Differential Information: An Information-Theoretic Perspective on Preference Optimization

Yunjae Won **Hyunji Lee** **Hyeonbin Hwang** **Minjoon Seo**

KAIST AI
{yunjae.won, hyunji.amy.lee, hbin0701, minjoon}@kaist.ac.kr

Abstract

Direct Preference Optimization (DPO) has become a standard technique for aligning language models with human preferences in a supervised manner. Despite its empirical success, the theoretical justification behind its log-ratio reward parameterization remains incomplete. In this work, we address this gap by utilizing the DIFFERENTIAL INFORMATION DISTRIBUTION (DID): a distribution over token sequences that captures the information gained during policy updates. First, we show that when preference labels encode the differential information required to transform a reference policy into a target policy, the log-ratio reward in DPO emerges as the uniquely optimal form for learning the target policy via preference optimization. This result naturally yields a closed-form expression for the optimal sampling distribution over rejected responses. Second, we find that the condition for preferences to encode differential information is fundamentally linked to an implicit assumption regarding log-margin ordered policies—an inductive bias widely used in preference optimization yet previously unrecognized. Finally, by analyzing the entropy of the DID, we characterize how learning low-entropy differential information reinforces the policy distribution, while high-entropy differential information induces a smoothing effect, which explains the commonly observed log-likelihood displacement phenomenon. We validate our theoretical findings in synthetic experiments and extend them to real-world instruction-following datasets. Our results suggest that learning high-entropy differential information is crucial for general instruction-following, while learning low-entropy differential information benefits knowledge-intensive question answering. Overall, our work presents a unifying perspective on the DPO objective, the structure of preference data, and resulting policy behaviors through the lens of differential information.[1]

1 Introduction

Aligning Large Language Models (LLMs) with human preferences is vital for safe deployment [1, 2]. Among various methods, Direct Preference Optimization (DPO) [3] has recently gained popularity due to its robust performance, training stability, and computational efficiency [4, 5]. DPO directly optimizes the policy to maximize the empirical preference likelihood, using a Bradley-Terry reward [6, 7] derived from the KL-regularized RL objective [8, 9], specifically $r = \beta \log(\pi/\pi_{\text{ref}})$, where π is the learned policy, π_{ref} a fixed reference policy, and the KL-regularization strength $\beta > 0$. While DPO variants [10–12] and alternative reward parameterizations derived from different policy regularization methods [13, 14] have been proposed, the original log-ratio form $\beta \log(\pi/\pi_{\text{ref}})$ remains the *de facto* standard reward for preference optimization [15–18, 4]. Existing theoretical analyses have largely focused on connections to distribution matching [19, 11, 14, 20] or optimization dynamics [21, 22].

[1] Model checkpoints and training/evaluation code will be released upon acceptance.

Preprint. Under review.

However, a fundamental understanding behind the effectiveness of this specific reward has not been fully established. *Why is this form effective, and under what conditions is it optimal?*

In this paper, we present an information-theoretic perspective on preference optimization, centered on the DIFFERENTIAL INFORMATION DISTRIBUTION (DID) (Definition 3.1). The DID represents the distribution over token sequences that hold the information that updates a prior distribution into the posterior. We hypothesize that learning from preferences equates to learning the differential information that updates the reference π_{ref} into the desired target policy π^*.

In Section 3.1, we formalize the Differential Information Distribution, and identify conditions under which preference labels naturally encode the differential information required to learn the target policy π^*. We prove that the log-ratio reward of DPO is uniquely optimal for recovering π^* when preferences encode such differential information (Section 3.2). This allows us to derive the optimal distribution for sampling rejected responses in the DPO framework. Crucially, we identify a fundamental connection between preferences encoding differential information, and an ordering in policies based on increasing log-margins, an inductive bias assumed by various preference optimization methods (*e.g.*, SLiC [23], SimPO [12], and CPO [11], Section 3.3).

In Section 3.5, using Energy-Based Models, we demonstrate how preference distributions naturally encode differential information. Testing various preference optimization objectives on this dataset further confirms that DPO's log-ratio reward uniquely learns the target policy when preferences encode differential information. Extending our analysis to real-world instruction-following datasets, we observe that their preferences are more accurately interpreted as encoding the differential information needed to learn the target policy, rather than directly reflecting the target policy itself.

Subsequently, we explore the characteristics of the DID itself, focusing on its uncertainty (Section 4). We argue that the entropy of the DID reflects the policy update characteristics: learning a low-entropy differential information leads to policy reinforcement (concentrating probability mass), whereas learning a high-entropy differential information induces smoothing (Claim 4.1). This provides a novel interpretation of log-likelihood displacement (LLD), which we attribute to the learning of high-entropy differential information. We hypothesize that such high-entropy DID frequently arises in complex, multifaceted alignment objectives (Section 4.2).

We verify our hypotheses on real-world instruction-following datasets (Section 4.3). Experiments support that these preference datasets typically encode high-entropy DID, inducing LLD during DPO training. Furthermore, we demonstrate a correlation between DID entropy and the acquisition of downstream capabilities: learning high-entropy DID appears to be crucial for general instruction-following, whereas low-entropy DID benefits knowledge-intensive question answering tasks. Overall, our perspective based on differential information offers a comprehensive explanation that unifies the DPO objective, characteristics of preference data, and the resultant policy behaviors.

2 Preliminaries

In this section, we will review standard definitions for policies, preference modeling via the Bradley-Terry framework, and the DPO objective. A key concept we will build upon is the established equivalence between preference optimization and distribution matching (Theorem 2.1).

Let $\mathcal{Y}$ be the discrete space of token sequences. A policy π defines a probability distribution over $\mathcal{Y}$. We assume policies have full support, *i.e.*, $\pi(y) > 0$ for all $y \in \mathcal{Y}$. Let π^* be the target policy and π_{ref} a fixed reference policy.

Preferences are pairs (y_w, y_l) where y_w is preferred over y_l, denoted as $y_w \succ y_l$. The Bradley-Terry (BT) model [6, 7] links preferences to a underlying score r^* or distribution p^*, related by $p^*(y) \propto \exp(r^*(y))$:

$$p^*(y_w \succ y_l) \coloneqq \frac{p^*(y_w)}{p^*(y_w) + p^*(y_l)} = \sigma(r^*(y_w) - r^*(y_l))$$

where $\sigma(x) = 1/(1 + e^{-x})$. Following prior work [19], we assume that preference datasets are sufficiently large, such that its samples are able to cover $\mathcal{Y}$, enabling train-test generalization.

Given a policy π and π_{ref}, DPO uses the implicit reward $r(y) = \beta \log(\pi(y)/\pi_{\text{ref}}(y))$ to model the preference probability as:

$$p(y_w \succ y_l \mid r) \coloneqq \sigma(r(y_w) - r(y_l)) = \sigma(\beta \log \frac{\pi(y_w)}{\pi_{\text{ref}}(y_w)} - \beta \log \frac{\pi(y_l)}{\pi_{\text{ref}}(y_l)}).$$

We can also associate a Boltzmann distribution with a reward $r : \mathcal{Y} \to \mathbb{R}$:

$$P(Y = y \mid r) = \frac{\exp(r(y))}{\sum_{y'} \exp(r(y'))}.$$

Preference optimization involves maximizing the empirical preference likelihood [3, 19], which is equivalent to minimizing the KL-divergence between preference distributions. A key result connects this to matching the underlying distributions [19]:

Theorem 2.1 (Preference vs. Distribution Matching [19])**.** *Let $\mathcal{D} = \{(y_w, y_l)\}$ be a sufficiently large preference dataset where the set of y_w and y_l covers $\mathcal{Y}$. Preference optimization on $\mathcal{D}$ is equivalent to fitting the reward-induced distribution $P(Y = y \mid r)$ on the implicit preference distribution $p^*(y)$:*

$$\begin{aligned} \max_r \mathbb{E}_{(y_w,y_l)\sim\mathcal{D}} \left[\log p(y_w \succ y_l \mid r)\right] &\iff \min_r \mathbb{E}_{(y_w,y_l)\sim\mathcal{D}} \left[\mathbb{D}_{\text{KL}}[p^*(y_w \succ y_l) \parallel p(y_w \succ y_l \mid r)]\right] \\ &\iff \min_r \mathbb{D}_{\text{KL}}[p^*(y) \parallel P(Y = y \mid r)]. \end{aligned}$$

(Proof in Appendix F.1) This allows reasoning about learning $p^*(y)$ via preference optimization.

3 Preference optimization as learning differential information

In the following, we introduce the notion of differential information to characterize *information gain* in policy updates (Section 3.1), and utilize it to interpret preference optimization. We first characterize how a preference dataset naturally encodes the differential information required to learn the target policy (Section 3.2). We derive the optimal reward for learning such preferences, finding that it matches the log-ratio reward of DPO (Theorem 3.4). We further show that the condition for preferences to encode such differential information is inherently linked to an implicit assumption involving log-margin ordered policies (Section 3.3). Based on these findings, we identify the ideal distribution for sampling rejected responses (Section 3.4).

3.1 Differential Information Distribution

We begin by formalizing the Differential Information Distribution (DID), motivated from a Bayesian perspective. Consider a prior distribution $\pi_{\text{ref}}(y)$ over token sequences y. Assume that we observe some information X which updates our prior into a posterior distribution $\pi(y) = P(Y = y \mid \pi_{\text{ref}}, X)$. Further assume that X depends only on y itself, not the model that generated it, *i.e.*, X is conditionally independent of π_{ref} given y. For instance, whether a token sequence y is "safe" or "mathematically correct" is a property depending only on y, not the model that generated it.

We define the distribution over token sequences embodying this information, $P(Y = y \mid X)$, to be the Differential Information Distribution from π_{ref} to π. (See Appendix D for a detailed explanation of this setup, including Lemma D.1 proving such an X can always be constructed).

Definition 3.1 (Differential Information Distribution)**.** Let π and π_{ref} be two probability distributions over $\mathcal{Y}$ with full support. Let X be an event that satisfies the following:

$$\begin{cases} P(X \mid Y = y, \pi_{\text{ref}}) = P(X \mid Y = y) & \text{(Conditional Independence)} \\ \pi(y) = P(Y = y \mid \pi_{\text{ref}}, X) & \text{(Bayesian Update)} \end{cases}$$

Then, $P(Y = y \mid X)$ is defined as the *Differential Information Distribution* (DID) from π_{ref} to π.

In essence, the DID $P(Y = y \mid X)$ characterizes the distribution over token sequences embodying the *information gain* acquired when updating π_{ref} into π. For instance, if the update from π_{ref} into π increases the probability of safe sentences, then $P(Y = y \mid X)$ would represent the distribution of safe sentences. The following theorem (proof in Appendix D.3) provides a direct way to compute the DID and connects it to the ratio of the policy probabilities.

Theorem 3.2 (Likelihood Ratio Representation of Differential Information Distribution)**.** *For policies π, π_{ref} over $\mathcal{Y}$ with full support, the Differential Information Distribution (DID) from π_{ref} to π is equivalent to the normalized ratio distribution:*

$$P(Y = y \mid X) = \frac{\pi(y)/\pi_{\text{ref}}(y)}{Z} := q_{\pi/\pi_{\text{ref}}}(y).$$

where $Z = \sum_{y' \in \mathcal{Y}} \frac{\pi(y')}{\pi_{\text{ref}}(y')}$ is the partition function.

If we are given a reference policy π_{ref} and a target policy π^* we aim to learn, the DID $q_{\pi^*/\pi_{\text{ref}}}$ characterizes the information required to update π_{ref} into π^*. From the context of preference optimization, it is the preference distribution of a dataset that drives the policy update process. *Then when does a preference distribution encode the differential information necessary to update a reference policy into the target policy?*

3.2 When do preferences encode differential information?

Our aim is to understand how a dataset construction process results in a preference distribution p^* that corresponds to the DID $q_{\pi^*/\pi_{\text{ref}}}$ encoding the differential information needed to update π_{ref} into π^*. The following theorem identifies that a power-law structure in the DID between policies results in a preference distribution encoding differential information. Our subsequent analysis reveals that this power-law is fundamentally related to an assumption regarding the ordering of policies (Section 3.3).

Theorem 3.3 (Preferences Encoding Differential Information)**.** *Let $\mathcal{D} = \{(y_w, y_l)\}$ be a preference data where $y_w \sim \pi_{\text{ref}}$[2] and $y_l \sim \pi_l$. Let π^* be the target policy. If the Differential Information Distribution between policies match up to an exponent $\beta > 0$:*

$$q_{\pi_{\text{ref}}/\pi_l}(y) \propto q_{\pi^*/\pi_{\text{ref}}}(y)^{\beta},$$

then the preference probability $p^*(y_w \succ y_l)$ can be expressed as preferences induced by the DID:

$$p^*(y_w \succ y_l) = \sigma\left(\beta \log q_{\pi^*/\pi_{\text{ref}}}(y_w) - \beta \log q_{\pi^*/\pi_{\text{ref}}}(y_l)\right).$$

(Proof in Appendix F.2) A preference distribution of a dataset that satisfies Theorem 3.3 encodes the differential information required to learn the target policy π^*. For such preference dataset, *which reward parameterization ensures preference optimization recovers π^*?* The following theorem shows that the log-ratio form is uniquely optimal for learning π^*.

Theorem 3.4 (Optimal Reward For Learning Differential Information)**.** *Let $\mathcal{D}$ be a preference dataset encoding the differential information required to learn the target policy (Theorem 3.3). We then have:*

$$\arg\max_{\pi} \mathbb{E}_{(y_w, y_l) \sim \mathcal{D}}[\log p(y_w \succ y_l \mid r)] = \arg\min_{\pi} \mathbb{D}_{\text{KL}}[\pi^*(y) \parallel \pi(y)]$$
$$\iff r(y) = \beta \log \frac{\pi(y)}{\pi_{\text{ref}}(y)} + C$$

(Proof in Appendix F.3). This justifies DPO's log-ratio structure: if preference captures the differential information needed to improve π_{ref} towards π^*, then using the reward $r = \beta \log(\pi/\pi_{\text{ref}})$ is not merely a heuristic choice, but the only functional form that ensures preference optimization recovers π^*.

Our derivation recovers the result of Rafailov et al. [3], originally motivated by the KL-regularized RL objective. This highlights the fundamental structure of DPO in learning differential information.

3.3 Power-law structure and log-margin orderings

One might question whether the power-law relationship between the DID of policies $q_{\pi_{\text{ref}}/\pi_l}(y) \propto q_{\pi^*/\pi_{\text{ref}}}(y)^{\beta}$ is a realistic assumption. *Under which condition does the power-law structure in DID hold?* We address this question by establishing a fundamental link between the power-law assumption and an ordering in policies based on increasing log-margins.

[2]Note that it is common practice to fine-tune the reference policy π_{ref} on the chosen samples y_w [3, 24].

