f-PO: Generalizing Preference Optimization with f-divergence Minimization

Jiaqi Han[*], **Mingjian Jiang**[*], **Yuxuan Song, Stefano Ermon, Minkai Xu**[*†]

[*]Equal contribution [†]Corresponding author
{jiaqihan,jiangm,minkai}@cs.stanford.edu
Computer Science, Stanford University

Abstract

Preference optimization has made significant progress recently, with numerous methods developed to align language models with human preferences. This paper introduces f-divergence Preference Optimization (f-PO), a novel framework that generalizes and extends existing approaches. f-PO minimizes f-divergences between the optimized policy and the optimal policy, encompassing a broad family of alignment methods using various divergences. Our approach unifies previous algorithms like DPO and EXO, while offering new variants through different choices of f-divergences. We provide theoretical analysis of f-PO's properties and conduct extensive experiments on state-of-the-art language models using benchmark datasets. Results demonstrate f-PO's effectiveness across various tasks, achieving superior performance compared to existing methods on popular benchmarks such as AlpacaEval 2, Arena-Hard, MT-Bench, and Open LLM Leaderboard v2. Additionally, we present ablation studies exploring the impact of different f-divergences, offering insights into the trade-offs between regularization and performance in offline preference optimization. Our work contributes both practical algorithms and theoretical understanding to the field of language model alignment. Code is available at https://github.com/MinkaiXu/fPO.

1 Introduction

Despite the impressive emerging ability of modern large language models (LLMs) by weak-supervised learning, further aligning these models with human feedback is still crucial (Leike et al., 2018), ensuring the LLMs are more helpful, honest (Askell et al., 2021), harmless (Bai et al., 2022), faithful (Ji et al., 2023), and unbiased (Bender et al., 2021). Reinforcement learning from human feedback (RLHF) (Christiano et al., 2017; Stiennon et al., 2020; Ouyang et al., 2022) is the canonical paradigm for fine-tuning LLMs towards effective alignment. It typically consists of several separate procedures, including training a reward model to capture the human values with well-labeled preference datasets, and optimizing the LLM as the policy model to maximize the reward. While these approaches have achieved remarkable results, they present notable optimization challenges due to the multi-stage process.

Lately, to mitigate this training instability and complexity, simpler offline methods such as Direct Preference Optimization (DPO) (Rafailov et al., 2023) have been attracting increasing attention (Azar et al., 2023; Zhao et al., 2023). Instead of a multi-stage RL pipeline, these methods propose to directly align LLMs with pairwise comparison datasets, which avoids the additional efforts for training the reward model. In DPO, the reward function is instead parameterized as the log density ratio between the policy model and a fixed reference model (typically the one after supervised fine-tuning). DPO enjoys efficient and stable optimization by training the log density ratio with a binary classification objective, following the Bradley-Terry model (Bradley and Terry, 1952). Afterward, numerous methods have been proposed to optimize the log density ratio under various different convex functions f such as IPO (Azar et al., 2023) and EXO (Ji et al., 2024), and GPO (Tang et al., 2024) further provides a unified view of the existing algorithms. How-

ever, the parameterization of the reward function to density ratio is not always guaranteed when the policy is suboptimal, and the specific choice of convex function f lacks theoretical guidance and remains heuristic.

In this paper, we argue that a more principled framework can be developed from a general distribution-matching perspective, which can also provide direct insight in choosing the convex function f based on desirable distribution behavior. To this end, we propose f-PO, a general and principled family of preference optimization algorithms. The key innovation of our framework is to formulate preference optimization as a distribution-matching problem between the policy model and the underlying optimal model via f-divergence minimization. Such formulation induces a general alignment objective by wrapping the density ratio of two policies in various f functions satisfying certain requirements. Importantly, our framework recovers DPO and EXO with reverse and forward KL divergences, and generalizes to other cases with arbitrary f-divergences. More concretely, we make the following contributions over state-of-the-art preference optimization algorithms:

- We derive generalized offline alignment objectives from a distribution-matching perspective using f-divergences, and cover several major previous methods as special cases under the pairwise comparison datasets setting.

- We introduce new algorithms not yet in the current literature by using novel f-divergences, such as α-divergence and Jeffrey's divergence.

- We provide principled ways to further combine f-PO with other state-of-the-art preference optimization methods such as SimPO (Meng et al., 2024), by alternative the inner density ratio with other approximations.

- We conduct detailed analysis with different f-divergences, and observe reasonable performance trade-offs across f-PO variants following corresponding divergence characteristics.

We conduct comprehensive experiments to compare f-PO with competitive existing alignment methods. Notably, when using α-divergence, our approach can consistently achieve superior or comparable performance against state-of-the-art algorithms on standard benchmarks, with up to 45.3% length-controlled winning rate on AlpacaEval 2 by finetuning on Llama3-8B-instruct. The results demonstrate that our approach leads to substantial improvements with the principled f-divergence design space.

2 Related Work

Aligning pretrained large language models with human preferences for high-quality responses is vital in natural language generation. RLHF (Christiano et al., 2017; Ouyang et al., 2022) has emerged as a common solution to address this problem by utilizing popular RL methods such as Proximal Policy Optimization (PPO) (Schulman et al., 2017a). However, this approach faces limitations due to training instability and the complexity introduced by its two-stage pipeline. DPO (Rafailov et al., 2023) overcomes this limitation by defining the preference loss as a function of the policy directly given the pairwise preference data. Followup works have extended this algorithm to utilize multiple ranked responses instead of pairwise preference data (Yuan et al., 2023b; Liu et al., 2024a; Song et al., 2024), and avoid dependence on the reference model, effectively merging the instruction tuning phase and preference optimization phase (Hong et al., 2024; Meng et al., 2024). GPO (Tang et al., 2024) unifies several DPO variants (Azar et al., 2023; Liu et al., 2024b) into a family of algorithms and provides an empirical analysis of the performance trade-off, but leave the theoretical analysis in a heuristic manner. Notably, Wang et al. (2023) also introduces f-divergence into the alignment problem, but the divergences are only applied to the regularization term of the RL formulation. Furthermore, all the above works are built on the reparameterization of reward function by policy likelihood, which cannot be guaranteed in practice (Ji et al., 2024). In contrast, our method unifies the algorithm based on the theoretical framework of statistical distribution divergence (Csiszár et al., 2004; Liese and Vajda, 2006), which offers a principled to choose the specific algorithm based on divergence characteristics. EXO (Ji et al., 2024) proposes learning the policy by exactly optimizing the RLHF objective that minimizes the reverse KL divergence against the optimal policy, which is a special case of our proposed framework. Xiao et al. (2024) leverages demonstration data for alignment using self-imitation learning. In addition, numerous offline preference optimization works propose different training objectives to emphasize various behaviors (Yuan et al., 2023a; Zhao et al., 2023; Xu et al., 2024; Ethayarajh et al., 2024).

3 Preliminaries

3.1 Reinforcement Learning from Human Feedback

We introduce the general Reinforcement Learning from Human Feedback (RLHF), an offline fine-tuning framework after the base model has been well pre-
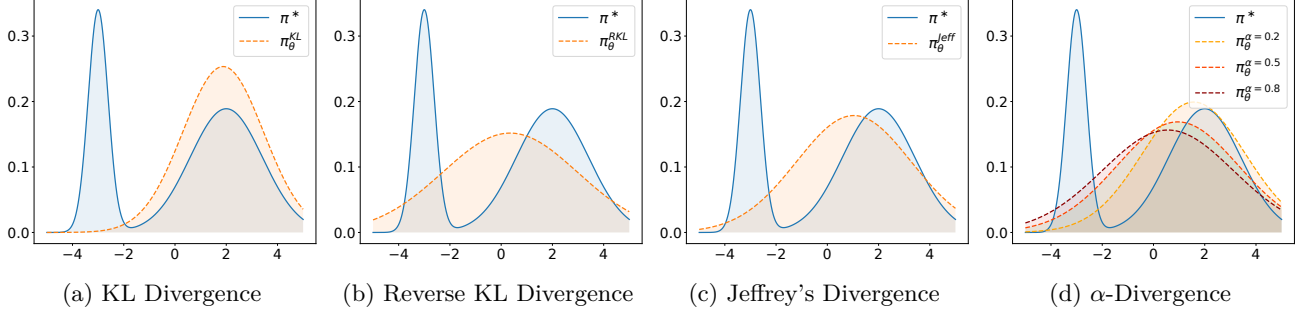
Jiaqi Han[*], Mingjian Jiang[*], Yuxuan Song, Stefano Ermon, Minkai Xu[*†]

Figure 1: Illustration of the behavior of several divergences $\mathbb{D}_f(\pi_\theta \| \pi^*)$[1] in the f-divergence family. π^* indicates the target policy while π_θ is the parameterized policy obtained by optimizing θ through minimizing certain f-divergence. Notably, α-divergence exhibits varying characteristics when adjusting the values of α.

trained. A typical RLHF process is composed of three steps: 1) supervised fine-tuning, 2) reward model learning, and 3) RL fine-tuning. In the RL fine-tuning process, given the dataset $\mathcal{D}$ consisting of prompts and reward function $r_\phi(\cdot|x)$ learned following the Bradley-Terry model (Bradley and Terry, 1952) on the preference dataset, the language model π_θ is optimized by maximizing the following objective:

$$\mathbb{E}_{x \sim \mathcal{D}, y \sim \pi_\theta(\cdot|x)} [r_\phi(x, y)] - \beta D_{\mathrm{KL}}(\pi_\theta(\cdot|x) \| \pi_{\mathrm{ref}}(\cdot|x)), \tag{1}$$

where $\pi_{\mathrm{ref}}(\cdot|x)$ is the fixed reference model obtained after supervised fine-tuning, and β is a coefficient controlling the reverse KL divergence penalty. Analytically, the solution to the regularized objective above can be written as (Peters and Schaal, 2007):

$$\pi^* = \frac{1}{Z(x)} \pi_{\mathrm{ref}} \exp\left(\beta^{-1} r_\phi(x, y)\right), \tag{2}$$

where $Z(x)$ is the partition function. The RL fine-tuning aims to learn a parameterized policy π_θ to approximate the optimal policy π^*.

