cdot)$, this framework can recover several known direct preference optimization methods.
- For example, if $f_w \equiv f_l \equiv 1$, it recovers the objective function of DPO.

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

Practical Implementation

- Set $\hat{r}([x,y^{<t}],y^t) = \beta log\frac{\pi_{\hat{\theta}}(y^t|[x,y^{<t}])}{\pi_{ref}(y^t|[x,y^{<t}])}$ be DPO's induced token-level reward
- Set $f_w\big(\hat{r}([x,y_w^{<t}],y_w^t)\big) = 1+\alpha\hat{r}([x,y_w^{<t}],y_w^t)$
- Set $f_l\left(\hat{r}([x,y_l^{<t}],y_l^t)\right) = 1-\alpha\hat{r}([x,y_l^{<t}],y_l^t)$

Motivation: Each token would exhibit varying degrees of deviation from the reference policy based on their respective rewards.

For example, preferred tokens within preferred responses receive higher weights, while dispreferred tokens within preferred responses receive lower weights.

Theoretical Results

Theorem. The preference function $\Pr(y_w \succ y_l|x) = \sigma\big(\varphi(\pi_\theta,f,\hat{r};x,y_w,y_l) + \delta(f,\hat{r};x,y_w,y_l)\big)$ has the same maxima and the same ascent directions as the function $\sigma(\varphi(\pi_\theta,f,\hat{r};x,y_w,y_l))$, where

$$\varphi(\pi_\theta,f,\hat{r};x,y_w,y_l) = \sum_{t=0}^{T_w-1}\beta f_w(\hat{r}([x,y_w^{<t}],y_w^t))log\frac{\pi_{\hat{\theta}}(y_w^t|[x,y_w^{<t}])}{\pi_{ref}(y_w^t|[x,y_w^{<t}])} - \sum_{t=0}^{T_l-1}\beta f_l\left(\hat{r}([x,y_l^{<t}],y_l^t)\right)log\frac{\pi_{\hat{\theta}}(y_l^t|[x,y_l^{<t}])}{\pi_{ref}(y_l^t|[x,y_l^{<t}])}$$

$$\delta(f,\hat{r};x,y_w,y_l) = \sum_{t=0}^{T_w-1}\beta f_w(\hat{r}([x,y_w^{<t}],y_w^t))logZ([x,y_w^{<t}]) - \sum_{t=0}^{T_l-1}\beta f_l\left(\hat{r}([x,y_l^{<t}],y_l^t)\right)logZ([x,y_l^{<t}])$$

Moreover, for two policies $\pi_{\theta 1}$ and $\pi_{\theta 2}$,

$\sigma\big(\varphi(\pi_{\theta 1},f,\hat{r};x,y_w,y_l)+\delta(f,\hat{r};x,y_w,y_l)\big) > \sigma\big(\varphi(\pi_{\theta 2},f,\hat{r};x,y_w,y_l)+\delta(f,\hat{r};x,y_w,y_l)\big)$

If and only if

$\sigma\big(\varphi(\pi_{\theta 1},f,\hat{r};x,y_w,y_l)\big) > \sigma\big(\varphi(\pi_{\theta 2},f,\hat{r};x,y_w,y_l)\big)$.

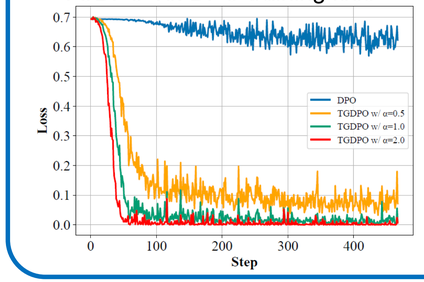
Experiment Results

- Alpaca Eval 2, Arena Hard, MT-Bench

Method	Llama3-8B-Instruct PairRM				Llama3-8B-Instruct ArmoRM			
	AlpacaEval 2	Arena-Hard	MT-Bench		AlpacaEval 2	Arena-Hard	MT-Bench	
	Win Rate (%)	Win Rate (%)	Score	Win Rate(%)	Win Rate (%)	Win Rate (%)	Score	Win Rate(%)
SFT	30.6	21.4	7.9	27.5	30.6	21.4	7.9	27.5
DPO	41.7	30.4	**8.0**	37.5	40.8	36.2	**8.2**	**46.3**
SimPO	39.8	28.7	7.8	32.5	37.0	28.1	7.8	42.5
TGDPO	**43.9**	**34.3**	**8.0**	**41.9**	**42.5**	**40.5**	7.9	45.0

Method	Llama3.2-3B-Instruct ArmoRM				Gemma2-2B-it ArmoRM			
	AlpacaEval 2	Arena-Hard	MT-Bench		AlpacaEval 2	Arena-Hard	MT-Bench	
	Win Rate (%)	Win Rate (%)	Score	Win Rate (%)	Win Rate (%)	Win Rate (%)	Score	Win Rate (%)
SFT	23.8	17.1	7.0	16.3	32.8	20.1	7.9	37.5
DPO	29.6	23.2	7.9	29.4	40.8	26.4	8.0	43.1
SimPO	26.2	22.6	7.4	15.7	34.8	21.1	7.8	40.0
TGDPO	**35.8**	**25.4**	**8.1**	**36.9**	**43.0**	**30.7**	**8.1**	**46.9**

- TGDPO vs. DPO training loss



Empirically, TGDPO effectively minimizes training loss while achieving strong performance. In contrast, the optimal hyperparameters for DPO barely reduce its training loss.

Take Aways

- TGDPO introduces a framework for incorporating token-level guidance into preference optimization.
- Flexible choices of $f(\cdot)$ and token-level reward function $\hat{r}(s_t,a_t)$ can be used for versatile guidance.
- We derive a new theoretical result for eliminating the partition function.