Theorem 3.5 (Differential Information of Log-Margin Ordered Policies)**.** *Consider three policies $\pi^*, \pi_{\text{ref}}, \pi_l$, and a preference data $\mathcal{D} = \{(y_w, y_l)\}$. Then the following statements are equivalent:*

1. ***Log-Margin Ordering*** *The log-margin of π_{ref} is sufficiently[3] larger than that of π_l if and only if the log-margin of π^* is sufficiently larger than that of π_{ref}:*

$$\begin{aligned} &\log \pi_{\text{ref}}(y_w) - \log \pi_{\text{ref}}(y_l) \gg \log \pi_l(y_w) - \log \pi_l(y_l) \\ \iff\ &\log \pi^*(y_w) - \log \pi^*(y_l) \gg \log \pi_{\text{ref}}(y_w) - \log \pi_{\text{ref}}(y_l) \end{aligned}$$

2. ***Power-Law Structure of Differential Information*** *For each outcome y in the set $\{y_w, y_l\}$, a power-law exists between the DID from π_l to π_{ref} and the DID from π_{ref} to π^*. That is, there exists some $\beta > 0$ such that:*

$$q_{\pi_{\text{ref}}/\pi_l}(y) \propto q_{\pi^*/\pi_{\text{ref}}}(y)^{\beta}.$$

(Proof in Appendix F.4) This shows that a preference over policies $\pi^* \succ \pi_{\text{ref}} \succ \pi_l$ consistent with increasing log-margins inherently assumes a power-law relationship in the DID between policies. Various methods in preference optimization (*e.g.*, SLiC [23], SimPO [12], CPO [11]) share the motivation of maximizing the log-margin to learn the target policy π^*. This reveals that the inductive bias of these methods implicitly assume a power-law structure in the DID of ordered policies.

3.4 Ideal policy for generating rejected responses

A crucial aspect of constructing preference datasets for DPO is determining *how* to generate the rejected responses y_l, assuming that chosen responses y_w are (approximately) sampled from π_{ref}. The literature has shown conflicting viewpoints: some approaches aim for "strong contrasting" signals by maximizing the quality gap between y_w and y_l [12, 25], while others favor "fine-grained" distinctions involving minimal differences [26, 17, 27]. Our work addresses this conflict by leveraging the perspective of differential information. Rather than focusing on contrast or similarity, we posit that the ideal distribution of y_l should accurately reflect the differential information between policies.

Recall that DPO recovers π^* if preferences reflect the differential information required to update π_{ref} into the target π^* (Theorem 3.4). The condition $q_{\pi_{\text{ref}}/\pi_l}(y) \propto q_{\pi^*/\pi_{\text{ref}}}(y)^{\beta}$ naturally yields an ideal distribution for sampling y_l:

$$\pi_l(y) \propto \pi_{\text{ref}}(y) \left(\frac{\pi_{\text{ref}}(y)}{\pi^*(y)} \right)^{\beta}. \tag{1}$$

If we sample the chosen and rejected responses respectively from π_{ref} and π_l, then Theorem 3.4 guarantees DPO training recovers π^*. To the best of our knowledge, this is the first closed-form expression for the ideal distribution to sample y_l from, within the DPO framework. See Appendix L for a discussion on the uniqueness of the distribution of π_l.

3.5 Experiments

We validate our theoretical findings in a synthetic setup, and extend our analysis to real-world data. We first test Theorems 3.3 and 3.4 using Energy-Based Models.

Setup We define policies $\pi_\theta(i) = \exp(\theta_i) / \sum_j \exp(\theta_j)$ for class $i \in \{1, ..., K\}$ and $\theta \in \mathbb{R}^K$. The logits of the reference policy π_{ref} are sampled from a normal distribution: $\theta_{\text{ref}} \sim \mathcal{N}(0, I)$. Next, we set the target logits $\theta^* = \theta_{\text{ref}}/\tau$ for some temperature τ (with $\tau < 1$ for reinforcing and $\tau > 1$ for smoothing) to construct the target policy π^*, ensuring it remains close to π_{ref}. The logits of π_l are set as $\theta_l = 2\theta_{\text{ref}} - \theta^*$, which aligns the DID between policies: $q_{\pi_{\text{ref}}/\pi_l} = q_{\pi^*/\pi_{\text{ref}}}$. Finally, preference pairs (y_w, y_l) are constructed by sampling $y_w \sim \pi_{\text{ref}}$ and $y_l \sim \pi_l$, and labeled as $y_w \succ y_l$.

This setup directly instantiates the conditions of Theorem 3.3, under which our theory predicts that DPO training with $r = \log \frac{\pi}{\pi_{\text{ref}}}$ should learn π^*. (Hyper-parameters in Appendix J.1.)

[3] See Appendix G for a discussion on how the "*sufficiently larger than*" ($\gg$) condition can be relaxed.

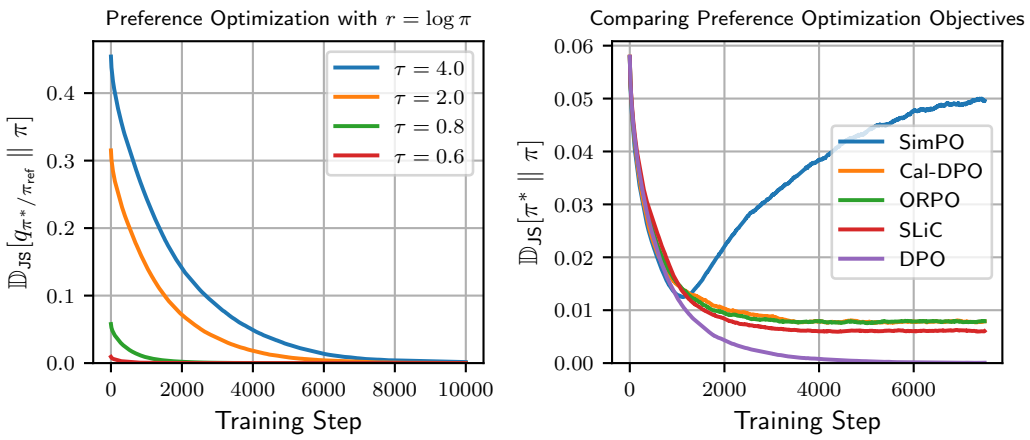

Figure 1: *Left*: Preference optimization with $r = \log \pi$ on synthetic data. The JS Divergence $\mathbb{D}_{\text{JS}}[q_{\pi^*/\pi_{\text{ref}}} \parallel \pi]$ converges to 0, confirming Theorem 3.3 that the preference encodes differential information. *Right*: Comparison of $\mathbb{D}_{\text{JS}}[\pi^* \parallel \pi]$ using different objectives on the same data with $\tau = 4$. Standard DPO ($r = \log(\pi/\pi_{\text{ref}})$, purple) optimally converges to π^*, validating Theorem 3.4.

Do preferences encode differential information? Theorem 3.3 predicts that the preference distribution should encode the differential information required to learn the target policy: $p^* = q_{\pi^*/\pi_{\text{ref}}}$. According to Theorem 2.1, a policy optimized using $r = \log \pi$ converges to the underlying preference distribution p^* [19, 11, 28]. If Theorem 3.3 holds, p^* should correspond to the Boltzmann distribution of $r^* = \log \frac{\pi^*}{\pi_{\text{ref}}}$, which is exactly $q_{\pi^*/\pi_{\text{ref}}}$. Therefore, we optimize a policy π with $r = \log \pi$ and measure the Jensen-Shannon (JS) divergence between $q_{\pi^*/\pi_{\text{ref}}}$ and π.

As shown in Figure 1, the JS divergence consistently converges towards zero. This demonstrates that the policy trained to directly fit p^* converges to the DID $q_{\pi^*/\pi_{\text{ref}}}$. We thus confirm that **our data generation process results in a preference distribution encoding the differential information required to learn the target policy**, as predicted by Theorem 3.3.

Is the log-ratio reward form optimal? Theorem 3.4 states that $r = \log \frac{\pi}{\pi_{\text{ref}}}$ is uniquely optimal for recovering π^* when preferences encode $q_{\pi^*/\pi_{\text{ref}}}$. We optimize policies using $r = \log \frac{\pi}{\pi_{\text{ref}}}$ and other objectives (SLiC [23], ORPO [10], SimPO [12], and Cal-DPO [29]) on $\mathcal{D}$ and compare $\mathbb{D}_{\text{JS}}[\pi^* \parallel \pi]$.

Figure 7 shows DPO training with $r = \log \frac{\pi}{\pi_{\text{ref}}}$ consistently minimizes $\mathbb{D}_{\text{JS}}[\pi^* \parallel \pi]$ across different settings of τ. This demonstrates **the optimality of the log-ratio reward when preference data encodes the differential information required to learn the target policy** (Theorem 3.4).

Do preferences in real datasets encode differential information? We extend our analysis to real datasets. We present a method for testing whether a preference dataset is better explained as encoding the differential information or directly encoding the target policy. If preferences encode the DID: $p^*(y) \propto q_{\pi^*/\pi_{\text{ref}}}(y)^\beta$, standard DPO $r = \beta \log(\pi/\pi_{\text{ref}})$ learns π^* (Theorem 3.4). Instead, if preferences directly encode the target: $p^*(y) \propto \pi^*(y)^\beta$, then $r' = \beta \log \pi$ would be optimal for learning π^* ([19, 11, 28], Theorem 2.1). We test which method yields better-aligned models.

We train `Mistral7B-v0.3` [30] on Ultra-Feedback [31] and Magpie-Pro [25], using DPO with $r = \beta \log \frac{\pi}{\pi_{\text{ref}}}$ vs. $r' = \beta \log \pi$ across $\beta \in \{0.2, 0.1, 0.05, 0.02\}$. We compare the estimated expected reward, using a reward model `Skywork-Reward-Gemma-2-27B-v0.2` [32].

Figure 6 shows standard DPO $r = \beta \log(\pi/\pi_{\text{ref}})$ achieves higher rewards than $r' = \beta \log \pi$. This suggests that the policy targeting $\pi^*(y) \propto \pi_{\text{ref}}(y) p^*(y)^{1/\beta}$, learned by $r = \beta \log(\pi/\pi_{\text{ref}})$, is closer to the optimal policy than the policy directly matching $\pi^*(y) \propto p^*(y)^{1/\beta}$, learned by $r' = \beta \log \pi$. This demonstrates that such **instruction-following datasets are better interpreted as preferences encoding the DID from π_{ref} to π^*, rather than directly encoding the target policy π^*.**

4 Uncertainty of differential information

We now analyze the characteristics of trained policies based on the Differential Information Distribution (DID). We relate the Shannon entropy of the DID, $H(q_{\pi/\pi_{\text{ref}}}) = -\sum_y q_{\pi/\pi_{\text{ref}}}(y) \log q_{\pi/\pi_{\text{ref}}}(y)$, with policy dynamics (Section 4.1), which leads to an information-theoretic explanation of log-likelihood displacement (Section 4.2). We empirically find that the DID entropy correlates with downstream performances (Section 4.3).

4.1 Entropy of differential information and policy dynamics

We begin by presenting an argument on the relationship between DID entropy and policy change.

Claim 4.1. *Consider a policy π derived from π_{ref} such that $\mathbb{D}_{\text{KL}}[\pi_{\text{ref}} \parallel \pi]$ is bounded. Let $q_{\pi/\pi_{\text{ref}}}$ be the corresponding Differential Information Distribution (DID). In addition, assume that for any $y' \in \{y \in \mathcal{Y} \mid \pi_{\text{ref}}(y) \approx 0\}$, we also have $\pi(y') \approx 0 \approx q_{\pi/\pi_{\text{ref}}}(y')$.*

- *If π is obtained by **reinforcing** π_{ref} (concentrating probability mass on modes of π_{ref}), we expect the DID to be deterministic, corresponding to learning a lower-entropy differential information $H(q_{\pi/\pi_{\text{ref}}}) < H(\pi_{\text{ref}})$.*
- *If π is obtained by **smoothing** π_{ref} (spreading probability mass more broadly), we expect the DID to be stochastic, corresponding to learning a higher-entropy differential information $H(q_{\pi/\pi_{\text{ref}}}) > H(\pi_{\text{ref}})$.*

(See Appendix E for a qualitative argument.) Under realistic assumptions, this claim argues that learning low-entropy differential information induces policy reinforcing, while high-entropy differential information induces policy smoothing . This provides an alternative explanation for a counterintuitive phenomenon commonly observed in DPO: *log-likelihood displacement*.

4.2 Information-theoretic view on log-likelihood displacement

Log-likelihood displacement (LLD) refers to the phenomenon in which the log-likelihood of the preferred response y_w decreases even as the model alignment improves [24, 33, 34]. Existing explanations often focus on sample similarity [35, 33] or gradient dynamics [21, 22]. We present a complementary, information-theoretic explanation.

If the preference encodes differential information that is inherently high-entropy (*i.e.*, $H(q_{\pi^*/\pi_{\text{ref}}}) > H(\pi_{\text{ref}})$), then learning this DID induces smoothing in π relative to π_{ref}, according to Claim 4.1. We hypothesize that such high-entropy preference is characteristic of complex alignment tasks, such as general instruction-following. These tasks often require learning diverse stylistic elements [36–39] or balancing multiple objectives like helpfulness and safety [40–43]. We posit that the preference p^* reflecting such diverse multifaceted criteria corresponds to a high-entropy DID.

In typical DPO setups, preferred responses y_w are often close to high-probability regions of π_{ref}. When DPO learns a high-entropy $q_{\pi^*/\pi_{\text{ref}}}$, the resulting policy smoothing redistributes probability mass from the peaks of π_{ref} more broadly to form π^*. This leads to a decrease in probability mass at these original peaks, resulting in LLD: $\log \pi^*(y_w) < \log \pi_{\text{ref}}(y_w)$.

Therefore, our perspective predicts that LLD occurs when policies are optimized with complex, multifaceted preferences, commonly found in general instruction-following datasets. This implies that policy smoothing is not a undesirable artifact, but an integral part of learning these capabilities. Conversely, if preferences encode low-entropy DID (*e.g.*, ranking based on factual correctness), policy reinforcing would be necessary to accurately learn the target policy.

4.3 Experiments

We empirically validate our hypotheses regarding the Differential Information Distribution (DID) entropy. First, we verify whether preferences in instruction-following datasets encode high-entropy DID and induce log-likelihood displacement (LLD). Second, we investigate whether the DID entropy correlates with the types of capabilities learned in the policy update process.

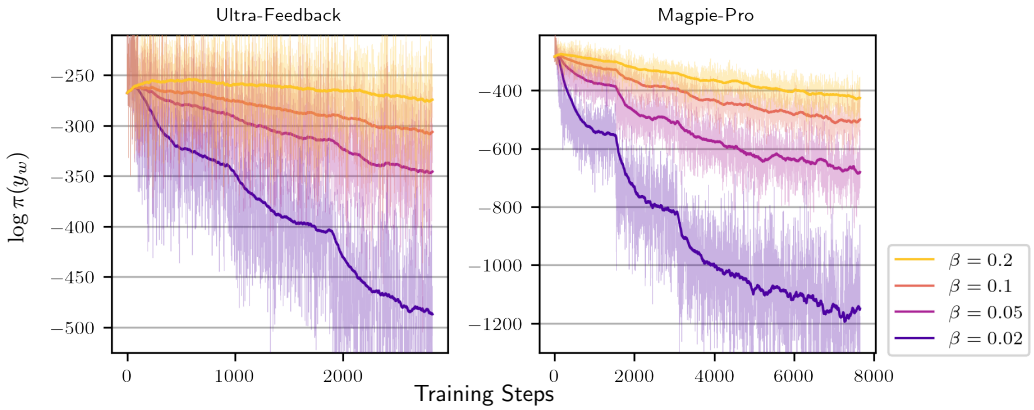

Figure 2: Observation of log-likelihood displacement (LLD) during DPO training on Ultra-Feedback (*Left*) and Magpie-Pro (*Right*). The log-probability of the chosen response consistently decreases under the DPO objective. In conjunction with Table 1, these findings support Claim 4.1, indicating that the preference distribution of instruction-following datasets often encodes high-entropy DID, which in turn induces LLD (Section 4.2).