3.2 Preference Optimization

The DPO method (Rafailov et al., 2023) is one of the most popular offline preference optimization methods. It uses the RL objective under the reverse KL divergence constraint Eq. (1) to build a functional mapping between the reward model and the optimal policy:

$$r(\cdot|x) = \beta \log \frac{\pi_\theta(\cdot|x)}{\pi_{\mathrm{ref}}(\cdot|x)} + \beta \log Z(x), \tag{3}$$

which allows the direct optimization of the policy π_θ. This is obtained by plugging the reward into the Bradley-Terry model and reparameterizing the reward function using the policy *i.e.*, the language model, in a supervised manner:

$$-\mathbb{E}_{(x, y_w, y_l) \sim \mathcal{D}} \Big[\log \sigma \Big(\beta \log \frac{\pi_\theta(y_w|x)}{\pi_{\mathrm{ref}}(y_w|x)} + \beta \log Z(x) - \beta \log \frac{\pi_\theta(y_l|x)}{\pi_{\mathrm{ref}}(y_l|x)} - \beta \log Z(x) \Big) \Big], \tag{4}$$

where $\mathcal{D} = (\{x, y^w, y^l\})$ is the dataset of ranked pairs of generations and their prompts, where y^w and y^l denote "winning" and "losing" samples, and σ is the sigmoid function, and the partition functions are canceled. However, this derivation of DPO is based on the assumption that optimal solution is achieved, which in practice is not guaranteed especially in the early stage training (Ji et al., 2024). This issue leads to a compromised approximation of the optimal policy, especially when the optimality of π_θ is not achieved in the early stage training.

3.3 f-divergence

Divergence describes the difference between two probability distributions. A general family of divergences is the f-divergences (Csiszár et al., 2004; Liese and Vajda, 2006), also known as the Ali-Silvey distances (Ali and Silvey, 1966). For a convex function $f : \mathbb{R}^+ \to \mathbb{R}$ that is lower-semicontinuous, strictly convex around 1, and satisfies $f(1) = 0$, given two distributions p and q, the corresponding f-divergence for these two distributions is defined as:

$$\mathbb{D}_f(p \| q) = \mathbb{E}_{q(x)} \left[f\left(\frac{p(x)}{q(x)} \right) \right], \tag{5}$$

where f is called *generator function*. Different choices of f-divergence can cover a wide class of popular divergences, including forward and reverse Kullback-Leibler

---

[1]Note that, to facilitate using f-divergence framework, in this paper we describe divergences by $\mathbb{D}_f(\pi_\theta \| \pi^*)$ instead of the conventional order $\mathbb{D}_f(\pi^* \| \pi_\theta)$, which results in interpretations of the divergence behaviors different from other literature (Ji et al., 2024).

Table 1: Summary of some typical f-divergences $\mathbb{D}_f(p\|q)$ together with generator functions. Part of the list of divergences are borrowed from Nielsen and Nock (2013); Nowozin et al. (2016). For all divergences we have the generator function $f : \mathbb{R}^+ \to \mathbb{R} \cup \{+\infty\}$ strictly convex and lower-semicontinuous. The function also satisfies $f(1) = 0$ which ensures that $\mathbb{D}_f(p\|q) = 0$ when $p(x) = q(x)$.

Name	$\mathbb{D}_f(p\|q)$	Generator $f(u)$
Kullback-Leibler	$\int p(x) \log \frac{p(x)}{q(x)} \, \mathrm{d}x$	$u \log u$
Reverse Kullback-Leibler	$\int q(x) \log \frac{q(x)}{p(x)} \, \mathrm{d}x$	$-\log u$
α-divergence ($\alpha \notin \{0,1\}$)	$\frac{1}{\alpha(\alpha-1)} \int \left(q(x) \left[\left(\frac{p(x)}{q(x)} \right)^{1-\alpha} - (1-\alpha)\left(\frac{p(x)}{q(x)} \right) - \alpha \right] \right) \mathrm{d}x$	$\frac{1}{\alpha(\alpha-1)} \left(u^{1-\alpha} - (1-\alpha)u - \alpha \right)$
Jeffrey	$\int (p(x) - q(x)) \log \left(\frac{p(x)}{q(x)} \right) \mathrm{d}x$	$(u-1) \log u$
Jensen-Shannon	$\frac{1}{2} \int p(x) \log \frac{2p(x)}{p(x)+q(x)} + q(x) \log \frac{2q(x)}{p(x)+q(x)} \, \mathrm{d}x$	$-(u+1) \log \frac{1+u}{2} + u \log u$
Squared Hellinger	$\int \left(\sqrt{p(x)} - \sqrt{q(x)} \right)^2 \mathrm{d}x$	$\left(\sqrt{u} - 1 \right)^2$

(KL) divergence, Jensen-Shannon (JS) divergence, and Jeffrey's divergence, etc. We provide a summary of their analytic forms and corresponding generator functions in Table 1. An illustration of different behaviors of these divergences is presented in Fig. 1.

4 Methodology

In this section, we present our proposed f-PO in details. We start by introducing the f-PO framework in Section 4.1, which conducts alignment under the probability matching perspective with f-divergence. In Section 4.2, we show practical f-PO implementation given common pair-wise preference data without reward labeling, and in Section 4.3, we further present several instances of f-PO with specified f-divergence, covering DPO as a specific case. In Section 4.4, we further discuss the connections between f-PO and several recent heuristic alignment methods, and further introduce several variants of f-PO.

4.1 The f-PO Framework

As shown in Section 3, the goal of preference optimization is to fine-tune the LLM π_θ toward the optimal policy π^*, *i.e.*, minimizing the distances between π_θ and π^*. In this paper, we explicitly define the LLM alignment task as a distribution matching problem:

Theorem 1. *Let* $\hat{\pi}_\theta(y|x) \propto \pi_\theta(y|x)^\beta \pi_{\mathrm{ref}}(y|x)^{(1-\beta)}$ *and* $\hat{\pi}^* \propto \pi_{\mathrm{ref}} \exp\left(r(x,y)\right)$. *We define our alignment objective as minimizing the following f-divergence:*

$$\mathbb{D}_f(\hat{\pi}_\theta \| \hat{\pi}^*) = \mathbb{E}_{\hat{\pi}^*} \left[f\left(\frac{\hat{\pi}_\theta}{\hat{\pi}^*} \right) \right] = \mathbb{E}_{\pi_{\mathrm{ref}}} \left[\frac{\hat{\pi}^*}{\pi_{\mathrm{ref}}} f\left(\frac{\hat{\pi}_\theta}{\hat{\pi}^*} \right) \right]. \tag{6}$$

With unlimited model capacity and perfect optimization, the optimal policy π_θ^* satisfies that $\pi_\theta^* = \pi^*$.