Do instruction-following datasets encode high-entropy DID? As hypothesized in Section 4.2, we first examine if instruction-following preferences encode high-entropy DID. According to Claim 4.1, learning a high-entropy DID ($H(q_{\pi^*/\pi_{\text{ref}}}) > H(\pi_{\text{ref}})$) should induce policy smoothing and potentially lead to LLD.

We revisit the experiment in Section 3.5 using the Ultra-Feedback [31] and Magpie-Pro [25] datasets. We estimate the preference distribution $p^*(y)^{1/\beta}$ by optimizing a policy with $r = \beta \log \pi$, and measure its entropy following the process described in Appendix K. We compare this to $H(\pi_{\text{ref}}(y))$, verifying whether $H(p^*(y)^{1/\beta})$ is higher than $H(\pi_{\text{ref}})$. Then, we train another policy with standard DPO ($r = \beta \log(\pi/\pi_{\text{ref}})$)[4] and check whether LLD indeed occurs by tracking $\log \pi(y_w)$.

Table 1 shows that for both datasets the estimated $H(p^*(y)^{1/\beta})$ is larger than $H(\pi_{\text{ref}})$. Meanwhile, Figure 2 concurrently shows that $\log \pi(y_w)$ decreases during DPO training, indicating LLD. These findings support our hypothesis: **instruction-following datasets tend to encode high-entropy DID, which leads to policy smoothing and eventually log-likelihood displacement**.

Table 1: Comparison of estimated values of $H(p^{*\frac{1}{\beta}})$ and $H(\pi_{\text{ref}})$, measured in the unit of nats (*i.e.*, base-e logarithm).

β	$H(p^{*\frac{1}{\beta}})$	$H(\pi_{\text{ref}})$
Ultra-Feedback		
0.2	695.52	374.65
0.1	619.79	
0.05	611.05	
0.02	464.20	
Magpie-Pro		
0.2	882.06	418.47
0.1	693.74	
0.05	731.28	
0.02	709.71	

Does the entropy of DID reflect learning different concepts? To further investigate whether the entropy of the DID correlates with acquiring distinct capabilities, we compare the effects of standard DPO, previously observed to induce policy smoothing, with a variant specifically designed for this analysis. We thereby design DPO-PG, a method that utilizes projected gradient descent to increase the log-likelihood of chosen samples while decreasing or maintaining that of rejected responses, with the aim of sharpening the policy distribution while also increasing the log-margin (see Appendix H for further details).

We train `Mistral7B-v0.3` on two high-quality instruction-following datasets (Magpie-Pro and Magpie-G27; details in Appendix J.2). We estimate the entropy of the DID as described in Appendix K, and evaluate along two axes: General instruction-following (Arena-Hard [37] and WildBench [44]), and Knowledge-Intensive Question Answering (PIQA [45], SIQA [46], HellaSwag [47], ARC-Easy/Challenge [48], MMLU [49], GSM8k [50], and BoolQ [51]).

[4]Recall that DPO training with $r = \beta \log(\pi/\pi_{\text{ref}})$ converges the policy to $\pi^*(y) \propto \pi_{\text{ref}}(y)p^*(y)^{1/\beta}$. Thus, the DID of the converged policy corresponds to the preference distribution: $q_{\pi^*/\pi_{\text{ref}}}(y) \propto p^*(y)^{1/\beta}$.

Table 2: Estimated DID entropy $H(q_{\pi/\pi_{\text{ref}}})$ [nats] and general instruction-following results. We report the win-rate [%] on Arena-Hard (AH) and ELO on Wild-Bench (WB). DPO-PG underperforms standard DPO, suggesting that learning high-entropy DID is key to general instruction-following. († Possible outlier due to unstable measurement; see Appendix K.1.)

(a) Magpie-Pro

Method	$H(q_{\pi/\pi_{\text{ref}}})$	AH	WB
DPO $\beta = 0.2$	425.70	18.5	1145.7
DPO $\beta = 0.1$	236.24†	23.4	1146.9
DPO $\beta = 0.05$	375.82	22.4	1145.0
DPO $\beta = 0.02$	334.34	20.1	1141.7
DPO-PG	263.92	19.6	1130.1

(b) Magpie-G27

Method	$H(q_{\pi/\pi_{\text{ref}}})$	AH	WB
DPO $\beta = 0.2$	462.64	30.0	1144.6
DPO $\beta = 0.1$	434.41	27.7	1144.9
DPO $\beta = 0.05$	420.95	27.0	1148.7
DPO $\beta = 0.02$	650.75	25.4	1136.9
DPO-PG	324.05	24.0	1130.1

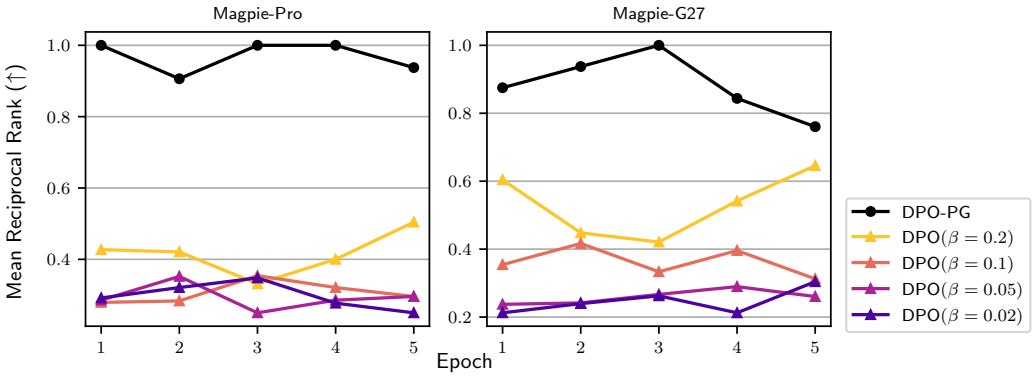

Figure 3: Mean Reciprocal Rank (MRR) across 8 QA benchmarks during training. The reinforcing method (DPO-PG) outperforms standard DPO which induces policy smoothing. This suggests QA capabilities are mainly associated with learning lower-entropy DID.

As detailed in Table 2, reinforcing the policy distribution via DPO-PG results in a lower-entropy DID, in line with Claim 4.1. DPO-PG underperforms standard DPO on general instruction-following, suggesting that the capabilities required for such tasks are tied to higher-entropy DID. In contrast, DPO-PG outperforms DPO on knowledge-intensive QA tasks (Figure 3), indicating that precise factual reasoning is mainly associated with low-entropy DID. We observe similar trends with `Gemma-2-9b` [52] (Table 3, Figure 12), demonstrating that **the DID entropy is closely related to the concepts learned from the policy update process**. These results are consistent with recent works on LLD [34, 53, 29], where we have related the DID entropy to learning dynamics.

5 Conclusion

In this work, we introduced the DIFFERENTIAL INFORMATION DISTRIBUTION (DID) to quantify information gain in policy updates, providing an information-theoretic foundation for preference optimization. We established that the log-ratio reward of DPO is uniquely optimal when preferences encode DID, a condition intrinsically related to policy orderings by log-margins. Furthermore, we linked the DID entropy to policy dynamics, which offered a new perspective on log-likelihood displacement. Empirical validation suggests learning high-entropy DID is effective for general instruction-following and low-entropy DID benefits knowledge-intensive QA. Ultimately, our framework offers a unifying perspective on DPO's objective, the structure of preference data, and emergent policy behaviors. Future work may explore how annotation protocols influence DID entropy and investigate the applicability of our framework to modalities beyond text.

References

[1] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F. Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. In Sanmi Koyejo, S. Mohamed, A. Agarwal, Danielle Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, 2022. URL http://papers.nips.cc/paper_files/paper/2022/hash/b1efde53be364a73914f58805a001731-Abstract-Conference.html.

[2] Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, et al. Training a helpful and harmless assistant with reinforcement learning from human feedback. *ArXiv preprint*, abs/2204.05862, 2022. URL https://arxiv.org/abs/2204.05862.

[3] Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D. Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine, editors, *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023. URL http://papers.nips.cc/paper_files/paper/2023/hash/a85b405ed65c6477a4fe8302b5e06ce7-Abstract-Conference.html.

[4] Wenyi Xiao, Zechuan Wang, Leilei Gan, Shuai Zhao, Wanggui He, Luu Anh Tuan, Long Chen, Hao Jiang, Zhou Zhao, and Fei Wu. A comprehensive survey of direct preference optimization: Datasets, theories, variants, and applications, 2024. URL https://arxiv.org/abs/2410.15595.

[5] Shunyu Liu, Wenkai Fang, Zetian Hu, Junjie Zhang, Yang Zhou, Kongcheng Zhang, Rongcheng Tu, Ting-En Lin, Fei Huang, Mingli Song, Yongbin Li, and Dacheng Tao. A survey of direct preference optimization, 2025. URL https://arxiv.org/abs/2503.11701.

[6] Ralph Allan Bradley and Milton E Terry. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39(3/4):324–345, 1952.

[7] R Duncan Luce et al. *Individual choice behavior*, volume 4. Wiley New York, 1959.

[8] Natasha Jaques, Shixiang Gu, Dzmitry Bahdanau, José Miguel Hernández-Lobato, Richard E. Turner, and Douglas Eck. Sequence tutor: Conservative fine-tuning of sequence generation models with kl-control. In Doina Precup and Yee Whye Teh, editors, *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, volume 70 of *Proceedings of Machine Learning Research*, pages 1645–1654. PMLR, 2017. URL http://proceedings.mlr.press/v70/jaques17a.html.

[9] Natasha Jaques, Judy Hanwen Shen, Asma Ghandeharioun, Craig Ferguson, Agata Lapedriza, Noah Jones, Shixiang Gu, and Rosalind Picard. Human-centric dialog training via offline reinforcement learning. In Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu, editors, *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3985–4003, Online, 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.327. URL https://aclanthology.org/2020.emnlp-main.327.

[10] Jiwoo Hong, Noah Lee, and James Thorne. Reference-free monolithic preference optimization with odds ratio. *ArXiv preprint*, abs/2403.07691, 2024. URL https://arxiv.org/abs/2403.07691.

[11] Haoran Xu, Amr Sharaf, Yunmo Chen, Weiting Tan, Lingfeng Shen, Benjamin Van Durme, Kenton Murray, and Young Jin Kim. Contrastive preference optimization: Pushing the boundaries of LLM performance in machine translation. In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=51iwkioZpn.

[12] Yu Meng, Mengzhou Xia, and Danqi Chen. Simpo: Simple preference optimization with a reference-free reward. In Amir Globersons, Lester Mackey, Danielle Belgrave, Angela Fan, Ulrich Paquet, Jakub M. Tomczak, and Cheng Zhang, editors, *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024. URL http://papers.nips.cc/paper_files/paper/2024/hash/e099c1c9699814af0be873a175361713-Abstract-Conference.html.

[13] Chaoqi Wang, Yibo Jiang, Chenghao Yang, Han Liu, and Yuxin Chen. Beyond reverse KL: generalizing direct preference optimization with diverse divergence constraints. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=2cRzmWXK9N.

[14] Jiaqi Han, Mingjian Jiang, Yuxuan Song, Stefano Ermon, and Minkai Xu. f-po: Generalizing preference optimization with f-divergence minimization. *ArXiv preprint*, abs/2410.21662, 2024. URL https://arxiv.org/abs/2410.21662.

[15] Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, Binyuan Hui, Luo Ji, Mei Li, Junyang Lin, Runji Lin, Dayiheng Liu, Gao Liu, Chengqiang Lu, Keming Lu, Jianxin Ma, Rui Men, Xingzhang Ren, Xuancheng Ren, Chuanqi Tan, Sinan Tan, Jianhong Tu, Peng Wang, Shijie Wang, Wei Wang, Shengguang Wu, Benfeng Xu, Jin Xu, An Yang, Hao Yang, Jian Yang, Shusheng Yang, Yang Yao, Bowen Yu, Hongyi Yuan, Zheng Yuan, Jianwei Zhang, Xingxuan Zhang, Yichang Zhang, Zhenru Zhang, Chang Zhou, Jingren Zhou, Xiaohuan Zhou, and Tianhang Zhu. Qwen technical report. *ArXiv preprint*, abs/2309.16609, 2023. URL https://arxiv.org/abs/2309.16609.

[16] Hamish Ivison, Yizhong Wang, Valentina Pyatkin, Nathan Lambert, Matthew Peters, Pradeep Dasigi, Joel Jang, David Wadden, Noah A. Smith, Iz Beltagy, and Hannaneh Hajishirzi. Camels in a changing climate: Enhancing lm adaptation with tulu 2, 2023.

[17] Lewis Tunstall, Edward Beeching, Nathan Lambert, Nazneen Rajani, Kashif Rasul, Younes Belkada, Shengyi Huang, Leandro von Werra, Clémentine Fourrier, Nathan Habib, et al. Zephyr: Direct distillation of lm alignment. *ArXiv preprint*, abs/2310.16944, 2023. URL https://arxiv.org/abs/2310.16944.

[18] Nathan Lambert, Jacob Morrison, Valentina Pyatkin, Shengyi Huang, Hamish Ivison, Faeze Brahman, Lester James V. Miranda, Alisa Liu, Nouha Dziri, Shane Lyu, Yuling Gu, Saumya Malik, Victoria Graf, Jena D. Hwang, Jiangjiang Yang, Ronan Le Bras, Oyvind Tafjord, Chris Wilhelm, Luca Soldaini, Noah A. Smith, Yizhong Wang, Pradeep Dasigi, and Hannaneh Hajishirzi. Tulu 3: Pushing frontiers in open language model post-training, 2024. URL https://arxiv.org/abs/2411.15124.

[19] Vincent Dumoulin, Daniel D. Johnson, Pablo Samuel Castro, Hugo Larochelle, and Yann Dauphin. A density estimation perspective on learning from pairwise human preferences, 2023. URL https://arxiv.org/abs/2311.14115.

[20] Haozhe Ji, Cheng Lu, Yilin Niu, Pei Ke, Hongning Wang, Jun Zhu, Jie Tang, and Minlie Huang. Towards efficient exact optimization of language model alignment. In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=66k81s33p3.

[21] Duanyu Feng, Bowen Qin, Chen Huang, Zheng Zhang, and Wenqiang Lei. Towards analyzing and understanding the limitations of dpo: A theoretical perspective. *ArXiv preprint*, abs/2404.04626, 2024. URL https://arxiv.org/abs/2404.04626.

[22] Xin Mao, Feng-Lin Li, Huimin Xu, Wei Zhang, Wang Chen, and Anh Tuan Luu. As simple as fine-tuning: Llm alignment via bidirectional negative feedback loss. *ArXiv preprint*, abs/2410.04834, 2024. URL https://arxiv.org/abs/2410.04834.

[23] Yao Zhao, Rishabh Joshi, Tianqi Liu, Misha Khalman, Mohammad Saleh, and Peter J Liu. Slic-hf: Sequence likelihood calibration with human feedback. *ArXiv preprint*, abs/2305.10425, 2023. URL https://arxiv.org/abs/2305.10425.

[24] Rafael Rafailov, Joey Hejna, Ryan Park, and Chelsea Finn. From r to q^*: Your language model is secretly a q-function, 2024. URL https://arxiv.org/abs/2404.12358.

[25] Zhangchen Xu, Fengqing Jiang, Luyao Niu, Yuntian Deng, Radha Poovendran, Yejin Choi, and Bill Yuchen Lin. Magpie: Alignment data synthesis from scratch by prompting aligned llms with nothing. *ArXiv preprint*, abs/2406.08464, 2024. URL https://arxiv.org/abs/2406.08464.

[26] Zicheng Lin, Tian Liang, Jiahao Xu, Qiuzhi Lin, Xing Wang, Ruilin Luo, Chufan Shi, Siheng Li, Yujiu Yang, and Zhaopeng Tu. Critical tokens matter: Token-level contrastive estimation enhances llm's reasoning capability, 2024. URL https://arxiv.org/abs/2411.19943.

[27] Geyang Guo, Ranchi Zhao, Tianyi Tang, Xin Zhao, and Ji-Rong Wen. Beyond imitation: Leveraging fine-grained quality signals for alignment. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=LNLjU5C5dK.

[28] Yixin Liu, Pengfei Liu, and Arman Cohan. Understanding reference policies in direct preference optimization, 2024. URL https://arxiv.org/abs/2407.13709.

[29] Teng Xiao, Yige Yuan, Huaisheng Zhu, Mingxiao Li, and Vasant G. Honavar. Cal-dpo: Calibrated direct preference optimization for language model alignment. In Amir Globersons, Lester Mackey, Danielle Belgrave, Angela Fan, Ulrich Paquet, Jakub M. Tomczak, and Cheng Zhang, editors, *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024. URL http://papers.nips.cc/paper_files/paper/2024/hash/cf8b2205e39f81726a8d828ecbe00ad0-Abstract-Conference.html.

[30] Albert Jiang, Alexandre Sablayrolles, Alexis Tacnet, Antoine Roux, Arthur Mensch, Audrey Herblin-Stoop, Baptiste Bout, Baudouin de Monicault, Blanche Savary, Bam4d, Caroline Feldman, Devendra Singh Chaplot, Diego de las Casas, Eleonore Arcelin, Emma Bou Hanna, Etienne Metzger, Gianna Lengyel, Guillaume Bour, Guillaume Lample, Harizo Rajaona, Jean-Malo Delignon, Jia Li, Justus Murke, Louis Martin, Louis Ternon, Lucile Saulnier, Lélio Renard Lavaud, Margaret Jennings, Marie Pellat, Marie Torelli, Marie-Anne Lachaux, Nicolas Schuhl, Patrick von Platen, Pierre Stock, Sandeep Subramanian, Sophia Yang, Szymon Antoniak, Teven Le Scao, Thibaut Lavril, Timothée Lacroix, Théophile Gervet, Thomas Wang, Valera Nemychnikova, William El Sayed, and William Marshall. mistralai/mistral-7b-v0.3, 2024. URL https://huggingface.co/mistralai/Mistral-7B-v0.3.

[31] Ganqu Cui, Lifan Yuan, Ning Ding, Guanming Yao, Wei Zhu, Yuan Ni, Guotong Xie, Zhiyuan Liu, and Maosong Sun. Ultrafeedback: Boosting language models with high-quality feedback, 2023.

[32] Chris Yuhao Liu, Liang Zeng, Jiacai Liu, Rui Yan, Jujie He, Chaojie Wang, Shuicheng Yan, Yang Liu, and Yahui Zhou. Skywork-reward: Bag of tricks for reward modeling in llms. *ArXiv preprint*, abs/2410.18451, 2024. URL https://arxiv.org/abs/2410.18451.

[33] Noam Razin, Sadhika Malladi, Adithya Bhaskar, Danqi Chen, Sanjeev Arora, and Boris Hanin. Unintentional unalignment: Likelihood displacement in direct preference optimization, 2024. URL https://arxiv.org/abs/2410.08847.

[34] Zhengyan Shi, Sander Land, Acyr Locatelli, Matthieu Geist, and Max Bartolo. Understanding likelihood over-optimisation in direct alignment algorithms, 2024. URL https://arxiv.org/abs/2410.11677.

[35] Arka Pal, Deep Karkhanis, Samuel Dooley, Manley Roberts, Siddartha Naidu, and Colin White. Smaug: Fixing failure modes of preference optimisation with dpo-positive. *ArXiv preprint*, abs/2402.13228, 2024. URL https://arxiv.org/abs/2402.13228.

[36] Yann Dubois, Balázs Galambosi, Percy Liang, and Tatsunori B. Hashimoto. Length-controlled alpacaeval: A simple way to debias automatic evaluators, 2024. URL https://arxiv.org/abs/2404.04475.

[37] Tianle Li*, Wei-Lin Chiang*, Evan Frick, Lisa Dunlap, Banghua Zhu, Joseph E. Gonzalez, and Ion Stoica. From live data to high-quality benchmarks: The arena-hard pipeline, 2024. URL https://lmsys.org/blog/2024-04-19-arena-hard/.

[38] Guiming Hardy Chen, Shunian Chen, Ziche Liu, Feng Jiang, and Benyou Wang. Humans or LLMs as the judge? a study on judgement bias. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen, editors, *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 8301–8327, Miami, Florida, USA, 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.474. URL https://aclanthology.org/2024.emnlp-main.474/.

[39] Ryan Park, Rafael Rafailov, Stefano Ermon, and Chelsea Finn. Disentangling length from quality in direct preference optimization. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, *Findings of the Association for Computational Linguistics: ACL 2024*, pages 4998–5017, Bangkok, Thailand, 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-acl.297. URL https://aclanthology.org/2024.findings-acl.297/.

[40] Zhanhui Zhou, Jie Liu, Jing Shao, Xiangyu Yue, Chao Yang, Wanli Ouyang, and Yu Qiao. Beyond one-preference-fits-all alignment: Multi-objective direct preference optimization, 2023. URL https://arxiv.org/abs/2310.03708.

[41] Jiaming Ji, Mickel Liu, Josef Dai, Xuehai Pan, Chi Zhang, Ce Bian, Boyuan Chen, Ruiyang Sun, Yizhou Wang, and Yaodong Yang. Beavertails: Towards improved safety alignment of LLM via a human-preference dataset. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine, editors, *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023. URL http://papers.nips.cc/paper_files/paper/2023/hash/4dbb61cb68671edc4ca3712d70083b9f-Abstract-Datasets_and_Benchmarks.html.

[42] Zeqiu Wu, Yushi Hu, Weijia Shi, Nouha Dziri, Alane Suhr, Prithviraj Ammanabrolu, Noah A. Smith, Mari Ostendorf, and Hannaneh Hajishirzi. Fine-grained human feedback gives better rewards for language model training. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine, editors, *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023. URL http://papers.nips.cc/paper_files/paper/2023/hash/b8c90b65739ae8417e61eadb521f63d5-Abstract-Conference.html.

[43] Alexandre Ramé, Guillaume Couairon, Corentin Dancette, Jean-Baptiste Gaya, Mustafa Shukor, Laure Soulier, and Matthieu Cord. Rewarded soups: towards pareto-optimal alignment by interpolating weights fine-tuned on diverse rewards. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine, editors, *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023. URL http://papers.nips.cc/paper_files/paper/2023/hash/e12a3b98b67e8395f639fde4c2b03168-Abstract-Conference.html.

[44] Bill Yuchen Lin, Yuntian Deng, Khyathi Chandu, Faeze Brahman, Abhilasha Ravichander, Valentina Pyatkin, Nouha Dziri, Ronan Le Bras, and Yejin Choi. Wildbench: Benchmarking llms with challenging tasks from real users in the wild, 2024. URL https://arxiv.org/abs/2406.04770.

[45] Yonatan Bisk, Rowan Zellers, Ronan LeBras, Jianfeng Gao, and Yejin Choi. PIQA: reasoning about physical commonsense in natural language. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 7432–7439. AAAI Press, 2020. URL https://aaai.org/ojs/index.php/AAAI/article/view/6239.

[46] Maarten Sap, Hannah Rashkin, Derek Chen, Ronan Le Bras, and Yejin Choi. Social IQa: Commonsense reasoning about social interactions. In Kentaro Inui, Jing Jiang, Vincent Ng,

and Xiaojun Wan, editors, *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4463–4473, Hong Kong, China, 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1454. URL https://aclanthology.org/D19-1454.

[47] Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. HellaSwag: Can a machine really finish your sentence? In Anna Korhonen, David Traum, and Lluís Màrquez, editors, *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4791–4800, Florence, Italy, 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1472. URL https://aclanthology.org/P19-1472.

[48] Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. Think you have solved question answering? try arc, the ai2 reasoning challenge. *ArXiv preprint*, abs/1803.05457, 2018. URL https://arxiv.org/abs/1803.05457.

[49] Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021. URL https://openreview.net/forum?id=d7KBjmI3GmQ.

[50] Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems. *ArXiv preprint*, abs/2110.14168, 2021. URL https://arxiv.org/abs/2110.14168.

[51] Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. BoolQ: Exploring the surprising difficulty of natural yes/no questions. In Jill Burstein, Christy Doran, and Thamar Solorio, editors, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2924–2936, Minneapolis, Minnesota, 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1300. URL https://aclanthology.org/N19-1300.

[52] Gemma Team, Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, Léonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ramé, Johan Ferret, Peter Liu, Pouya Tafti, Abe Friesen, Michelle Casbon, Sabela Ramos, Ravin Kumar, Charline Le Lan, Sammy Jerome, Anton Tsitsulin, Nino Vieillard, Piotr Stanczyk, Sertan Girgin, Nikola Momchev, Matt Hoffman, Shantanu Thakoor, Jean-Bastien Grill, Behnam Neyshabur, Olivier Bachem, Alanna Walton, Aliaksei Severyn, Alicia Parrish, Aliya Ahmad, Allen Hutchison, Alvin Abdagic, Amanda Carl, Amy Shen, Andy Brock, Andy Coenen, Anthony Laforge, Antonia Paterson, Ben Bastian, Bilal Piot, Bo Wu, Brandon Royal, Charlie Chen, Chintu Kumar, Chris Perry, Chris Welty, Christopher A. Choquette-Choo, Danila Sinopalnikov, David Weinberger, Dimple Vijaykumar, Dominika Rogozińska, Dustin Herbison, Elisa Bandy, Emma Wang, Eric Noland, Erica Moreira, Evan Senter, Evgenii Eltyshev, Francesco Visin, Gabriel Rasskin, Gary Wei, Glenn Cameron, Gus Martins, Hadi Hashemi, Hanna Klimczak-Plucińska, Harleen Batra, Harsh Dhand, Ivan Nardini, Jacinda Mein, Jack Zhou, James Svensson, Jeff Stanway, Jetha Chan, Jin Peng Zhou, Joana Carrasqueira, Joana Iljazi, Jocelyn Becker, Joe Fernandez, Joost van Amersfoort, Josh Gordon, Josh Lipschultz, Josh Newlan, Ju yeong Ji, Kareem Mohamed, Kartikeya Badola, Kat Black, Katie Millican, Keelin McDonell, Kelvin Nguyen, Kiranbir Sodhia, Kish Greene, Lars Lowe Sjoesund, Lauren Usui, Laurent Sifre, Lena Heuermann, Leticia Lago, Lilly McNealus, Livio Baldini Soares, Logan Kilpatrick, Lucas Dixon, Luciano Martins, Machel Reid, Manvinder Singh, Mark Iverson, Martin Görner, Mat Velloso, Mateo Wirth, Matt Davidow, Matt Miller, Matthew Rahtz, Matthew Watson, Meg Risdal, Mehran Kazemi, Michael Moynihan, Ming Zhang, Minsuk Kahng, Minwoo Park, Mofi Rahman, Mohit Khatwani, Natalie Dao, Nenshad Bardoliwalla, Nesh Devanathan, Neta Dumai, Nilay Chauhan, Oscar Wahltinez, Pankil Botarda, Parker Barnes, Paul Barham, Paul Michel, Pengchong Jin, Petko Georgiev, Phil Culliton, Pradeep Kuppala, Ramona Comanescu, Ramona Merhej, Reena Jana, Reza Ardeshir Rokni, Rishabh Agarwal, Ryan Mullins, Samaneh Saadat, Sara Mc Carthy, Sarah Cogan, Sarah Perrin, Sébastien M. R. Arnold, Sebastian Krause,

Shengyang Dai, Shruti Garg, Shruti Sheth, Sue Ronstrom, Susan Chan, Timothy Jordan, Ting Yu, Tom Eccles, Tom Hennigan, Tomas Kocisky, Tulsee Doshi, Vihan Jain, Vikas Yadav, Vilobh Meshram, Vishal Dharmadhikari, Warren Barkley, Wei Wei, Wenming Ye, Woohyun Han, Woosuk Kwon, Xiang Xu, Zhe Shen, Zhitao Gong, Zichuan Wei, Victor Cotruta, Phoebe Kirk, Anand Rao, Minh Giang, Ludovic Peran, Tris Warkentin, Eli Collins, Joelle Barral, Zoubin Ghahramani, Raia Hadsell, D. Sculley, Jeanine Banks, Anca Dragan, Slav Petrov, Oriol Vinyals, Jeff Dean, Demis Hassabis, Koray Kavukcuoglu, Clement Farabet, Elena Buchatskaya, Sebastian Borgeaud, Noah Fiedel, Armand Joulin, Kathleen Kenealy, Robert Dadashi, and Alek Andreev. Gemma 2: Improving open language models at a practical size, 2024. URL `https://arxiv.org/abs/2408.00118`.

[53] Huayu Chen, Guande He, Lifan Yuan, Ganqu Cui, Hang Su, and Jun Zhu. Noise contrastive alignment of language models with explicit rewards. In Amir Globersons, Lester Mackey, Danielle Belgrave, Angela Fan, Ulrich Paquet, Jakub M. Tomczak, and Cheng Zhang, editors, *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024. URL `http://papers.nips.cc/paper_files/paper/2024/hash/d5a58d198afa370a3dff0e1ca4fe1802-Abstract-Conference.html`.

[54] Tomasz Korbak, Hady Elsahar, Germán Kruszewski, and Marc Dymetman. On reinforcement learning and distribution matching for fine-tuning language models with no catastrophic forgetting. In Sanmi Koyejo, S. Mohamed, A. Agarwal, Danielle Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, 2022. URL `http://papers.nips.cc/paper_files/paper/2022/hash/67496dfa96afddab795530cc7c69b57a-Abstract-Conference.html`.