We leave the full proof in Appendix A. Intuitively,

this objective aims to optimize $\hat{\pi}_\theta(y|x)$ towards $\hat{\pi}^*$, which is equivalent to optimizing $\pi_\theta(y|x)$ towards π^* in Eq. (2). According to the definition of $\hat{\pi}^*$, we have that the reward function can be written as $r(x,y) = \log \hat{\pi}^*(y|x) - \log \pi_{\mathrm{ref}}(y|x)$. By further defining the log ratio $g_\theta(x,y) = \log \hat{\pi}_\theta(y|x) - \log \pi_{\mathrm{ref}}(y|x) = \beta(\log \pi_\theta(y|x) - \log \pi_{\mathrm{ref}}(y|x))$, the above objective can be simplified as:

$$\mathbb{D}_f(\hat{\pi}_\theta \| \hat{\pi}^*) = \mathbb{E}_{\pi_{\mathrm{ref}}} \left[\pi^r f\left(\frac{\pi^{g_\theta}}{\pi^r} \right) \right], \tag{7}$$

where we define $\pi^{g_\theta} = \frac{1}{Z_{g_\theta}(x)} \exp(g_\theta(x,y))$ and $\pi^r = \frac{1}{Z_r(x)} \exp(r(x,y))$, with Z_{g_θ} and Z_r being the partition functions. In practice, it is intractable to compute these partition functions in the high dimensional space. Instead, we take Monte Carlo estimation by the multiple samples from the preference dataset. In general case, for every prompt x we have K *i.i.d.* completions $y_{1:K} = \{y_1, \cdots, y_K\}$ drawn from $\pi_{\mathrm{ref}}(y|x)$, where we approximate the distributions π^{g_θ} and π^r by normalizing the exponential rewards over the K samples:

$$\pi^{g_\theta}(i|y_{1:K}, x) = \frac{e^{g_\theta(x,y_i)}}{\sum_{j=1}^K e^{g_\theta(x,y_j)}}, \tag{8}$$

$$\pi^r(i|y_{1:K}, x) = \frac{e^{r(x,y_i)}}{\sum_{j=1}^K e^{r(x,y_j)}}. \tag{9}$$

Plugging the above expression into Eq. (7), we have the complete form of f-PO objective $\mathcal{L}_{f\text{-PO}}$ as follows:

$$\mathcal{L}_{f\text{-PO}}(\pi_\theta) = \mathbb{E}_{x \sim \mathcal{D}} \mathbb{E}_{y_{1:K} \sim \pi_{\mathrm{ref}}} \Bigg[\frac{e^{r(x,y_i)}}{\sum_{j=1}^K e^{r(x,y_j)}} f\left(\frac{e^{g_\theta(x,y_i)}}{\sum_{j=1}^K e^{g_\theta(x,y_j)}} \Bigg/ \frac{e^{r(x,y_i)}}{\sum_{j=1}^K e^{r(x,y_j)}} \right) \Bigg]. \tag{10}$$

Formally, the approximated objective with K samples enjoys the following theoretical property,

Jiaqi Han*, Mingjian Jiang*, Yuxuan Song, Stefano Ermon, Minkai Xu*†

Theorem 2. *With $K \to \infty$, asymptotically we have*

$$\mathcal{L}_{f\text{-PO}}(\pi_\theta) = \mathbb{E}_{x \sim \mathcal{D}} \mathbb{D}_f \left(\hat{\pi}_\theta(y|x) \| \hat{\pi}^*(y|x) \right), \qquad (11)$$

where $\hat{\pi}_\theta(y|x)$ and $\hat{\pi}^*(y|x)$ follows the definition in Theorem 1.

The theorem states that the general f-PO objective is an unbiased optimizer for f-divergences between $\hat{\pi}_\theta$ and $\hat{\pi}^*$. In the following, we give the practical form of $\mathcal{L}_{f\text{-PO}}$ under typical preference optimization scenario.

4.2 Generalized Preference Optimization

Under the distribution matching perspective, we have presented the general f-PO framework where we use f-divergence to align π_θ, with K data completions and an underlying reward model r. In this section, we show the instance of f-PO objective under practical preference optimization settings:

Pair-wise Preference Data. The most common preference dataset typically consists of pair-wise completions for each prompt, *i.e.*, setting $K = 2$. Since there are two completions y for each prompt x, we can denote them as "winning" sample y_w and "losing" sample y_l. Then Eq. (10) can be simplified as:

$$\mathcal{L}_{f\text{-PO}}(\pi_\theta) = \mathbb{E}_{x \sim \mathcal{D}} \mathbb{E}_{y_{1:2} \sim \pi_{\text{ref}}} \left[\pi^r f \left(\frac{e^{g_\theta(x, y_i)}}{\sum_{j=1}^2 e^{g_\theta(x, y_j)}} \middle/ \pi^r \right) \right],$$

$$= \mathbb{E}_{x \sim \mathcal{D}} \mathbb{E}_{\{y_w, y_l\} \sim \pi_{\text{ref}}} \left[\pi^{r_w} f \left(\frac{\sigma(g_\theta(x, y_w) - g_\theta(x, y_l))}{\pi^{r_w}} \right) \right.$$

$$\left. + \pi^{r_l} f \left(\frac{\sigma(g_\theta(x, y_l) - g_\theta(x, y_w))}{\pi^{r_l}} \right) \right],$$
$$(12)$$

where $\pi^{r_w} = \sigma(r_w - r_l)$ and $\pi^{r_l} = \sigma(r_l - r_w)$ are self-normalized reward values for winning and losing samples, respectively. Such formulation follows the Bradley-Terry model (Bradley and Terry, 1952).

Preference Data without Reward Value. In common cases, the preference data is accessible without reward label. We follow the BT model form and turn π^{r_w} and π^{r_l} to binary supervision labels, *i.e.*, we set $\pi^{r_w} \approx 1$ and $\pi^{r_l} \approx 0$. To avoid the numerical issue, we smooth the labels by setting $\pi^{r_w} = 1 - \epsilon$ and $\pi^{r_l} = \epsilon$, where $\epsilon > 0$ is a hyperparameter controlling the smoothness. Then we further simplify Eq. (12) as:

$$\mathcal{L}_{f\text{-PO}}(\pi_\theta) = \mathbb{E}_{x \sim \mathcal{D}} \mathbb{E}_{\{y_w, y_l\} \sim \pi_{\text{ref}}} \Bigg[$$

$$(1 - \epsilon) f \left(\frac{\sigma(g_\theta(x, y_w) - g_\theta(x, y_l))}{1 - \epsilon} \right) \qquad (13)$$

$$+ \epsilon f \left(\frac{\sigma(g_\theta(x, y_l) - g_\theta(x, y_w))}{\epsilon} \right) \Bigg].$$

By far we have derived the f-PO objective with common pair-wise preference data. By plugging in any function f that satisfies the requirement in Section 3.3, *e.g.* the generator functions in Table 1, we can get practical f-PO optimization objectives. We will introduce several instances of f-PO in the following section.

4.3 Instances of f-PO

In this section, we show several instances of f-PO with specified f-divergences. We first briefly show that when taking the reverse KL divergence as instance, f-PO can recover the original DPO objective. Then we present the specific instance with α-divergence, *a.k.a* α-PO, as an example, which achieves the most competitive results in our experiments.

DPO as f-PO with Reverse KL. First, we show that from our f-divergence minimization perspective, it can be easily demonstrated that DPO (Rafailov et al., 2023) corresponds to minimizing the reverse KL divergence $\mathbb{D}_{\text{RKL}}(\pi_\theta \| \pi^*)$, as briefly derived here:

Proof. By realizing the function f in Eq. (13) as the generator function of reverse KL, *i.e.*, $-\log u$, we have:

$$\mathcal{L}_{f\text{-PO-RKL}}(\pi_\theta) = \mathbb{E}_{x \sim \mathcal{D}} \mathbb{E}_{\{y_w, y_l\} \sim \pi_{\text{ref}}} \Big[$$

$$- (1 - \epsilon) \log \left(\sigma(g_\theta(x, y_w) - g_\theta(x, y_l)) \right)$$

$$- \epsilon \log \left(\sigma(g_\theta(x, y_l) - g_\theta(x, y_w)) \right)$$

$$- (1 - \epsilon) \log(1 - \epsilon) - \epsilon \log \epsilon \Big].$$

When $\epsilon \to 0$, we have that $\lim_{\epsilon \to 0}(1 - \epsilon) \log(1 - \epsilon) = 0$ and $\lim_{\epsilon \to 0} \epsilon \log \epsilon = 0$. Besides, we also have $\lim_{\epsilon \to 0} \epsilon \log \left(\sigma(g_\theta(x, y_l) - g_\theta(x, y_w)) \right) = 0$. Therefore, without reward label smoothing, the objective is:

$$\lim_{\epsilon \to 0} \mathcal{L}_{f\text{-PO-RKL}}(\pi_\theta) = - \log \left(\sigma(g_\theta(x, y_w) - g_\theta(x, y_l)) \right),$$

which recovers the original DPO objective Eq. (4). □

Similarly, when combined with forward KL function $u \log u$, the f-PO instance corresponds to EXO (Ji et al., 2024). The derivation is provided in Appendix A due to limited space, akin to the DPO derivation above but with a different generator function.

f-PO with α-divergence. We present the alignment objective when using α-divergence, *a.k.a* α-PO, as an example. By plugging the generator function of α-divergence into Eq. (13), we have the objective as:

$$\mathcal{L}_{f\text{-PO}}(\pi_\theta) = \mathbb{E}_{x \sim \mathcal{D}} \mathbb{E}_{\{y_w, y_l\} \sim \pi_{\text{ref}}} \Bigg[\frac{1}{\alpha(\alpha - 1)}$$

$$(1 - \epsilon) \left(u_1^{1-\alpha} - (1 - \alpha) u_1 - \alpha \right) \qquad (14)$$

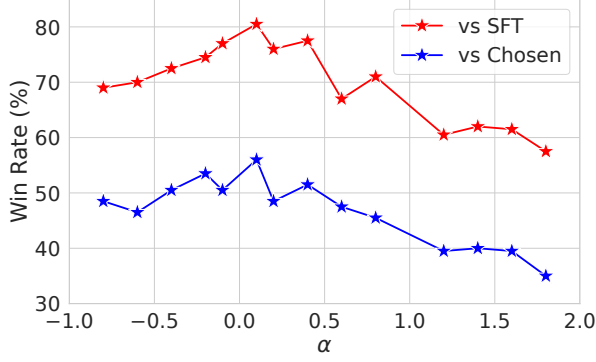$$+ \epsilon \left(u_2^{1-\alpha} - (1 - \alpha) u_2 - \alpha \right) \Bigg],$$