[55] Anton Bakhtin, Yuntian Deng, Sam Gross, Myle Ott, Marc'Aurelio Ranzato, and Arthur Szlam. Residual energy-based models for text. *J. Mach. Learn. Res.*, 22:40:1–40:41, 2021. URL `http://jmlr.org/papers/v22/20-326.html`.

[56] Yuzhong Hong, Hanshan Zhang, Junwei Bao, Hongfei Jiang, and Yang Song. Energy-based preference model offers better offline alignment than the bradley-terry preference model. *ArXiv preprint*, abs/2412.13862, 2024. URL `https://arxiv.org/abs/2412.13862`.

[57] Stephen P. Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge University Press, 2014. ISBN 978-0-521-83378-3. doi: 10.1017/CBO9780511804441. URL `https://web.stanford.edu/%7Eboyd/cvxbook/`.

[58] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. URL `http://arxiv.org/abs/1412.6980`.

[59] Tijmen Tieleman and Geoffrey Hinton. Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. *COURSERA: Neural networks for machine learning*, 4(2): 26–31, 2012.

[60] Yunhao Tang, Zhaohan Daniel Guo, Zeyu Zheng, Daniele Calandriello, Rémi Munos, Mark Rowland, Pierre Harvey Richemond, Michal Valko, Bernardo Ávila Pires, and Bilal Piot. Generalized preference optimization: A unified approach to offline alignment. In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*. OpenReview.net, 2024. URL `https://openreview.net/forum?id=gu3nacA9AH`.

[61] Trevor Hastie. The elements of statistical learning: data mining, inference, and prediction, 2009.

[62] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d'Alché-Buc, Emily B. Fox, and Roman Garnett, editors, *Advances*

in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada, pages 8024–8035, 2019. URL https://proceedings.neurips.cc/paper/2019/hash/bdbca288fee7f92f2bfa9f7012727740-Abstract.html.

[63] Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*, 2023.

[64] Yanli Zhao, Andrew Gu, Rohan Varma, Liang Luo, Chien-Chin Huang, Min Xu, Less Wright, Hamid Shojanazeri, Myle Ott, Sam Shleifer, Alban Desmaison, Can Balioglu, Pritam Damania, Bernard Nguyen, Geeta Chauhan, Yuchen Hao, Ajit Mathews, and Shen Li. Pytorch fsdp: Experiences on scaling fully sharded data parallel, 2023. URL https://arxiv.org/abs/2304.11277.

[65] Leo Gao, Jonathan Tow, Baber Abbasi, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Alain Le Noac'h, Haonan Li, Kyle McDonell, Niklas Muennighoff, Chris Ociepa, Jason Phang, Laria Reynolds, Hailey Schoelkopf, Aviya Skowron, Lintang Sutawika, Eric Tang, Anish Thite, Ben Wang, Kevin Wang, and Andy Zou. A framework for few-shot language model evaluation, 2024. URL https://zenodo.org/records/12608602.

[66] stochasticboy321 (https://math.stackexchange.com/users/269063/stochasticboy321). Can i draw conclusion based on the following monte carlo estimate of entropy? Mathematics Stack Exchange, 2023. URL https://math.stackexchange.com/q/4803332. URL:https://math.stackexchange.com/q/4803332 (version: 2023-11-09).

[67] F.M. Dekking. *A Modern Introduction to Probability and Statistics: Understanding Why and How*. Springer Texts in Statistics. Springer, 2005. ISBN 9781852338961. URL https://books.google.co.kr/books?id=XLUMIlombgQC.

A Limitations

While our perspective offers novel insights, we acknowledge limitations for future work. First, Theorem 2.1, established from prior work [19], assumes sufficient data coverage and train-test generalization. Second, sampling inferior responses from the ideal π_l (Section 3.4) requires access to the unknown target policy π^*. This necessitates future work for approximating π^*, *e.g.*, iterative refinement using strong teacher models. Yet, the closed-form expression of π_l remains a useful conceptual target motivated from the assumption of Theorem 3.3. Finally, the connection between DID entropy and policy dynamics (Claim 4.1) is qualitative and based on information-theoretic intuition (Section E); despite experimental support (Section 4.3), a formal treatment would strengthen this aspect of our work.

B Related Work

Direct Preference Optimization Direct Preference Optimization (DPO) [3] has been widely used for aligning LMs with human preferences in a supervised fashion [4, 5]. Recent research has investigated the theoretical foundations of preference optimization: connecting preference optimization to distribution matching [54, 19, 11, 28, 20], and analyzing the optimization dynamics of log-likelihood displacement [35, 21, 22]. While Chen et al. [53] reinterpret the DPO objective from a noise contrastive estimation perspective, their approach is based on the optimal policy of the KL-regularized RL objective and leaves its justification open for discussion.

In this work, we complement prior works by offering an information-theoretic justification for the implicit reward structure in DPO, linking its optimality to the differential information captured by the preference data. Furthermore, we present a novel interpretation of log-likelihood displacement, showing its relationship to the entropy of the learned DID.

Residual Energy Based Models DPO's optimal policy $\pi^*(y) \propto \pi_{\text{ref}}(y)\exp(r(y)/\beta)$ [3] resembles Residual Energy Based Models (EBMs) [55, 56], where $p(y) \propto p_{\text{base}}(y)\exp(-E_{\text{residual}}(y))$. Residual EBMs can be equivalently considered as modeling the log-probability ratio $\log(p(y)/p_{\text{base}}(y))$ for generation tasks. Research in this area often focuses on practical aspects like efficient training, sampling techniques, and demonstrating downstream task improvements.

While sharing the structure of modeling a ratio relative to a base distribution, our work's focus differs significantly. We analyze the *information-theoretic* properties of π/π_{ref} in the context of *preference optimization*, understanding why the DPO objective is optimal for learning from preferences that encode the Differential Information Distribution (Theorem 3.4).

C Broader Impact

This work provides a theoretical framework for understanding preference optimization, a key technique for aligning Large Language Models (LLMs) with human preferences, centered on the concept of DIFFERENTIAL INFORMATION DISTRIBUTION (DID).

A deeper understanding of preference optimization, as offered by our DID framework, can lead to the development of more robust and reliable methods for aligning LLMs. This can contribute to creating AI systems that are safer, more helpful, and better adhere to desired human values. For instance, insights into DID entropy and its relation to policy dynamics (smoothing vs. reinforcement) could allow practitioners to tailor alignment strategies for specific tasks, potentially improving model performance in areas like general instruction-following or knowledge-intensive QA. Furthermore, understanding the conditions for DPO's optimality and the ideal generation of preference data could lead to more effective alignment processes.

While the primary goal is to improve alignment for beneficial outcomes, any advancement that makes LLMs more capable or easier to align also carries potential risks if misused. For instance, aligned models could potentially be used for sophisticated malicious purposes (*e.g.*, generating more convincing disinformation or manipulative content) if the human preferences they are aligned to are themselves harmful or if the models are repurposed. Our work focuses on the mechanism of alignment, not the ethical evaluation of the preferences themselves.

Continued research into the ethical implications of preference data, robust evaluation of aligned models, and governance frameworks for powerful AI systems will be crucial to ensure that advancements in alignment techniques are used responsibly and for societal benefit. This paper does not introduce a new deployable system but offers a theoretical lens that could inform the development of future systems.

D Probabilistic interpretation of Differential Information Distribution

This section provides a more detailed derivation and explanation of the probabilistic interpretation of the DIFFERENTIAL INFORMATION DISTRIBUTION (DID), as introduced in Section 3.1. Our goal is to illustrate the intuition that the DID $q_{\pi/\pi_{\text{ref}}}$ represents the distribution over token sequences y that embody the information needed to transform the reference policy π_{ref} into the target policy π through Bayesian conditioning.

D.1 Information as an abstract event

Let's begin by establishing a Bayesian framework to reason about information associated with sentences. Consider the space $\mathcal{Y}$ of all possible token sequences (*i.e.*, sentences), assuming a uniform prior distribution $P(Y = y) = 1/|\mathcal{Y}|$.

Now, imagine an abstract "event" or "property" X that can be associated with sentences. This event X represents some specific characteristic or information content. We can quantify the association between a sentence y and the property X using the conditional probability $P(X \mid Y = y)$. This term represents the likelihood that a given sentence y possesses the property X. Some examples include:

1. If X represents the property "is a mathematically correct statement", then:
   - $P(X \mid Y =$"`1+1=2`"$) = 1.0$

- $P(X \mid Y =$"`1+0=1`"$) = 1.0$
- $P(X \mid Y =$"`2+2=5`"$) = 0.0$

2. If X represents the property "is a safe statement", then the probabilities may be in the range of $[0, 1]$:
   - $P(X \mid Y =$"`Apples are red.`"$) = 0.99$
   - $P(X \mid Y =$"`Alcohol is good for relaxation.`"$) = 0.3$
   - $P(X \mid Y =$"`Let's promote violence!`"$) = 0.0$

A crucial assumption in our framework is that the property X is inherent to the sentence y itself, regardless of which language model might have generated it. For instance, the mathematical correctness or safeness of a sentence should not depend on whether it came from `Mistral7B-v0.3` or `Gemma2-9B`; it's a property of the content in y itself.

Formally, this means we assume that the event X is conditionally independent of the generating model (*e.g.*, π_{ref}) given the sentence $Y = y$:

$$P(X \mid Y = y, \pi_{\text{ref}}) = P(X \mid Y = y)$$

which is equivalent to stating that the joint probability factors as:

$$P(X, \pi_{\text{ref}} \mid Y = y) = P(X \mid Y = y)P(\pi_{\text{ref}} \mid Y = y).$$

This assumption allows us to treat $P(X \mid Y = y)$ as a property purely of the sentence y and the abstract information X.

D.2 Interpreting the distribution $P(Y = y \mid X)$

Given the likelihood $P(X \mid Y = y)$ that a sentence y possesses property X, what does the distribution $P(Y = y \mid X)$ represent? This is the distribution over sentences for which the property X holds. If X represents "mathematical correctness", then sampling from $P(Y = y \mid X)$ would yield mathematically correct statements.

We can derive this distribution using Bayes' theorem and our uniform prior $P(Y = y) = 1/|\mathcal{Y}|$:

$$\begin{aligned}
P(Y = y \mid X) &= \frac{P(X \mid Y = y)P(Y = y)}{P(X)} \\
&= \frac{P(X \mid Y = y)P(Y = y)}{\sum_{y' \in \mathcal{Y}} P(X \mid Y = y')P(Y = y')} \\
&= \frac{P(X \mid Y = y)(1/|\mathcal{Y}|)}{\sum_{y' \in \mathcal{Y}} P(X \mid Y = y')(1/|\mathcal{Y}|)} \\
&= \frac{P(X \mid Y = y)}{\sum_{y' \in \mathcal{Y}} P(X \mid Y = y')} \\
&\propto P(X \mid Y = y)
\end{aligned}$$

This confirms the intuition: the probability of sampling a sentence y under the condition X is directly proportional to the likelihood that sentence y possesses the property X. Sentences that strongly exhibit property X (high $P(X \mid Y = y)$) are more likely to be sampled from $P(Y = y \mid X)$.

Using the distribution over Y conditioned on X, we can characterize the uncertainty associated with X using the Shannon entropy $H(Y = y \mid X) = -\sum_y P(Y = y \mid X) \log P(Y = y \mid X)$. For instance, the information X associated with a very specific topic (*e.g.*, the birthplace of George Washington) can be characterized as being deterministic, since only a very small subset of $\mathcal{Y}$ would have sufficiently large values of $P(X \mid Y = y)$. On the other hand, the information X' associated with a wide range of concepts (*e.g.*, general instruction-following) can be associated with high uncertainty, since a relatively large subset of $\mathcal{Y}$ would have high $P(X' \mid Y = y)$, resulting in a spread-out distribution of $P(Y = y \mid X')$.

D.3 Information gain of policy updates

Now, let's shift our focus to the context of comparing two language models, π and π_{ref}, both assumed to have full support over $\mathcal{Y}$. We are interested in the *information gain* from updating these models.

Specifically, we want to characterize the information that, when incorporated into π_{ref}, transforms it into π.

Let's hypothesize that such information can be represented by an abstract event X, which we will call the DIFFERENTIAL INFORMATION from π_{ref} to π. We seek an X such that conditioning π_{ref} on X yields π. Formally, given $\pi_{\text{ref}}(y) = P(Y = y \mid \pi_{\text{ref}})$, we want X to satisfy:

$$\pi(y) = P(Y = y \mid \pi_{\text{ref}}, X)$$

Furthermore, we maintain our key assumption that this information X is intrinsic to the sentences, meaning it is conditionally independent of the initial model π_{ref} given the sentence y:

$$P(X \mid Y = y, \pi_{\text{ref}}) = P(X \mid Y = y)$$

Before proceeding, we should confirm that such an event X can always be constructed. The following lemma guarantees its existence.

Lemma D.1 (Existence of Differential Information)**.** *For any two probability distributions π, π_{ref} with full support on $\mathcal{Y}$, there exists an event X such that:*

$$\begin{cases} P(X \mid Y = y, \pi_{\text{ref}}) = P(X \mid Y = y) & \textit{(Conditional Independence)} \\ \pi(y) = P(Y = y \mid \pi_{\text{ref}}, X) & \textit{(Bayesian Update)} \end{cases}$$

Proof. Define X as a random variable that satisfies the conditional independence property $P(X \mid Y = y, \pi_{\text{ref}}) = P(X \mid Y = y)$. We need to show that we can define $P(X \mid Y = y)$ such that the Bayesian update rule holds.

First, choose a base probability $P(X \mid \pi_{\text{ref}})$ such that $0 < P(X \mid \pi_{\text{ref}}) < 1/\max_{y'} \left[\frac{\pi(y')}{\pi_{\text{ref}}(y')}\right]$. This ensures that the resulting conditional probability $P(X \mid Y = y)$ defined below is valid (*i.e.*, $0 \leq P(X \mid Y = y) \leq 1$). Now, define the likelihood of X given y as:

$$P(X \mid Y = y) := \frac{P(X \mid \pi_{\text{ref}})\pi(y)}{\pi_{\text{ref}}(y)}$$

Note that since π_{ref} has full support, we have $\pi_{\text{ref}}(y) > 0$. We must check if $P(X \mid Y = y) \leq 1$. This holds because by our choice of $P(X \mid \pi_{\text{ref}})$, we have:

$$\begin{aligned} P(X \mid Y = y) &= P(X \mid \pi_{\text{ref}}) \frac{\pi(y)}{\pi_{\text{ref}}(y)} \\ &\leq P(X \mid \pi_{\text{ref}}) \max_{y'} \left[\frac{\pi(y')}{\pi_{\text{ref}}(y')}\right] < 1. \end{aligned}$$

Now, using Bayes' rule we verify the Bayesian update condition:

$$\begin{aligned} P(Y = y \mid X, \pi_{\text{ref}}) &= \frac{P(X \mid Y = y, \pi_{\text{ref}})P(Y = y \mid \pi_{\text{ref}})}{P(X \mid \pi_{\text{ref}})} \quad \text{(Bayes' Rule)} \\ &= \frac{P(X \mid Y = y)\pi_{\text{ref}}(y)}{P(X \mid \pi_{\text{ref}})} \quad \text{(Conditional Independence)} \\ &= \frac{\left(\frac{P(X \mid \pi_{\text{ref}})\pi(y)}{\pi_{\text{ref}}(y)}\right)\pi_{\text{ref}}(y)}{P(X \mid \pi_{\text{ref}})} \quad \text{(Definition of } P(X \mid Y = y)\text{)} \\ &= \frac{P(X \mid \pi_{\text{ref}})\pi(y)}{P(X \mid \pi_{\text{ref}})} \\ &= \pi(y) \end{aligned}$$

Thus, we have constructed an event X satisfying both conditions. □

This lemma confirms that it is always possible to conceptualize the transformation from π_{ref} to π as a Bayesian update based on some underlying information X that satisfies our conditional independence assumption. We defined such X as the differential information that updates π_{ref} to π. Now, we connect this concept directly to the Differential Information Distribution (DID). The following theorem demonstrates that the distribution over token sequences conditioned on this differential information X is precisely the normalized ratio distribution $q_{\pi/\pi_{\text{ref}}}$.