Figure 2: The win rate (%) against SFT and Chosen of Pythia-2.8B fine-tuned by *f*-PO on Anthropic HH dataset with different values of α.
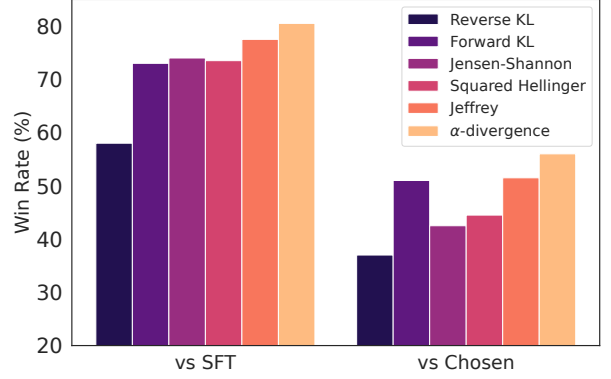

Figure 3: The win rate (%) of Pythia-2.8B fine-tuned by 6 different *f*-divergences using *f*-PO on Anthropic HH dataset. α-divergence generally performs the best.

where the densities ratios u are defined as $u_1 = \frac{\sigma(g_\theta(x,y_w) - g_\theta(x,y_l))}{1-\epsilon}$ and $u_2 = \frac{\sigma(g_\theta(x,y_l) - g_\theta(x,y_w))}{\epsilon}$, respectively. α-divergence covers a family of divergences with varying α values. As $|\alpha|$ increases, the divergence becomes more sensitive to differences in the tails of the distributions. Specifically, it converges to KL divergence when $\alpha \to 0$ and reverse KL divergence when $\alpha \to 1$. This property allows us to interpolate between KL and reverse KL. In our empirical study, we achieve the most competitive results using α divergence among all *f*-PO variants.

4.4 Empirical Variants with Approximations

As shown in the objective $\mathcal{L}_{f\text{-PO}}(\pi_\theta)$ in Eq. (13), we conduct optimization over the log density odds $g_\theta(x, y_w) - g_\theta(x, y_l) = \beta(\log \pi_\theta(y_w|x) - \log \pi_{\text{ref}}(y_w|x) - \log \pi_\theta(y_l|x) + \log \pi_{\text{ref}}(y_l|x))$. This expression is reported to have several drawbacks (Meng et al., 2024): 1) the presence of π_{ref} incurs additional training cost, and 2) the objective mismatch the generation metric where sentence likelihood is averaged by sentence length. To this end, heuristic approximations have been proposed, which yield better empirical results:

$$\hat{g}_\theta(x, y_w) - \hat{g}_\theta(x, y_l) = \frac{\beta}{|y_w|} \log \pi_\theta(y_w|x) - \frac{\beta}{|y_l|} \log \pi_\theta(y_l|x) - \gamma,$$
(15)

where $|y|$ is the sentence length of corresponding completions y, and γ is a target margin hyperparameter approximating $\beta(\log \pi_{\text{ref}}(y_w|x) - \log \pi_{\text{ref}}(y_l|x))$. Recent progress shows that this practical implementation simplifies the training objective and greatly improves alignment performance (Meng et al., 2024). Thanks to the generality of our framework, empirically we can take this approximation into our method by substituting the g_θ odds in Eq. (13) with the $\hat{g}_\theta$ one above. This empirical variant enjoys improved training efficiency and also leads to significantly higher performance.

5 Experiments

In this section, we start by investigating the choices of the *f*-divergence in our *f*-PO in Section 5.1 that helps us identity the most performant form, which is further evaluated on popular LLM benchmarks in Section 5.2.

5.1 Experiments on the Choice of *f* in *f*-PO

In this subsection, we empirically investigate the effect of different instantiations of *f* in performing preference optimization of LLMs. We detail our setups as follows.

Datasets. Here we utilize two datasets, the Reddit TL;DR summarization dataset (Völske et al., 2017) and the Anthropic Helpful and Harmless (HH) dialogue datasets (Bai et al., 2022), in this subsection. The TL;DR dataset contains Reddit posts with user-generated summaries. We employ the filtered dataset (Stiennon et al., 2020) to train the SFT model, and the preference dataset for preference optimization. The HH dataset provides multi-turn dialogues between users and AI assistants. We leverage the chosen response for SFT model and the helpful subset of this preference dataset for preference optimization.

Setups. Following Ji et al. (2024), we explore two alignment protocols for each dataset. **1.** Direct preference training (w/ `Preferences`): We employ a dataset $\mathcal{D}_{\text{pref}}$ with data points in the form of (x, y_w, y_l), where x is the input, y_w and y_l represent the preferred and rejected response, respectively. **2.** Reward-based training (w/ `Reward Model`): We employ a dataset $\mathcal{D}_{\text{rw}}$, where each instance comprises an input x and K pairs of (y_k, r_k), with y_k being a response generated by the SFT model, and r_k being the corresponding reward labeled by certain reward model trained on the preference dataset $\mathcal{D}_{\text{pref}}$.

Jiaqi Han*, Mingjian Jiang*, Yuxuan Song, Stefano Ermon, Minkai Xu*†

Table 2: The win rate (%) against SFT and Chosen of Pythia-2.8B fine-tuned by various alignment algorithms on TLDR and Anthropic HH datasets. Results of baselines are taken from Ji et al. (2024). Best numbers of preference optimization methods are boldfaced.

| | **TLDR** | | **Anthropic HH** | |
	vs SFT	**vs Chosen**	**vs SFT**	**vs Chosen**
w/ Preferences				
DPO_{pref}	57.0	30.5	58.0	37.0
EXO_{pref}	83.0	55.0	73.0	51.0
$f\text{-PO}_{pref}$	**84.0**	**61.5**	**80.5**	**56.0**
w/ Reward Model				
Best-of-N	83.5	60.0	86.0	63.0
PPO	77.0	52.0	66.5	52.0
DPO_{rw}	70.0	41.0	75.5	49.0
EXO_{rw}	84.5	**64.0**	83.5	60.0
$f\text{-PO}_{rw}$	**87.5**	60.0	**85.0**	**63.5**

Model and evaluation. Here we use Pythia-2.8B (Biderman et al., 2023) as the backbone to perform preference optimization. For evaluation, we adopt GPT-4 for zero-shot pair-wise comparisons of outputs generated by our current model against those produced by (1) the SFT model and (2) the preferred responses from our original preference dataset. We reuse the evaluation prompt from Ji et al. (2024) detailed in Appendix C, which has been shown to align closely with human judgments (Rafailov et al., 2023).

The effect of α in α-PO. We first examine the impact of α in α-PO in the preference training setup by performing a sweep over α, with results illustrated in Fig. 2. We find that by controlling α we are able to effectively maneuver between mode-seeking and mode-covering, which results in a smooth change of win rate. Notably, the performance approximates that of forward KL when α approaches 0 and that of reverse KL when α becomes close to 1 (*c.f.* Fig. 3), aligning with our theoretical analysis. We obtain the best result with $\alpha = 0.1$ under this experimental setup.

The effect of different f in the f-divergence family. We further study the empirical performance when optimizing the model with different f enumerated in Table 1. For α-divergence, we use α that leads to the best performance as investigated in the previous study. As depicted in Fig. 3, we observe that α-divergence generally contributes to the highest win rate with up to $> 20\%$ enhancement compared with reverse KL (*i.e.*, DPO), while some other types of f-divergences, *e.g.*, Jeffrey's divergence, also demonstrate competitive performance. Inspired by these results, we will prioritize α-PO in the benchmarks in the following sections due to its superior performance.

Overall comparison. We thoroughly compare f-

PO against several baselines (*e.g.*, DPO and EXO) in both w/ Preferences and w/ Reward Model setting with results displayed in Table 2. For the reward model setting, we additionally report the numbers of PPO (Schulman et al., 2017b) and Best-of-N, which samples $N = 128$ outputs from the SFT policy and then selects the highest scored response judged by the reward model. As shown in Table 2, our f-PO outperforms both DPO and EXO by a significant margin in both settings across the two datasets with, for instance, an average of 4% gain in win rate against SFT model and 6% against the chosen response over EXO_{pref}. Notably, these improvements generalize to reward-based training paradigms, underscoring the method's robustness. The substantial enhancements validate our theoretical framework and highlight the practical advantages of our generalized approach towards preference optimization of LLMs.

5.2 Benchmarking f-PO

In this subsection, we further demonstrate the strong performance of our f-PO on four popular benchmarks through four different settings.

Models and training details. We adopt two base models: Llama-3-8B (Dubey et al., 2024) and Mistral-7B (Jiang et al., 2023), under two setups, *Base* and *Instruct*, following the approach outlined in Meng et al. (2024). In the *Base* setup, we first train an SFT model on UltraChat-200k (Ding et al., 2023a) before performing preference optimization on UltraFeedback (Cui et al., 2023), offering high reproducibility with open-source data and methods. The *Instruct* setup instead utilizes publicly available instruction-tuned models (Llama-3-8B-Instruct and Mistral-7B-Instruct) as the SFT models, which are more performant but less transparent due to undisclosed finetuning procedures. We use specifically constructed preference datasets for Llama-3 and Mistral, following Meng et al. (2024). More details are deferred to Appendix B.

This leads to four configurations in total: Llama-3-Base, Llama-3-Instruct, Mistral-Base, and Mistral-Instruct. For our f-PO, we employed the α-divergence algorithm family with a properly tuned α parameter, inspired by the study in Section 5.1. Notably, we also utilize the SimPO-style parameterization of f-PO in Eq. (15) due to its superior performance, and report the best performance over a range of α. Detailed hyperparameters can be found in Appendix B.

Evaluation metrics. We evaluate our models using four popular open-ended instruction-following benchmarks: AlpacaEval 2 (Dubois et al., 2024), MT-Bench (Zheng et al., 2023), Arena-Hard 0.1 (Li et al., 2024), and Open LLM Leaderboard v2 (Fourrier et al.,

Table 3: Results on AlpacaEval 2 (Li et al., 2023), Arena-Hard (Li et al., 2024), and MT-Bench (Zheng et al., 2023) under four settings. LC and WR denote length-controlled and raw win rate, respectively. SFT models are trained on UltraChat dataset for Base settings and initialized as off-the-shelf models for Instruct settings. Results of baselines are taken from Meng et al. (2024).

| Method | Mistral-Base (7B) | | | | | Mistral-Instruct (7B) | | | | |
| | AlpacaEval 2 | | Arena-Hard | MT-Bench | | AlpacaEval 2 | | Arena-Hard | MT-Bench | |
	LC (%)	WR (%)	WR (%)	GPT-4 Turbo	GPT-4	LC (%)	WR (%)	WR (%)	GPT-4 Turbo	GPT-4
SFT	8.4	6.2	1.3	4.8	6.3	17.1	14.7	12.6	6.2	7.5
RRHF	11.6	10.2	5.8	5.4	6.7	25.3	24.8	18.1	6.5	7.6
SLiC-HF	10.9	8.9	7.3	5.8	**7.4**	24.1	24.6	18.9	6.5	**7.8**
DPO	15.1	12.5	10.4	5.9	7.3	26.8	24.9	16.3	6.3	7.6
IPO	11.8	9.4	7.5	5.5	7.2	20.3	20.3	16.2	6.4	**7.8**
CPO	9.8	8.9	6.9	5.4	6.8	23.8	28.8	**22.6**	6.3	7.5
KTO	13.1	9.1	5.6	5.4	7.0	24.5	23.6	17.9	6.4	7.7
ORPO	14.7	12.2	7.0	5.8	7.3	24.5	24.9	20.8	6.4	7.7
R-DPO	17.4	12.8	8.0	5.9	**7.4**	27.3	24.5	16.1	6.2	7.5
SimPO	21.5	20.8	16.6	**6.0**	7.3	32.1	34.8	21.0	**6.6**	7.6
f-PO	**23.7**	**22.0**	**16.8**	**6.0**	**7.4**	**32.9**	**35.8**	21.5	**6.6**	7.6

| Method | Llama3-Base (8B) | | | | | Llama3-Instruct (8B) | | | | |
| | AlpacaEval 2 | | Arena-Hard | MT-Bench | | AlpacaEval 2 | | Arena-Hard | MT-Bench | |
	LC (%)	WR (%)	WR (%)	GPT-4 Turbo	GPT-4	LC (%)	WR (%)	WR (%)	GPT-4 Turbo	GPT-4
SFT	6.2	4.6	3.3	5.2	6.6	26.0	25.3	22.3	6.9	8.1
RRHF	12.1	10.1	6.3	5.8	7.0	31.3	28.4	26.5	6.7	7.9
SLiC-HF	12.3	13.7	6.0	6.3	7.6	26.9	27.5	26.2	6.8	8.1
DPO	18.2	15.5	15.9	6.5	7.7	40.3	37.9	32.6	7.0	8.0
IPO	14.4	14.2	17.8	6.5	7.4	35.6	35.6	30.5	7.0	**8.3**
CPO	10.8	8.1	5.8	6.0	7.4	28.9	32.2	28.8	7.0	8.0
KTO	14.2	12.4	12.5	6.3	**7.8**	33.1	31.8	26.4	6.9	8.2
ORPO	12.2	10.6	10.8	6.1	7.6	28.5	27.4	25.8	6.8	8.0
R-DPO	17.6	14.4	17.2	**6.6**	7.5	41.1	37.8	33.1	7.0	8.0
SimPO	22.0	20.3	23.4	**6.6**	7.7	44.7	40.5	**33.8**	7.0	8.0
f-PO	**23.5**	**20.9**	**28.2**	**6.6**	**7.8**	**45.3**	**41.0**	33.5	**7.1**	8.2

Table 4: Results on LLM Leaderboard v2 with Llama3-Instruct (8B) and Mistral-Base (7B).

	IFEval	BBH	MATH Lvl 5	GPQA	MUSR	MMLU-PRO	Average
SimPO-Llama-3-8B-Instruct	65.04	26.71	2.57	**5.82**	8.15	27.66	22.66
f-PO-Llama3-8B-Instruct	**67.69**	**28.75**	**4.98**	4.92	**9.08**	**28.79**	**24.04**
SimPO-Mistral-7B-Base	47.01	22.33	0.60	4.47	8.03	18.91	16.89
f-PO-Mistral-7B-Base	**48.46**	**24.16**	**1.66**	**6.38**	**12.73**	**19.72**	**18.85**

2024). These benchmarks assess diverse conversational abilities across various queries. For AlpacaEval 2, we report both the raw win rates (WR) and length-controlled win rates (LC). MT-Bench scores are averaged using GPT-4 and GPT-4-Preview-1106 as judges. Arena-Hard results are reported as win rates against the baseline model.

Baselines. We compare our f-PO with a comprehensive set of offline preference optimization methods: (i) **RRHF** (Yuan et al., 2023a) and **SLiC-HF** (Zhao et al., 2023) rank losses using length-normalized and direct log-likelihood with an SFT objective, respectively; (ii) **DPO** (Rafailov et al., 2023) refers to the original direct preference optimization approach; (iii) **IPO** (Azar et al., 2023) commits to preventing potential overfitting problems in DPO; (iv) **CPO** (Xu

et al., 2024) employs direct likelihood as a reward and trains in conjunction with a behavior cloning objective for winning responses; (v) **KTO** (Ethayarajh et al., 2024) eliminates the need for pair-wise preference datasets; (vi) **ORPO** (Hong et al., 2024) introduces a reference-model-free odd ratio term to penalize undesired generation styles; (vii) **R-DPO** (Park et al., 2024) incorporates additional regularization to prevent length exploitation; (viii) **SimPO** (Meng et al., 2024) proposes reference-free rewards that incorporate length normalization and target reward margins between winning and losing responses.

Results. Table 3 shows the primary evaluation outcomes of Mistral-7B and Llama-3-8B in both Base and Instruct configurations. While all methods effectively boost the performance over the SFT model, our

proposed method, f-PO, achieves the highest score on 16 out of 20 metrics while being competitive on the rest, which highlights the efficacy and robustness of our technique. Quantitatively, f-PO achieves an average improvement of 1.4% over the strongest baseline SimPO on the AlpacaEval 2 length-controlled win rate metric. Interestingly, in the Llama-3-8B Base setup, we are able to obtain a remarkable enhancement of 4.8% (28.2% against 23.4%) in win rate compared with SimPO on Arena-Hard, further demonstrating the superiority of f-PO. The results consistently advocate f-PO as an effective and broadly applicable LLM alignment approach across a wide suite of models and tasks. The results in Table 4 further verifies the superiority of our f-PO, where f-PO outperforms SimPO consistently with both models (Llama3-8B-Instruct and Mistral-7B-Base) by a significant margin, leading to 8% relative improvement in average score. Notably, f-PO enhances the score of Mistral-7B-Base on the MUSR benchmark from 8.03 to 12.73, and Llama3-8B-Instruct on the IFEval benchmark from 22.66 to 24.04. The results further strengthen our findings of f-PO as a general and effective preference optimization approach.

6 Conclusion

In this paper, we introduce f-PO, a generalized approach towards preference optimization through f-divergence minimization. Our key insight lies in viewing LLM alignment task as distribution matching between the optimal and parameterized policy with the objective defined with f-divergence. Our approach generalizes existing techniques to a broader family of alignment objectives, among which the variant of α-PO has been demonstrated to achieve state-of-the-art performance on various challenging LLM benchmarks. Our work offers a novel theoretical perspective in understanding preference optimization while enjoying strong empirical performance and wide applicability across different models and tasks.

Acknowledgment

References

Ali, S. M. and Silvey, S. D. (1966). A general class of coefficients of divergence of one distribution from another. *Journal of the Royal Statistical Society: Series B (Methodological)*, 28(1):131–142.