Theorem (Likelihood Ratio Representation of Differential Information Distribution)**.** *For policies π, π_{ref} over $\mathcal{Y}$ with full support, the Differential Information Distribution (DID) from π_{ref} to π is equivalent to the normalized ratio distribution:*

$$P(Y = y \mid X) = \frac{\pi(y)/\pi_{\text{ref}}(y)}{Z} := q_{\pi/\pi_{\text{ref}}}(y).$$

where $Z = \sum_{y' \in \mathcal{Y}} \frac{\pi(y')}{\pi_{\text{ref}}(y')}$ is the partition function.

Proof. Let X be the event that satisfies Lemma D.1. The Bayes' Theorem states that:

$$\begin{aligned}
\pi(y) &= P(Y = y \mid \pi_{\text{ref}}, X) \\
&= \frac{P(X \mid Y = y, \pi_{\text{ref}}) P(Y = y \mid \pi_{\text{ref}})}{P(X \mid \pi_{\text{ref}})} \\
&= \frac{P(X \mid Y = y) P(Y = y \mid \pi_{\text{ref}})}{P(X \mid \pi_{\text{ref}})}
\end{aligned}$$

We thus have $\frac{\pi(y)}{\pi_{\text{ref}}(y)} = \frac{P(X|Y=y)}{P(X|\pi_{\text{ref}})}$. Now, consider the following relationship:

$$\begin{aligned}
\frac{\pi(y)}{\pi_{\text{ref}}(y) Z} &= \frac{\pi(y)/\pi_{\text{ref}}(y)}{\sum_{y' \in \mathcal{Y}} \pi(y')/\pi_{\text{ref}}(y')} \\
&= \frac{P(X \mid Y = y)/P(X \mid \pi_{\text{ref}})}{\sum_{y' \in \mathcal{Y}} P(X \mid Y = y')/P(X \mid \pi_{\text{ref}})} \\
&= \frac{P(X \mid Y = y)}{\sum_{y' \in \mathcal{Y}} P(X \mid Y = y')}
\end{aligned}$$

Since $P(Y = y)$ is a uniform distribution, we arrive at the relationship:

$$\begin{aligned}
P(Y = y \mid X) &= \frac{P(X \mid Y = y) P(Y = y)}{\sum_{y' \in \mathcal{Y}} P(X \mid Y = y') P(Y = y')} \\
&= \frac{P(X \mid Y = y)}{\sum_{y' \in \mathcal{Y}} P(X \mid Y = y')} \\
&= \frac{\pi(y)}{\pi_{\text{ref}}(y) Z} \\
&= q_{\pi/\pi_{\text{ref}}}(y)
\end{aligned}$$

□

In summary, this theorem validates the probabilistic interpretation of DID: the normalized ratio distribution $q_{\pi/\pi_{\text{ref}}}(y)$ is precisely the probability distribution over sentences $Y = y$ conditioned on the information X that updates π_{ref} into π. **Therefore, sampling from the ratio distribution $q_{\pi/\pi_{\text{ref}}}$ can be interpreted as sampling a sentence that embodies the differential information required to update π_{ref} into π via Bayesian conditioning.**

E A qualitative discussion of claim 4.1

In this section, we provide a qualitative argument that supports Claim 4.1. We restate our assumptions:

1. For any $y' \in \{y \in \mathcal{Y} \mid \pi_{\text{ref}}(y) \approx 0\}$, we have $\pi(y') \approx 0 \approx q_{\pi/\pi_{\text{ref}}}(y')$.
2. There is some reasonable upper-bound $c > 0$ such that $\mathbb{D}_{\text{KL}}[\pi_{\text{ref}} \parallel \pi] < c$.

The first condition assumes that π_{ref} is "reasonably" trained, in that for "meaningless" y' such that $\pi_{\text{ref}}(y') \approx 0$, we also have $\pi(y') \approx 0 \approx q_{\pi/\pi_{\text{ref}}}(y')$. The second condition states that π_{ref} and π should not differ significantly, such that $\mathbb{D}_{\text{KL}}[\pi_{\text{ref}} \parallel \pi]$ is bounded.

We now consider each cases of policy reinforcing and smoothing, and infer the relationship between $H(q_{\pi/\pi_{\text{ref}}})$ and $H(\pi_{\text{ref}})$.

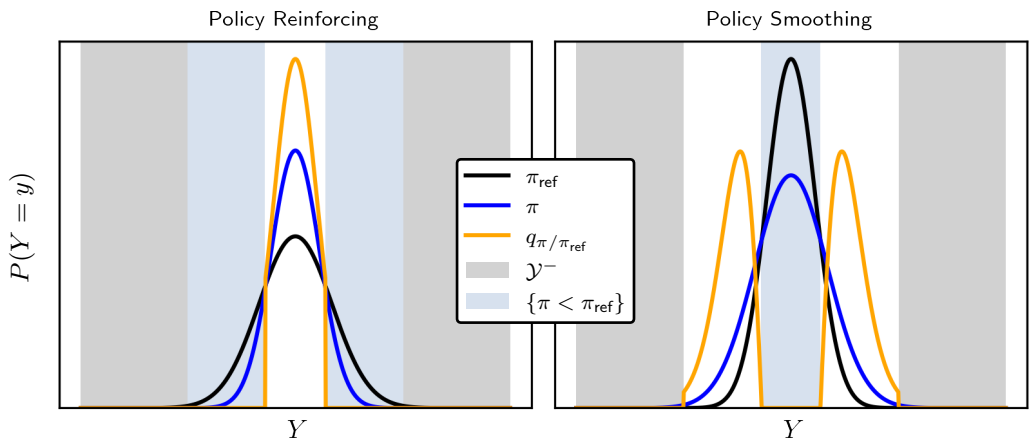

Figure 4: Illustration of policy reinforcement (*left*) and smoothing (*right*). The gray region corresponds to $\mathcal{Y}^- = \{y' \in \mathcal{Y} \mid \pi_{\text{ref}}(y') \approx 0\}$, and the light-blue region $\{\pi < \pi_{\text{ref}}\}$ corresponds to $\{\tilde{y} \in |\pi(\tilde{y}) < \pi_{\text{ref}}(\tilde{y})\}$. *Note: This plot serves only as an illustrative example, and does not represent the true DID* q_{π/π^*}.

Policy Reinforcing We first consider the case when the policy π reinforces its distribution with respect to the reference policy π_{ref}. If π reinforces the distribution of π_{ref}, then under the assumption of $\pi(y') \approx 0 \approx q_{\pi/\pi_{\text{ref}}}(y')$, samples with $\pi(\tilde{y}) < \pi_{\text{ref}}(\tilde{y})$ should satisfy $\frac{\pi(\tilde{y})}{\pi_{\text{ref}}(\tilde{y})} < 1 \approx \frac{\pi(y')}{\pi_{\text{ref}}(y')}$. Since $q_{\pi/\pi_{\text{ref}}}(\tilde{y}) < q_{\pi/\pi_{\text{ref}}}(y') \approx 0$, we expect $q_{\pi/\pi_{\text{ref}}}(y)$ to concentrate its probability mass towards samples with $\pi(y) > \pi_{\text{ref}}(y)$ and sufficient probability of $\pi_{\text{ref}}(y) > 0$. Thus, the number of samples y with sufficiently large $q_{\pi/\pi_{\text{ref}}}(y)$ is expected to be *less* than the number of samples with sufficiently large $\pi_{\text{ref}}(y)$. As a result, we expect the relationship: $H(q_{\pi/\pi_{\text{ref}}}) < H(\pi_{\text{ref}})$. We visualize this intuition as the left plot in Figure 4.

Policy Smoothing Now, consider the case when the policy π smooths its distribution with respect to π_{ref}. A key relation between $H(q_{\pi/\pi_{\text{ref}}})$ and $H(\pi_{\text{ref}})$ is the following:

$$H(q_{\pi/\pi_{\text{ref}}}) - H(\pi_{\text{ref}}) =$$
$$\mathbb{D}_{\text{KL}}[q_{\pi/\pi_{\text{ref}}} \parallel \pi] - \mathbb{D}_{\text{KL}}[q_{\pi/\pi_{\text{ref}}} \parallel \pi_{\text{ref}}] + \mathbb{D}_{\text{KL}}[\pi_{\text{ref}} \parallel q_{\pi/\pi_{\text{ref}}}] - \mathbb{D}_{\text{KL}}[\pi_{\text{ref}} \parallel \pi].$$

Since we have assumed that π and π_{ref} do not diverge significantly, we mainly expect the last two terms to dominate:

$$\left|\mathbb{D}_{\text{KL}}[q_{\pi/\pi_{\text{ref}}} \parallel \pi] - \mathbb{D}_{\text{KL}}[q_{\pi/\pi_{\text{ref}}} \parallel \pi_{\text{ref}}]\right| < \left|\mathbb{D}_{\text{KL}}[\pi_{\text{ref}} \parallel q_{\pi/\pi_{\text{ref}}}] - \mathbb{D}_{\text{KL}}[\pi_{\text{ref}} \parallel \pi]\right|.$$

See the right plot in Figure 4 for a visual intuition. When π smooths its distribution with respect to π_{ref}, we can expect $\mathbb{D}_{\text{KL}}[\pi_{\text{ref}} \parallel q_{\pi/\pi_{\text{ref}}}] > \mathbb{D}_{\text{KL}}[\pi_{\text{ref}} \parallel \pi]$. This results in the relationship: $H(q_{\pi/\pi_{\text{ref}}}) > H(\pi_{\text{ref}})$.

F Derivations and proofs

In this section we provide the detailed proofs supporting our theoretical findings. A key equivalence relationship that we utilize throughout this section is the following: for any $a, b, c > 0$, we have:

$$\begin{aligned} c \log a > c \log b &\iff \\ c\,(\log a - \log b) > 0 = \log 1 &\iff \\ \log \left(\frac{a}{b}\right)^c > \log 1 &\iff \\ a^c > b^c. & \end{aligned}$$

F.1 Proof for equivalence of preference optimization

Theorem (Preference vs. Distribution Matching [19]). *Let $\mathcal{D} = \{(y_w, y_l)\}$ be a sufficiently large preference dataset where the set of y_w and y_l covers $\mathcal{Y}$. Preference optimization on $\mathcal{D}$ is equivalent to fitting the reward-induced distribution $P(Y = y \mid r)$ on the implicit preference distribution $p^*(y)$:*

$$\begin{aligned}\max_r \mathbb{E}_{(y_w,y_l)\sim\mathcal{D}} \left[\log p(y_w \succ y_l \mid r)\right] &\iff \min_r \mathbb{E}_{(y_w,y_l)\sim\mathcal{D}} \left[\mathbb{D}_{\text{KL}}[p^*(y_w \succ y_l) \parallel p(y_w \succ y_l \mid r)]\right] \\ &\iff \min_r \mathbb{D}_{\text{KL}}[p^*(y) \parallel P(Y = y \mid r)].\end{aligned}$$

We restate the proof in [19] for reference.

Proof. For any two reward functions f_1 and f_2, the loss function $\mathbb{D}_{\text{KL}}[p(y_w \succ y_l \mid f_1) \parallel p(y_w \succ y_l \mid f_2)]$ is minimized if and only if $f_1(y) = f_2(y) + C$ for all $y \in \mathcal{Y}$ and for some constant C. If we let $f_1(y) = \log p^*(y)$, we have $p(y_w \succ y_l \mid f_1) = p^*(y_w \succ y_l)$. Now, set $f_2(y) = r(y)$ and the following relationship holds:

$$\begin{aligned}\mathbb{E}_{(y_w,y_l)\sim\mathcal{D}} \left[\mathbb{D}_{\text{KL}} \left[p(y_w \succ y_l \mid f_1) \parallel p(y_w \succ y_l \mid f_2)\right]\right] = 0 &\iff \\ \mathbb{E}_{(y_w,y_l)\sim\mathcal{D}} \left[\mathbb{D}_{\text{KL}} \left[p^*(y_w \succ y_l) \parallel p(y_w \succ y_l \mid r)\right]\right] = 0 &\iff \\ \forall y \in \mathcal{Y} : \log p^*(y) = r(y) + C &\iff \\ \forall y \in \mathcal{Y} : p^*(y) \propto \exp(r(y)) &\iff \\ \forall y \in \mathcal{Y} : P(Y = y \mid p^*) = P(Y = y \mid r) &\iff \\ \mathbb{D}_{\text{KL}} \left[p^*(y) \parallel P(Y = y \mid r)\right] = 0 &\end{aligned}$$

□

F.2 Proof for preferences encoding differential information

Theorem (Preferences Encoding Differential Information). *Let $\mathcal{D} = \{(y_w, y_l)\}$ be a preference data where $y_w \sim \pi_{\text{ref}}$ and $y_l \sim \pi_l$. Let π^* be the target policy. If the Differential Information Distribution between policies match up to an exponent $\beta > 0$:*

$$q_{\pi_{\text{ref}}/\pi_l}(y) \propto q_{\pi^*/\pi_{\text{ref}}}(y)^\beta$$

Then the preference probability $p^*(y_w \succ y_l)$ can be expressed as preferences induced by the DID:

$$p^*(y_w \succ y_l) = \sigma\left(\beta \log q_{\pi^*/\pi_{\text{ref}}}(y_w) - \beta \log q_{\pi^*/\pi_{\text{ref}}}(y_l)\right).$$

Proof. Given two samples y_1, y_2 in which $y_1 \neq y_2$, recall that we have labeled $y_1 \succ y_2$ if y_1 was sampled from π_{ref} and y_2 was sampled from π_l. Since we are assuming $y_1 \neq y_2$, there are a total two cases of y_1, y_2 each being sampled from either π_{ref} or π_l:

$$\begin{cases} y_1 \sim \pi_{\text{ref}}, y_2 \sim \pi_l \\ y_2 \sim \pi_{\text{ref}}, y_1 \sim \pi_l. \end{cases}$$

Thus, given two samples y_1, y_2, the probability of y_1 being preferred over y_2 is computed as the following:

$$\begin{aligned} p^*(y_1 \succ y_2) &= \frac{P(y_1 \sim \pi_{\text{ref}}, y_2 \sim \pi_l)}{P(y_1 \sim \pi_{\text{ref}}, y_2 \sim \pi_l) + P(y_2 \sim \pi_{\text{ref}}, y_1 \sim \pi_l)} \\ &= \frac{\pi_{\text{ref}}(y_1)\pi_l(y_2)}{\pi_{\text{ref}}(y_1)\pi_l(y_2) + \pi_{\text{ref}}(y_2)\pi_l(y_1)} \\ &= \frac{\frac{\pi_{\text{ref}}(y_1)}{\pi_l(y_1)}}{\frac{\pi_{\text{ref}}(y_1)}{\pi_l(y_1)} + \frac{\pi_{\text{ref}}(y_2)}{\pi_l(y_2)}} \\ &= \sigma\left(\log \frac{\pi_{\text{ref}}(y_1)}{\pi_l(y_1)} - \log \frac{\pi_{\text{ref}}(y_2)}{\pi_l(y_2)}\right) \\ &= \sigma\left(\log q_{\pi_{\text{ref}}/\pi_l}(y_1) - \log q_{\pi_{\text{ref}}/\pi_l}(y_2)\right) \\ &= \sigma\left(\beta \log q_{\pi^*/\pi_{\text{ref}}}(y_1) - \beta \log q_{\pi^*/\pi_{\text{ref}}}(y_2)\right) \end{aligned}$$

□

F.3 Proof for optimal reward for learning differential information

Theorem (Optimal Reward For Learning Differential Information)**.** *Let $\mathcal{D}$ be a preference dataset encoding the differential information required to learn the target policy, as defined in Theorem 3.3. We then have:*

$$\arg\max_{\pi} \mathbb{E}_{(y_w,y_l)\sim\mathcal{D}}[\log p(y_w \succ y_l \mid r)] = \arg\min_{\pi} \mathbb{D}_{\text{KL}}[\pi^*(y) \parallel \pi(y)]$$
$$\iff r(y) = \beta \log \frac{\pi(y)}{\pi_{\text{ref}}(y)} + C$$

Proof. The equivalence between preference optimization and distribution matching (Theorem 2.1) yields the following relationship:

$$\begin{aligned}\mathbb{E}_{(y_w,y_l)\sim\mathcal{D}}\left[\mathbb{D}_{\text{KL}}\left[p^*(y_w \succ y_l) \parallel p(y_w \succ y_l \mid r)\right]\right] = 0 \iff\\ \mathbb{E}_{(y_w,y_l)\sim\mathcal{D}}\left[\mathbb{D}_{\text{KL}}\left[p(y_w \succ y_l \mid r^*) \parallel p(y_w \succ y_l \mid r)\right]\right] = 0 \iff\\ \mathbb{D}_{\text{KL}}\left[P(Y = y \mid r^*) \parallel P(Y = y \mid r)\right] = 0\end{aligned}$$

where $r^* = \beta \log \frac{\pi^*}{\pi_{\text{ref}}}$. Now, observe the following relationship:

$$\begin{aligned}\mathbb{D}_{\text{KL}}\left[\pi^*(y) \parallel \pi(y)\right] = 0 &\iff\\ \forall y \in \mathcal{Y}, \pi^*(y) = \pi(y) &\iff\\ \forall y \in \mathcal{Y}, q_{\pi^*/\pi_{\text{ref}}}(y) = q_{\pi/\pi_{\text{ref}}}(y) &\iff\\ \forall y \in \mathcal{Y}, q_{\pi^*/\pi_{\text{ref}}}(y)^\beta = q_{\pi/\pi_{\text{ref}}}(y)^\beta &\iff\\ \mathbb{D}_{\text{KL}}\left[q_{\pi^*/\pi_{\text{ref}}}(y)^\beta \parallel q_{\pi/\pi_{\text{ref}}}(y)^\beta\right] = 0 &\iff\\ \mathbb{D}_{\text{KL}}\left[P(Y = y \mid r^*) \parallel q_{\pi/\pi_{\text{ref}}}(y)^\beta\right] = 0\end{aligned}$$

Where the last line follows from the fact that $r^* = \beta \log \frac{\pi^*}{\pi_{\text{ref}}}$.

Therefore, in order to have the following equivalence:

$$\begin{aligned}\mathbb{E}_{(y_w,y_l)\sim\mathcal{D}}\left[p^*(y_w \succ y_l) \parallel p(y_w \succ y_l \mid r)\right] = 0 \iff\\ \mathbb{D}_{\text{KL}}\left[\pi^*(y) \parallel \pi(y)\right] = 0\end{aligned}$$

we must have $\mathbb{D}_{\text{KL}}\left[P(Y = y \mid r) \parallel q_{\pi/\pi_{\text{ref}}}(y)^\beta\right] = 0$. In other words, we require:

$$\begin{aligned}\mathbb{D}_{\text{KL}}\left[P(Y = y \mid r) \parallel q_{\pi/\pi_{\text{ref}}}(y)^\beta\right] = 0 &\iff\\ \forall y \in \mathcal{Y}, P(Y = y \mid r) = q_{\pi/\pi_{\text{ref}}}(y)^\beta &\iff\\ \forall y \in \mathcal{Y}, \exp(r(y)) \propto (\frac{\pi(y)}{\pi_{\text{ref}}(y)})^\beta &\iff\\ \forall y \in \mathcal{Y}, r(y) = \beta \log \frac{\pi(y)}{\pi_{\text{ref}}(y)} + C\end{aligned}$$

for some constant C. □

F.4 Proof for differential information of log-margin ordered policies

Theorem (Differential Information of Log-Margin Ordered Policies)**.** *Consider three policies $\pi^*, \pi_{\text{ref}}, \pi_l$, and a preference data $\mathcal{D} = \{(y_w, y_l)\}$. Then the following statements are equivalent:*

1. ***Log-Margin Ordering*** *The log-margin of π_{ref} is sufficiently larger than that of π_l if and only if the log-margin of π^* is sufficiently larger than that of π_{ref}:*

$$\log \pi_{\text{ref}}(y_w) - \log \pi_{\text{ref}}(y_l) \gg \log \pi_l(y_w) - \log \pi_l(y_l)$$
$$\iff$$
$$\log \pi^*(y_w) - \log \pi^*(y_l) \gg \log \pi_{\text{ref}}(y_w) - \log \pi_{\text{ref}}(y_l)$$

2. ***Power-Law Structure of Differential Information*** *For each outcome y in the set $\{y_w, y_l\}$, a power-law exists between the DID from π_l to π_{ref} and the DID from π_{ref} to π^*. That is, there exists some $\beta > 0$ such that:*

$$q_{\pi_{\text{ref}}/\pi_l}(y) \propto q_{\pi^*/\pi_{\text{ref}}}(y)^{\beta}.$$

Proof. **Case 1** First, consider the log-margin relationship between π_{ref} and π_l:

$$\begin{aligned}
\forall (y_w, y_l) \in \mathcal{D}, \log \pi_{\text{ref}}(y_w) - \log \pi_{\text{ref}}(y_l) \gg \log \pi_l(y_w) - \log \pi_l(y_l) &\iff \\
\forall (y_w, y_l) \in \mathcal{D}, \log \frac{\pi_{\text{ref}}(y_w)}{\pi_l(y_w)} \gg \log \frac{\pi_{\text{ref}}(y_l)}{\pi_l(y_l)} &\iff \\
\forall (y_w, y_l) \in \mathcal{D}, \frac{\pi_{\text{ref}}(y_w)}{\pi_l(y_w)} \gg \frac{\pi_{\text{ref}}(y_l)}{\pi_l(y_l)} &\iff \\
\forall (y_w, y_l) \in \mathcal{D}, \frac{\frac{\pi_{\text{ref}}(y_w)}{\pi_l(y_w)}}{\frac{\pi_{\text{ref}}(y_w)}{\pi_l(y_w)} + \frac{\pi_{\text{ref}}(y_l)}{\pi_l(y_l)}} \approx 1 = p^*(y_w \succ y_l) &\iff \\
\mathbb{E}_{(y_w, y_l) \sim \mathcal{D}} \left[\mathbb{D}_{\text{KL}} \left[p^*(y_w \succ y_l) \parallel p(y_w \succ y_l \mid r') \right] \right] \approx 0 &
\end{aligned}$$

where $r'(y) = \log \frac{\pi_{\text{ref}}(y)}{\pi_l(y)}$.

Now consider the relationship between π_{ref} and π^*:

$$\begin{aligned}
\forall (y_w, y_l) \in \mathcal{D}, \log \pi^*(y_w) - \log \pi^*(y_l) \gg \log \pi_{\text{ref}}(y_w) - \log \pi_{\text{ref}}(y_l) &\iff \\
\forall (y_w, y_l) \in \mathcal{D}, \log \pi^*(y_w) - \log \pi_{\text{ref}}(y_w) \gg \log \pi^*(y_l) - \log \pi_{\text{ref}}(y_l) &\iff \\
\forall (y_w, y_l) \in \mathcal{D}, \log \frac{\pi^*(y_w)}{\pi_{\text{ref}}(y_w)} \gg \log \frac{\pi^*(y_l)}{\pi_{\text{ref}}(y_l)} &\iff \\
\forall (y_w, y_l) \in \mathcal{D}, \frac{\pi^*(y_w)}{\pi_{\text{ref}}(y_w)} \gg \frac{\pi^*(y_l)}{\pi_{\text{ref}}(y_l)} &\iff \\
\forall (y_w, y_l) \in \mathcal{D}, \frac{\frac{\pi^*(y_w)}{\pi_{\text{ref}}(y_w)}}{\frac{\pi^*(y_w)}{\pi_{\text{ref}}(y_w)} + \frac{\pi^*(y_l)}{\pi_{\text{ref}}(y_l)}} \approx 1 = p^*(y_w \succ y_l) &\iff \\
\mathbb{E}_{(y_w, y_l) \sim \mathcal{D}} \left[\mathbb{D}_{\text{KL}} \left[p^*(y_w \succ y_l) \parallel p(y_w \succ y_l \mid r^*) \right] \right] \approx 0 &
\end{aligned}$$

where $r^*(y) = \log \frac{\pi^*(y)}{\pi_{\text{ref}}(y)}$.

Therefore, for all $(y_w, y_l) \in \mathcal{D}$, in order to have an equivalence relation between the log-margins:

$$\begin{aligned}
&\log \pi_{\text{ref}}(y_w) - \log \pi_{\text{ref}}(y_l) \gg \log \pi_l(y_w) - \log \pi_l(y_l) \\
&\iff \\
&\log \pi^*(y_w) - \log \pi^*(y_l) \gg \log \pi_{\text{ref}}(y_w) - \log \pi_{\text{ref}}(y_l),
\end{aligned}$$

we must have $\mathbb{E}_{(y_w, y_l) \sim \mathcal{D}} \left[\mathbb{D}_{\text{KL}} \left[p(y_w \succ y_l \mid r') \parallel p(y_w \succ y_l \mid r^*) \right] \right] \approx 0$. This results in the following:

$$\begin{aligned}
\mathbb{E}_{(y_w, y_l) \sim \mathcal{D}} \left[\mathbb{D}_{\text{KL}} \left[p(y_w \succ y_l \mid r') \parallel p(y_w \succ y_l \mid r^*) \right] \right] \approx 0 &\iff \\
\mathbb{D}_{\text{KL}} \left[P(Y = y \mid r') \parallel P(Y = y \mid r^*) \right] \approx 0 &\iff \\
\forall y \in \mathcal{Y}, \frac{\pi_{\text{ref}}(y)}{\pi_l(y)} \propto \frac{\pi^*(y)}{\pi_{\text{ref}}(y)} &\iff \\
\forall y \in \mathcal{Y}, q_{\pi_{\text{ref}}/\pi_l}(y) \approx q_{\pi^*/\pi_{\text{ref}}}(y) &
\end{aligned}$$

where the first equivalence comes from applying Theorem 2.1.

Thus, we have shown that, for all $(y_w, y_l) \in \mathcal{D}$, if the following relation between log-margins holds:

$$\begin{aligned}
&\log \pi_{\text{ref}}(y_w) - \log \pi_{\text{ref}}(y_l) \gg \log \pi_l(y_w) - \log \pi_l(y_l) \\
&\iff \\
&\log \pi^*(y_w) - \log \pi^*(y_l) \gg \log \pi_{\text{ref}}(y_w) - \log \pi_{\text{ref}}(y_l)
\end{aligned}$$

then the DID between policies are approximately related by:

$$q_{\pi_{\text{ref}}/\pi_l}(y) \propto q_{\pi^*/\pi_{\text{ref}}}(y)^\beta$$

with $\beta = 1$.

Case 2 Now, consider the inverse case. Assume that there exists some $\beta > 0$ such that:

$$\forall y \in \mathcal{Y}, q_{\pi_{\text{ref}}/\pi_l}(y) \propto q_{\pi^*/\pi_{\text{ref}}}(y)^\beta.$$

Then an equivalence between the log-margin relations holds:

$$\begin{aligned}
\forall (y_w, y_l) \in \mathcal{D}, \log \pi_{\text{ref}}(y_w) - \log \pi_{\text{ref}}(y_l) \gg \log \pi_l(y_w) - \log \pi_l(y_l) \iff \\
\forall (y_w, y_l) \in \mathcal{D}, \log \frac{\pi_{\text{ref}}(y_w)}{\pi_l(y_w)} \gg \log \frac{\pi_{\text{ref}}(y_l)}{\pi_l(y_l)} \iff \\
\forall (y_w, y_l) \in \mathcal{D}, \frac{\pi_{\text{ref}}(y_w)}{\pi_l(y_w)} \gg \frac{\pi_{\text{ref}}(y_l)}{\pi_l(y_l)} \iff \\
\forall (y_w, y_l) \in \mathcal{D}, \frac{\frac{\pi_{\text{ref}}(y_w)}{\pi_l(y_w)}}{\frac{\pi_{\text{ref}}(y_w)}{\pi_l(y_w)} + \frac{\pi_{\text{ref}}(y_l)}{\pi_l(y_l)}} \approx 1 \iff \\
\forall (y_w, y_l) \in \mathcal{D}, \frac{q_{\pi_{\text{ref}}/\pi_l}(y_w)}{q_{\pi_{\text{ref}}/\pi_l}(y_w) + q_{\pi_{\text{ref}}/\pi_l}(y_l)} \approx 1 \iff \\
\forall (y_w, y_l) \in \mathcal{D}, \frac{q_{\pi^*/\pi_{\text{ref}}}(y_w)^\beta}{q_{\pi^*/\pi_{\text{ref}}}(y_w)^\beta + q_{\pi^*/\pi_{\text{ref}}}(y_l)^\beta} \approx 1 \iff \\
\forall (y_w, y_l) \in \mathcal{D}, \frac{\left(\frac{\pi^*(y_w)}{\pi_{\text{ref}}(y_w)}\right)^\beta}{\left(\frac{\pi^*(y_w)}{\pi_{\text{ref}}(y_w)}\right)^\beta + \left(\frac{\pi^*(y_l)}{\pi_{\text{ref}}(y_l)}\right)^\beta} \approx 1 \iff \\
\forall (y_w, y_l) \in \mathcal{D}, \left(\frac{\pi^*(y_w)}{\pi_{\text{ref}}(y_w)}\right)^\beta \gg \left(\frac{\pi^*(y_l)}{\pi_{\text{ref}}(y_l)}\right)^\beta \iff \\
\forall (y_w, y_l) \in \mathcal{D}, \frac{\pi^*(y_w)}{\pi_{\text{ref}}(y_w)} \gg \frac{\pi^*(y_l)}{\pi_{\text{ref}}(y_l)} \iff \\
\forall (y_w, y_l) \in \mathcal{D}, \log \pi^*(y_w) - \log \pi^*(y_l) \gg \log \pi_{\text{ref}}(y_w) - \log \pi_{\text{ref}}(y_l)
\end{aligned}$$

Thus, we have shown that for some $\beta > 0$, assuming

$$q_{\pi_{\text{ref}}/\pi_l}(y) \propto q_{\pi^*/\pi_{\text{ref}}}(y)^\beta$$

for all $y \in \mathcal{Y}$ implies an equivalence relation between the log-margins:

$$\begin{aligned}
&\log \pi_{\text{ref}}(y_w) - \log \pi_{\text{ref}}(y_l) \gg \log \pi_l(y_w) - \log \pi_l(y_l) \\
&\iff \\
&\log \pi^*(y_w) - \log \pi^*(y_l) \gg \log \pi_{\text{ref}}(y_w) - \log \pi_{\text{ref}}(y_l)
\end{aligned}$$

for all $(y_w, y_l) \in \mathcal{D}$.