Askell, A., Bai, Y., Chen, A., Drain, D., Ganguli, D., Henighan, T., Jones, A., Joseph, N., Mann, B., DasSarma, N., et al. (2021). A general language assistant as a laboratory for alignment. *ArXiv preprint*, abs/2112.00861.

Azar, M. G., Rowland, M., Piot, B., Guo, D., Calandriello, D., Valko, M., and Munos, R. (2023). A general theoretical paradigm to understand learning from human preferences. *ArXiv*, abs/2310.12036.

Bai, Y., Jones, A., Ndousse, K., Askell, A., Chen, A., DasSarma, N., Drain, D., Fort, S., Ganguli, D., Henighan, T., Joseph, N., Kadavath, S., Kernion, J., Conerly, T., El-Showk, S., Elhage, N., Hatfield-Dodds, Z., Hernandez, D., Hume, T., Johnston, S., Kravec, S., Lovitt, L., Nanda, N., Olsson, C., Amodei, D., Brown, T., Clark, J., McCandlish, S., Olah, C., Mann, B., and Kaplan, J. (2022). Training a helpful and harmless assistant with reinforcement learning from human feedback.

Bender, E. M., Gebru, T., McMillan-Major, A., and Shmitchell, S. (2021). On the dangers of stochastic parrots: Can language models be too big? In *Proceedings of the 2021 ACM conference on fairness, accountability, and transparency*, pages 610–623.

Biderman, S., Schoelkopf, H., Anthony, Q., Bradley, H., O'Brien, K., Hallahan, E., Khan, M. A., Purohit, S., Prashanth, U. S., Raff, E., Skowron, A., Sutawika, L., and van der Wal, O. (2023). Pythia: A suite for analyzing large language models across training and scaling.

Bradley, R. A. and Terry, M. E. (1952). Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39(3/4):324–345.

Christiano, P. F., Leike, J., Brown, T. B., Martic, M., Legg, S., and Amodei, D. (2017). Deep reinforcement learning from human preferences. In Guyon, I., von Luxburg, U., Bengio, S., Wallach, H. M., Fergus, R., Vishwanathan, S. V. N., and Garnett, R., editors, *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 4299–4307.

Csiszár, I., Shields, P. C., et al. (2004). Information theory and statistics: A tutorial. *Foundations and Trends® in Communications and Information Theory*, 1(4):417–528.

Cui, G., Yuan, L., Ding, N., Yao, G., Zhu, W., Ni,

Y., Xie, G., Liu, Z., and Sun, M. (2023). Ultrafeedback: Boosting language models with high-quality feedback.

Ding, N., Chen, Y., Xu, B., Qin, Y., Zheng, Z., Hu, S., Liu, Z., Sun, M., and Zhou, B. (2023a). Enhancing chat language models by scaling high-quality instructional conversations.

Ding, N., Chen, Y., Xu, B., Qin, Y., Zheng, Z., Hu, S., Liu, Z., Sun, M., and Zhou, B. (2023b). Enhancing chat language models by scaling high-quality instructional conversations. In *EMNLP*.

Dubey, A., Jauhri, A., Pandey, A., Kadian, A., Al-Dahle, A., Letman, A., Mathur, A., et al. (2024). The llama 3 herd of models.

Dubois, Y., Galambosi, B., Liang, P., and Hashimoto, T. B. (2024). Length-controlled AlpacaEval: A simple way to debias automatic evaluators. *ArXiv*, abs/2404.04475.

Ethayarajh, K., Xu, W., Muennighoff, N., Jurafsky, D., and Kiela, D. (2024). KTO: Model alignment as prospect theoretic optimization. *ArXiv*, abs/2402.01306.

Fourrier, C., Habib, N., Lozovskaya, A., Szafer, K., and Wolf, T. (2024). Open llm leaderboard v2. https://huggingface.co/spaces/open-llm-leaderboard/open_llm_leaderboard.

Hong, J., Lee, N., and Thorne, J. (2024). ORPO: Monolithic preference optimization without reference model. *ArXiv*, abs/2403.07691.

Ji, H., Lu, C., Niu, Y., Ke, P., Wang, H., Zhu, J., Tang, J., and Huang, M. (2024). Towards efficient and exact optimization of language model alignment. *arXiv preprint arXiv:2402.00856*.

Ji, Z., Lee, N., Frieske, R., Yu, T., Su, D., Xu, Y., Ishii, E., Bang, Y. J., Madotto, A., and Fung, P. (2023). Survey of hallucination in natural language generation. *ACM Computing Surveys*, 55(12):1–38.

Jiang, A. Q., Sablayrolles, A., Mensch, A., Bamford, C., Chaplot, D. S., de las Casas, D., Bressand, F., Lengyel, G., Lample, G., Saulnier, L., Lavaud, L. R., Lachaux, M.-A., Stock, P., Scao, T. L., Lavril, T., Wang, T., Lacroix, T., and Sayed, W. E. (2023). Mistral 7b.

Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Leike, J., Krueger, D., Everitt, T., Martic, M., Maini, V., and Legg, S. (2018). Scalable agent alignment via reward modeling: a research direction. *ArXiv preprint*, abs/1811.07871.

Li, T., Chiang, W.-L., Frick, E., Dunlap, L., Wu, T., Zhu, B., Gonzalez, J. E., and Stoica, I. (2024).

From crowdsourced data to high-quality benchmarks: Arena-hard and benchbuilder pipeline.

Li, X., Zhang, T., Dubois, Y., Taori, R., Gulrajani, I., Guestrin, C., Liang, P., and Hashimoto, T. B. (2023). AlpacaEval: An automatic evaluator of instruction-following models. https://github.com/tatsu-lab/alpaca_eval.

Liese, F. and Vajda, I. (2006). On divergences and informations in statistics and information theory. *IEEE Transactions on Information Theory*, 52(10):4394–4412.

Liu, T., Qin, Z., Wu, J., Shen, J., Khalman, M., Joshi, R., Zhao, Y., Saleh, M., Baumgartner, S., Liu, J., Liu, P. J., and Wang, X. (2024a). Lipo: Listwise preference optimization through learning-to-rank.

Liu, T., Zhao, Y., Joshi, R., Khalman, M., Saleh, M., Liu, P. J., and Liu, J. (2024b). Statistical rejection sampling improves preference optimization.

Meng, Y., Xia, M., and Chen, D. (2024). Simpo: Simple preference optimization with a reference-free reward. *arXiv preprint arXiv:2405.14734*.

Nielsen, F. and Nock, R. (2013). On the chi square and higher-order chi distances for approximating f-divergences. *IEEE Signal Processing Letters*, 21(1):10–13.

Nowozin, S., Cseke, B., and Tomioka, R. (2016). f-gan: Training generative neural samplers using variational divergence minimization. *Advances in neural information processing systems*, 29.

Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C., Mishkin, P., Zhang, C., Agarwal, S., Slama, K., Ray, A., et al. (2022). Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744.

Park, R., Rafailov, R., Ermon, S., and Finn, C. (2024). Disentangling length from quality in direct preference optimization. *ArXiv*, abs/2403.19159.

Peters, J. and Schaal, S. (2007). Reinforcement learning by reward-weighted regression for operational space control. In *Proceedings of the 24th international conference on Machine learning*, pages 745–750.

Rafailov, R., Sharma, A., Mitchell, E., Ermon, S., Manning, C. D., and Finn, C. (2023). Direct preference optimization: Your language model is secretly a reward model. *ArXiv preprint*, abs/2305.18290.

Schulman, J., Wolski, F., Dhariwal, P., Radford, A., and Klimov, O. (2017a). Proximal policy optimization algorithms.

Schulman, J., Wolski, F., Dhariwal, P., Radford, A., and Klimov, O. (2017b). Proximal policy optimization algorithms. *ArXiv preprint*, abs/1707.06347.

Song, F., Yu, B., Li, M., Yu, H., Huang, F., Li, Y., and Wang, H. (2024). Preference ranking optimization for human alignment.

Stiennon, N., Ouyang, L., Wu, J., Ziegler, D., Lowe, R., Voss, C., Radford, A., Amodei, D., and Christiano, P. F. (2020). Learning to summarize with human feedback. *Advances in Neural Information Processing Systems*, 33:3008–3021.

Tang, Y., Guo, Z. D., Zheng, Z., Calandriello, D., Munos, R., Rowland, M., Richemond, P. H., Valko, M., Pires, B. Á., and Piot, B. (2024). Generalized preference optimization: A unified approach to offline alignment. *arXiv preprint arXiv:2402.05749*.

Völske, M., Potthast, M., Syed, S., and Stein, B. (2017). TL;DR: Mining Reddit to learn automatic summarization. In Wang, L., Cheung, J. C. K., Carenini, G., and Liu, F., editors, *Proceedings of the Workshop on New Frontiers in Summarization*, pages 59–63, Copenhagen, Denmark. Association for Computational Linguistics.

Wang, C., Jiang, Y., Yang, C., Liu, H., and Chen, Y. (2023). Beyond reverse kl: Generalizing direct preference optimization with diverse divergence constraints. *arXiv preprint arXiv:2309.16240*.

Xiao, T., Li, M., Yuan, Y., Zhu, H., Cui, C., and Honavar, V. G. (2024). How to leverage demonstration data in alignment for large language model? a self-imitation learning perspective. *arXiv preprint arXiv:2410.10093*.

Xu, H., Sharaf, A., Chen, Y., Tan, W., Shen, L., Durme, B. V., Murray, K., and Kim, Y. J. (2024). Contrastive preference optimization: Pushing the boundaries of LLM performance in machine translation. *ArXiv*, abs/2401.08417.

Yuan, H., Yuan, Z., Tan, C., Wang, W., Huang, S., and Huang, F. (2023a). RRHF: Rank responses to align language models with human feedback. In *NeurIPS*.

Yuan, Z., Yuan, H., Tan, C., Wang, W., Huang, S., and Huang, F. (2023b). Rrhf: Rank responses to align language models with human feedback without tears.

Zhao, Y., Joshi, R., Liu, T., Khalman, M., Saleh, M., and Liu, P. J. (2023). SLiC-HF: Sequence likelihood calibration with human feedback. *ArXiv*, abs/2305.10425.

Zheng, L., Chiang, W.-L., Sheng, Y., Zhuang, S., Wu, Z., Zhuang, Y., Lin, Z., Li, Z., Li, D., Xing, E., et al. (2023). Judging LLM-as-a-judge with MT-Bench and Chatbot Arena. In *NeurIPS Datasets and Benchmarks Track*.

Supplementary Materials for
f-PO: Generalizing Preference Optimization with
f-divergence Minimization

A Proofs

A.1 Proof of Theorem 1

In this section, we present the formal proof of Theorem 1. For clarity we first repeat the theorem below.

Theorem 1. *Let $\hat{\pi}_\theta(y|x) \propto \pi_\theta(y|x)^\beta \pi_{\text{ref}}(y|x)^{(1-\beta)}$ and $\hat{\pi}^* \propto \pi_{\text{ref}} \exp(r(x,y))$. We define our alignment objective as minimizing the following f-divergence:*

$$\mathbb{D}_f(\hat{\pi}_\theta \| \hat{\pi}^*) = \mathbb{E}_{\hat{\pi}^*}\left[f\left(\frac{\hat{\pi}_\theta}{\hat{\pi}^*} \right) \right] = \mathbb{E}_{\pi_{\text{ref}}}\left[\frac{\hat{\pi}^*}{\pi_{\text{ref}}} f\left(\frac{\hat{\pi}_\theta}{\hat{\pi}^*} \right) \right]. \tag{16}$$

With unlimited model capacity and perfect optimization, the optimal policy π_θ^* satisfies that $\pi_\theta^* = \pi^*$.

To prove the theorem, we consider the following property of f-divergence.

Lemma 1. $\mathbb{D}_f(p\|q) \geq 0$ *and the global optimal of $\mathbb{D}_f(p\|q)$ only achieves when $q = p$.*

Proof. By the definition of a convex function, for any $t \geq 0, f(t) \geq f(1) + f'(1)(t-1)$. Given that $f(1) = 0$, the inequality simplifies to $f(t) \geq f'(1)(t-1)$. Then we have

$$f\left(\frac{p(x)}{q(x)} \right) \geq f'(1)\left(\frac{p(x)}{q(x)} - 1 \right). \tag{17}$$

Taking expectation over q, we have that

$$\mathbb{D}_f(p\|q) = \int q(x) f\left(\frac{p(x)}{q(x)} \right) dx \geq \int q(x) f'(1)\left(\frac{p(x)}{q(x)} - 1 \right) dx,$$

$$= f'(1)\left(\int p(x)dx - \int q(x)dx \right) = f'(1)(1-1) = 0. \tag{18}$$

The equation is satisfied only when $f\left(\frac{p(x)}{q(x)} \right) = 0$, *i.e.* $\frac{p(x)}{q(x)} = 1$. Then we finish the proof of Lemma 1. □

Now we get back to the definition of $\hat{\pi}_\theta$ and $\hat{\pi}^*$:

$$\hat{\pi}_\theta(y \mid x) = \pi_\theta(y \mid x)^\beta \pi_{\text{ref}}(y \mid x)^{(1-\beta)}/Z_\theta(x), \qquad \hat{\pi}^* = \pi_{\text{ref}}(y|x) \exp(r(x,y))/Z_r(x). \tag{19}$$

Here $Z_\theta(x)$ and $Z_r(x)$ are the corresponding partition functions. By Lemma 1 we know that the minimizer of our f-divergence objective is achieved only when $\hat{\pi}_{\theta^*} = \hat{\pi}^*$, which is equivalent to

$$\pi_{\theta^*}(y \mid x)^\beta \pi_{\text{ref}}(y \mid x)^{(1-\beta)}/Z_\theta(x) = \pi_{\text{ref}}(y|x) \exp(r(x,y))/Z_r(x),$$

$$\Leftrightarrow \pi_{\theta^*}(y \mid x)^\beta/Z_\theta(x) = \pi_{\text{ref}}^\beta(y|x) \exp(r(x,y))/Z_r(x),$$

$$\Leftrightarrow \pi_{\theta^*}(y \mid x) = \pi_{\text{ref}}(y|x) \exp(\beta^{-1}r(x,y))(Z_\theta(x)/Z_r(x))^{\frac{1}{\beta}}. \tag{20}$$

Eliminating the partition-related constant, we have:

$$\pi_{\theta^*}(y \mid x) \propto \pi_{\text{ref}}(y|x) \exp(\beta^{-1}r(x,y)) = \frac{1}{Z(x)} \pi_{\text{ref}}(y|x) \exp(\beta^{-1}r(x,y)) = \pi^*, \tag{21}$$

which concludes the proof.

Jiaqi Han*, **Mingjian Jiang***, **Yuxuan Song, Stefano Ermon, Minkai Xu***†

A.2 Proof of Theorem 2

Theorem 2. *With $K \to \infty$, asymptotically we have*

$$\mathcal{L}_{f\text{-PO}}(\pi_\theta) = \mathbb{E}_{x \sim \mathcal{D}} \mathbb{D}_f \left(\hat{\pi}_\theta(y|x) \| \hat{\pi}^*(y|x) \right), \tag{22}$$

where $\hat{\pi}_\theta(y|x)$ and $\hat{\pi}^*(y|x)$ follows the definition in Theorem 1.

Proof. We first consider following equation $\sum_{j=1}^{K} e^{g_\theta(x,y_j)}/K = \mathbb{E}_{\pi_{\text{ref}}(y|x)} e^{g_\theta(x,y)}$. When $K \to \infty$, the above equation holds because the LHS is the unbiased Monte Carlo estimation of the expectation. Note that $g_\theta(x,y) = \log \hat{\pi}_\theta(y \mid x) - \log \pi_{\text{ref}}(y \mid x) = \log \frac{\hat{\pi}_\theta(y|x)}{\pi_{\text{ref}}(y|x)}$. Then we have the fact that:

$$\sum_{j=1}^{K} e^{g_\theta(x,y_j)} = K \mathbb{E}_{\pi_{\text{ref}}(y|x)} \frac{\hat{\pi}_\theta(y \mid x)}{\pi_{\text{ref}}(y \mid x)} = K. \tag{23}$$

Similarly, we obtain an expectation formulation of $\sum_{j=1}^{K} e^{r(x,y_j)}$ as:

$$\sum_{j=1}^{K} e^{r(x,y_j)} = K \mathbb{E}_{\pi_{\text{ref}}(y|x)} e^{r(x,y)} = K Z_r(x). \tag{24}$$

Here $Z_r(x)$ is a partition function with respect to r and x. Putting the above Eq. 24 and Eq. 23 to the Eq. 10, we obtain

$$\mathcal{L}_{f\text{-PO}}(\pi_\theta) = \mathbb{E}_{x \sim \mathcal{D}} \mathbb{E}_{y_{1:K} \sim \pi_{\text{ref}}} \left[\frac{e^{r(x,y_i)}}{\sum_{j=1}^{K} e^{r(x,y_j)}} f \left(\frac{e^{g_\theta(x,y_i)}}{\sum_{j=1}^{K} e^{g_\theta(x,y_j)}} \Big/ \frac{e^{r(x,y_i)}}{\sum_{j=1}^{K} e^{r(x,y_j)}} \right) \right],$$

$$= \mathbb{E}_{x \sim \mathcal{D}} \left[\mathbb{E}_{y_{1:K} \sim \pi_{\text{ref}}} \frac{e^{r(x,y_i)}}{K Z_r(x)} f \left(\frac{\frac{\hat{\pi}_\theta(y_i|x)}{\pi_{\text{ref}}(y_i|x)}}{K} \Big/ \frac{e^{r(x,y_i)}}{K Z_r(x)} \right) \right],$$

$$= \mathbb{E}_{x \sim \mathcal{D}} \left[\frac{1}{K} \sum_{i=1}^{K} \frac{\pi_{\text{ref}}(y_i \mid x) e^{r(x,y_i)}}{Z_r(x)} f(\hat{\pi}_\theta(y_i \mid x) \Big/ \frac{\pi_{\text{ref}}(y_i \mid x) e^{r(x,y_i)}}{Z_r(x)}) \right]. \tag{25}$$

Recalling that $\frac{\pi_{\text{ref}}(y_i|x) e^{r(x,y_i)}}{Z_r(x)} = \hat{\pi}^*(y \mid x)$ given the definition, we have that

$$\mathcal{L}_{f\text{-PO}}(\pi_\theta) = \mathbb{E}_{x \sim \mathcal{D}} \left[\mathbb{E}_{\hat{\pi}^*(y|x)} f(\frac{\hat{\pi}_\theta(y_i \mid x)}{\hat{\pi}^*(y \mid x)}) \right] = \mathbb{E}_{x \sim \mathcal{D}} \mathbb{D}_f \left(\hat{\pi}_\theta(y \mid x) \| \hat{\pi}^*(y \mid x) \right), \tag{26}$$

which finishes the proof. □

A.3 Derivation for EXO as f-PO with Forward KL

In this section, we provide a brief derivation on how EXO (Ji et al., 2024) corresponds to minimizing the forward KL divergence $\mathbb{D}_{\text{KL}}(\pi_\theta \| \pi^*)$.

Proof. By realizing the function f in Eq. (13) as the generator function of forward KL, *i.e.*, $u \log u$, we have,

$$\mathcal{L}_{f\text{-PO-RKL}}(\pi_\theta) = \mathbb{E}_{x \sim \mathcal{D}} \mathbb{E}_{\{y_w, y_l\} \sim \pi_{\text{ref}}} \left[(1 - \epsilon) \cdot \frac{\sigma(g_\theta(x, y_w) - g_\theta(x, y_l))}{1 - \epsilon} \log \left(\frac{\sigma(g_\theta(x, y_w) - g_\theta(x, y_l))}{1 - \epsilon} \right) \right.$$

$$\left. + \epsilon \cdot \frac{\sigma(g_\theta(x, y_l) - g_\theta(x, y_w))}{\epsilon} \log \left(\frac{\sigma(g_\theta(x, y_l) - g_\theta(x, y_w))}{\epsilon} \right) \right],$$

$$= \mathbb{E}_{x \sim \mathcal{D}} \mathbb{E}_{\{y_w, y_l\} \sim \pi_{\text{ref}}} \left[\sigma(g_\theta(x, y_w) - g_\theta(x, y_l)) \log \left(\frac{\sigma(g_\theta(x, y_w) - g_\theta(x, y_l))}{1 - \epsilon} \right) \right.$$

$$\left. + \sigma(g_\theta(x, y_l) - g_\theta(x, y_w)) \log \left(\frac{\sigma(g_\theta(x, y_l) - g_\theta(x, y_w))}{\epsilon} \right) \right], \tag{27}$$

which recovers the original EXO objective Eq. (23) in Ji et al. (2024). □

B   More Experiment Details

This section provides comprehensive details on the experimental setup for our studies presented in Section 5.1 and Section 5.2. We adhere to the training and evaluation protocols outlined in EXO (Ji et al., 2024) and SimPO (Meng et al., 2024) for our method, while utilizing reported hyperparameters and results for baseline comparisons. The following subsections elaborate on the specific hyperparameters, training procedures, and evaluation methodologies employed in our experiments.

B.1   Experiment Details in Section 5.1

In our experiments, we follow the training and evaluation hyperparameters outlined in EXO (Ji et al., 2024), while for all baselines, we utilized their reported hyperparameters and results for comparison. Below, we detail the key hyperparameters employed in Ji et al. (2024), as well as those used for our method.

Hyperparameters. For f-PO$_{\mathrm{pref}}$, we set $\beta_r = 0.5$, $\beta_\pi = 0.5$, and $\alpha = 0.1$. For f-PO$_{\mathrm{rw}}$, we use $\beta_r = 0.1$, $\beta_\pi = 0.35$, $\alpha = 0.1$, and $K = 4$. The label smoothing hyperparameter ϵ in f-PO$_{\mathrm{pref}}$ is set to 1e-3. Both methods employ the Adam optimizer with a learning rate of 1e-6 and a batch size of 64, training for one epoch on each dataset, which is more than sufficient for convergence.

Evaluation Details. We sample 4 completions from the learned policy for each of 512 prompts from the test set across all datasets. For consistency, we maintain a sampling temperature of $t = 0.8$ during both training and inference. To calculate the win rate using the reward model, we compare all pairs of completions between the learned policy and base completions (either from the SFT policy or the chosen completion in the dataset).

For GPT-4 evaluations, we sample 100 prompts with 1 completion per prompt for each policy. To mitigate position bias, we evaluate each pair of generations twice, swapping their order. We use the concise prompt from (Rafailov et al., 2023) for summary quality assessment and a modified prompt for evaluating helpfulness in multi-turn dialogues. The evaluation prompts are detailed in Appendix C.

B.2   Experiment Details in Section 5.2

Hyperparameter tuning is critical for maximizing the performance of preference optimization methods. In our experiments, we adhered to the general training hyperparameters outlined in SimPO (Meng et al., 2024), while for all baselines, we utilized their reported hyperparameters and results for comparison. Below, we detail the setup and key hyperparameters employed in Meng et al. (2024), as well as those used for our method, f-PO.

Setup. In Base setup, models were fine-tuned on the UltraChat-200k corpus (Ding et al., 2023b) with a learning rate of 2e-5, batches of 128 samples, and sequence length capped at 2048 tokens. The learning schedule followed a cosine curve with a 10% warm-up phase over a single epoch. The Adam optimizer (Kingma and Ba, 2014) was used for all model training. For the preference optimization dataset, the 2048 token limit was maintained and the same cosine learning rate schedule and warm-up strategy were used. For Instruct setting, we use specifically constructed preference datasets princeton-nlp/llama3-ultrafeedback-armorm for Llama-3 and princeton-nlp/mistral-instruct-ultrafeedback for Mistral. These datasets are created using prompts from Ultra-Feedback, with chosen and rejected response pairs (y_w, y_l) regenerated using the SFT models and annotated by reward models, to approximate an on-policy setting for the Instruct models.

Hyperparameters. We used the following hyperparameters for each experimental scenario: For Mistral-Base, we set α=0.99, β=2.0, γ=1.6, with a learning rate of 3e-7. For Mistral-Instruct, we used α=0.925, β=2.5, γ=0.3, and a learning rate of 6e-7. In the Llama3-Base setting, we employed α=0.99, β=2.0, γ=1.0, with a learning rate of 7e-7. Finally, for Llama3-Instruct, we set α=0.99, β=2.6, γ=1.43, and used a learning rate of 1e-6.

Evaluation Details. For AlpacaEval 2, we generated responses using a sampling strategy. We set the temperature to 0.7 for Mistral-Base (in line with `zephyr-7b-beta`),[2] 0.5 for Mistral-Instruct (following `Snorkel-Mistral-PairRM-DPO`), and 0.9 for both Llama3 variants. Arena-Hard saw us employ greedy decoding across the board. For MT-Bench, we adhered to the official guidelines, which prescribe different sampling temperatures for various categories.

---

[2]https://github.com/tatsu-lab/alpaca_eval/blob/main/src/alpaca_eval/models_configs/zephyr-7b-beta/configs.yaml

Jiaqi Han[*], **Mingjian Jiang**[*], **Yuxuan Song, Stefano Ermon, Minkai Xu**[*†]

B.3 Computing Resources

In this work, we ran all our training experiments on four 80GB A100 GPUs, and all inference process using one 80GB A100 GPUs for each model.

C Prompt Details

In this section, we provide the specific prompts used in our model evaluation, as described in Sections 5.1 and 5.2.

C.1 Dialogue Generation Evaluation Prompt

For evaluating the helpfulness of generated dialogues, particularly in multi-turn settings, we use the following prompt structure:

Table 5: Prompt for GPT-4 evaluation on the Anthropic HH dataset. Texts in blue are placeholders to be substituted by the real data.

For the following dialogue history to a chatbot, which response is more helpful?
Dialogue history: <dialogue history>
Response A: <Response A>
Response B: <Response B>
FIRST, provide a one-sentence comparison of the two responses and explain which you feel is more helpful.
SECOND, on a new line, state only **"A"** or **"B"** to indicate which response is more helpful. Your response should use the format:
Comparison: <one-sentence comparison and explanation>
More helpful: <"A" or "B">

C.2 Summarization Evaluation Prompt

For assessing the quality of summaries, we employ the concise prompt structure from Rafailov et al. (2023):

Table 6: Prompt for GPT-4 evaluation on the TL;DR dataset. Texts in blue are placeholders to be substituted by real data.

Which of the following summaries does a better job of summarizing the most important points in the given forum post, without including unimportant or irrelevant details? A good summary is both precise and concise.
Post: <post>
Summary A: <Summary A>
Summary B: <Summary B>
FIRST, provide a one-sentence comparison of the two summaries, explaining which you prefer and why.
SECOND, on a new line, state only **"A"** or **"B"** to indicate your choice. Your response should use the format:
Comparison: <one-sentence comparison and explanation>
Preferred: <"A" or "B">