□

G Relaxing the sufficient log-margin condition

From Theorem 3.5, we discussed how the power-law structure of differential information is inherently related to a log-margin ordering in policies. However, it should be noted that the theorem relies on the "sufficient difference" in the log-margins. How can we incorporate practical conditions at which the log-margin does not have a significant difference? We can relax this condition by introducing a scaling temperature $\alpha, \beta > 0$ so that the log-margins are further increased to a sufficient extent.

Theorem (Generalized Differential Information of Log-Margin Ordered Policies)**.** *Consider three policies $\pi^*, \pi_{\text{ref}}, \pi_l$, and a preference data $\mathcal{D} = \{(y_w, y_l)\}$. Then the following statements are equivalent:*

1. ***Log-Margin Ordering*** *There exists some $\alpha, \beta > 0$ such that the α-scaled log-margin of π_{ref} is sufficiently larger than that of π_l if and only if the β-scaled log-margin of π^* is sufficiently larger than that of π_{ref}:*

$$\begin{gathered}\alpha\left(\log \pi_{\text{ref}}(y_w) - \log \pi_{\text{ref}}(y_l)\right) \gg \alpha\left(\log \pi_l(y_w) - \log \pi_l(y_l)\right) \\ \Longleftrightarrow \\ \beta\left(\log \pi^*(y_w) - \log \pi^*(y_l)\right) \gg \beta\left(\log \pi_{\text{ref}}(y_w) - \log \pi_{\text{ref}}(y_l)\right)\end{gathered}$$

2. ***Power-Law Structure of Differential Information*** *For each outcome y in the set $\{y_w, y_l\}$, a power-law exists between the DID from π_l to π_{ref} and the DID from π_{ref} to π^*. That is, there exists some $\gamma > 0$ such that:*

$$q_{\pi_{\text{ref}}/\pi_l}(y) \propto q_{\pi^*/\pi_{\text{ref}}}(y)^\gamma.$$

The proof follows the same structure as Section F.4.

Proof. **Case 1** First, consider the log-margin relationship between π_{ref} and π_l:

$$\begin{aligned}\forall (y_w, y_l) \in \mathcal{D}, \alpha\left(\log \pi_{\text{ref}}(y_w) - \log \pi_{\text{ref}}(y_l)\right) \gg \alpha\left(\log \pi_l(y_w) - \log \pi_l(y_l)\right) &\iff \\ \forall (y_w, y_l) \in \mathcal{D}, \alpha \log \frac{\pi_{\text{ref}}(y_w)}{\pi_l(y_w)} \gg \alpha \log \frac{\pi_{\text{ref}}(y_l)}{\pi_l(y_l)} &\iff \\ \forall (y_w, y_l) \in \mathcal{D}, \left(\frac{\pi_{\text{ref}}(y_w)}{\pi_l(y_w)}\right)^\alpha \gg \left(\frac{\pi_{\text{ref}}(y_l)}{\pi_l(y_l)}\right)^\alpha &\iff \\ \forall (y_w, y_l) \in \mathcal{D}, \frac{\left(\frac{\pi_{\text{ref}}(y_w)}{\pi_l(y_w)}\right)^\alpha}{\left(\frac{\pi_{\text{ref}}(y_w)}{\pi_l(y_w)}\right)^\alpha + \left(\frac{\pi_{\text{ref}}(y_l)}{\pi_l(y_l)}\right)^\alpha} \approx 1 = p^*(y_w \succ y_l) &\iff \\ \mathbb{E}_{(y_w, y_l) \sim \mathcal{D}}\left[\mathbb{D}_{\text{KL}}\left[p^*(y_w \succ y_l) \parallel p(y_w \succ y_l \mid r')\right]\right] \approx 0 &\end{aligned}$$

where $r'(y) = \alpha \log \frac{\pi_{\text{ref}}(y)}{\pi_l(y)}$.

Now consider the relationship between π_{ref} and π^*:

$$\begin{aligned}\forall (y_w, y_l) \in \mathcal{D}, \beta\left(\log \pi^*(y_w) - \log \pi^*(y_l)\right) \gg \beta\left(\log \pi_{\text{ref}}(y_w) - \log \pi_{\text{ref}}(y_l)\right) &\iff \\ \forall (y_w, y_l) \in \mathcal{D}, \beta\left(\log \pi^*(y_w) - \log \pi_{\text{ref}}(y_w)\right) \gg \beta\left(\log \pi^*(y_l) - \log \pi_{\text{ref}}(y_l)\right) &\iff \\ \forall (y_w, y_l) \in \mathcal{D}, \beta \log \frac{\pi^*(y_w)}{\pi_{\text{ref}}(y_w)} \gg \beta \log \frac{\pi^*(y_l)}{\pi_{\text{ref}}(y_l)} &\iff \\ \forall (y_w, y_l) \in \mathcal{D}, \left(\frac{\pi^*(y_w)}{\pi_{\text{ref}}(y_w)}\right)^\beta \gg \left(\frac{\pi^*(y_l)}{\pi_{\text{ref}}(y_l)}\right)^\beta &\iff \\ \forall (y_w, y_l) \in \mathcal{D}, \frac{\left(\frac{\pi^*(y_w)}{\pi_{\text{ref}}(y_w)}\right)^\beta}{\left(\frac{\pi^*(y_w)}{\pi_{\text{ref}}(y_w)}\right)^\beta + \left(\frac{\pi^*(y_l)}{\pi_{\text{ref}}(y_l)}\right)^\beta} \approx 1 = p^*(y_w \succ y_l) &\iff \\ \mathbb{E}_{(y_w, y_l) \sim \mathcal{D}}\left[\mathbb{D}_{\text{KL}}\left[p^*(y_w \succ y_l) \parallel p(y_w \succ y_l \mid r^*)\right]\right] \approx 0 &\end{aligned}$$

where $r^*(y) = \beta \log \frac{\pi^*(y)}{\pi_{\text{ref}}(y)}$.

For all $(y_w, y_l) \in \mathcal{D}$, in order to have the following equivalence relation:

$$\begin{aligned}&\alpha\left(\log \pi_{\text{ref}}(y_w) - \log \pi_{\text{ref}}(y_l)\right) \gg \alpha\left(\log \pi_l(y_w) - \log \pi_l(y_l)\right) \\ \iff\ &\beta\left(\log \pi^*(y_w) - \log \pi^*(y_l)\right) \gg \beta\left(\log \pi_{\text{ref}}(y_w) - \log \pi_{\text{ref}}(y_l)\right),\end{aligned}$$

we must have $\mathbb{E}_{(y_w,y_l)\sim\mathcal{D}}\left[\mathbb{D}_{\text{KL}}\left[p(y_w \succ y_l \mid r') \parallel p(y_w \succ y_l \mid r^*)\right]\right] \approx 0$. This results in the following:

$$\begin{aligned}
\mathbb{E}_{(y_w,y_l)\sim\mathcal{D}}\left[\mathbb{D}_{\text{KL}}\left[p(y_w \succ y_l \mid r') \parallel p(y_w \succ y_l \mid r^*)\right]\right] \approx 0 &\iff \\
\mathbb{D}_{\text{KL}}\left[P(Y = y \mid r') \parallel P(Y = y \mid r^*)\right] \approx 0 &\iff \\
\forall y \in \mathcal{Y}, \left(\frac{\pi_{\text{ref}}(y)}{\pi_l(y)}\right)^{\alpha} \propto \left(\frac{\pi^*(y)}{\pi_{\text{ref}}(y)}\right)^{\beta} &\iff \\
\forall y \in \mathcal{Y}, q_{\pi_{\text{ref}}/\pi_l}(y) \approx q_{\pi^*/\pi_{\text{ref}}}(y)^{\frac{\beta}{\alpha}} &\iff \\
\forall y \in \mathcal{Y}, q_{\pi_{\text{ref}}/\pi_l}(y) \approx q_{\pi^*/\pi_{\text{ref}}}(y)^{\gamma}&
\end{aligned}$$

where $\gamma = \frac{\beta}{\alpha} > 0$. Thus, for all $(y_w, y_l) \in \mathcal{D}$, if the following relation between log-margins holds:

$$\begin{aligned}
&\alpha\left(\log \pi_{\text{ref}}(y_w) - \log \pi_{\text{ref}}(y_l)\right) \gg \alpha\left(\log \pi_l(y_w) - \log \pi_l(y_l)\right) \\
&\iff \\
&\beta\left(\log \pi^*(y_w) - \log \pi^*(y_l)\right) \gg \beta\left(\log \pi_{\text{ref}}(y_w) - \log \pi_{\text{ref}}(y_l)\right)
\end{aligned}$$

then the DID between policies are approximately related by:

$$q_{\pi_{\text{ref}}/\pi_l}(y) \propto q_{\pi^*/\pi_{\text{ref}}}(y)^{\gamma}$$

with $\gamma = \frac{\beta}{\alpha} > 0$.

Case 2 Now, consider the inverse case. Assume that there exists some $\gamma > 0$ such that:

$$\forall y \in \mathcal{Y}, q_{\pi_{\text{ref}}/\pi_l}(y) \propto q_{\pi^*/\pi_{\text{ref}}}(y)^{\gamma}.$$

Then, for all $(y_w, y_l) \in \mathcal{D}$, an equivalence between the log-margins holds:

$$\begin{aligned}
\alpha\left(\log \pi_{\text{ref}}(y_w) - \log \pi_{\text{ref}}(y_l)\right) \gg \alpha\left(\log \pi_l(y_w) - \log \pi_l(y_l)\right) &\iff \\
\alpha \log \frac{\pi_{\text{ref}}(y_w)}{\pi_l(y_w)} \gg \alpha \log \frac{\pi_{\text{ref}}(y_l)}{\pi_l(y_l)} &\iff \\
\left(\frac{\pi_{\text{ref}}(y_w)}{\pi_l(y_w)}\right)^{\alpha} \gg \left(\frac{\pi_{\text{ref}}(y_l)}{\pi_l(y_l)}\right)^{\alpha} &\iff \\
\frac{\left(\frac{\pi_{\text{ref}}(y_w)}{\pi_l(y_w)}\right)^{\alpha}}{\left(\frac{\pi_{\text{ref}}(y_w)}{\pi_l(y_w)}\right)^{\alpha} + \left(\frac{\pi_{\text{ref}}(y_l)}{\pi_l(y_l)}\right)^{\alpha}} \approx 1 &\iff \\
\frac{q_{\pi_{\text{ref}}/\pi_l}(y_w)^{\alpha}}{q_{\pi_{\text{ref}}/\pi_l}(y_w)^{\alpha} + q_{\pi_{\text{ref}}/\pi_l}(y_l)^{\alpha}} \approx 1 &\iff \\
\frac{q_{\pi^*/\pi_{\text{ref}}}(y_w)^{\alpha\gamma}}{q_{\pi^*/\pi_{\text{ref}}}(y_w)^{\alpha\gamma} + q_{\pi^*/\pi_{\text{ref}}}(y_l)^{\alpha\gamma}} \approx 1 &\iff \\
\frac{\left(\frac{\pi^*(y_w)}{\pi_{\text{ref}}(y_w)}\right)^{\alpha\gamma}}{\left(\frac{\pi^*(y_w)}{\pi_{\text{ref}}(y_w)}\right)^{\alpha\gamma} + \left(\frac{\pi^*(y_l)}{\pi_{\text{ref}}(y_l)}\right)^{\alpha\gamma}} \approx 1 &\iff \\
\left(\frac{\pi^*(y_w)}{\pi_{\text{ref}}(y_w)}\right)^{\alpha\gamma} \gg \left(\frac{\pi^*(y_l)}{\pi_{\text{ref}}(y_l)}\right)^{\alpha\gamma} &\iff \\
\left(\frac{\pi^*(y_w)}{\pi^*(y_l)}\right)^{\alpha\gamma} \gg \left(\frac{\pi_{\text{ref}}(y_w)}{\pi_{\text{ref}}(y_l)}\right)^{\alpha\gamma} &\iff \\
\alpha\gamma\left(\log \pi^*(y_w) - \log \pi^*(y_l)\right) \gg \alpha\gamma\left(\log \pi_{\text{ref}}(y_w) - \log \pi_{\text{ref}}(y_l)\right) &\iff \\
\beta\left(\log \pi^*(y_w) - \log \pi^*(y_l)\right) \gg \beta\left(\log \pi_{\text{ref}}(y_w) - \log \pi_{\text{ref}}(y_l)\right)&
\end{aligned}$$

where $\beta = \alpha\gamma > 0$.

Thus, we have shown that for some $\gamma > 0$, assuming

$$q_{\pi_{\text{ref}}/\pi_l}(y) \propto q_{\pi^*/\pi_{\text{ref}}}(y)^{\gamma}$$

for all $y \in \mathcal{Y}$ implies an equivalence relation between the log-margins:

$$\alpha \left(\log \pi_{\text{ref}}(y_w) - \log \pi_{\text{ref}}(y_l)\right) \gg \alpha \left(\log \pi_l(y_w) - \log \pi_l(y_l)\right)$$
$$\iff$$
$$\beta \left(\log \pi^*(y_w) - \log \pi^*(y_l)\right) \gg \beta \left(\log \pi_{\text{ref}}(y_w) - \log \pi_{\text{ref}}(y_l)\right)$$

for all $(y_w, y_l) \in \mathcal{D}$ and for some $\alpha, \beta > 0$.

□

H DPO-Projected Gradient (DPO-PG)

While several variants of DPO have been proposed to address log-likelihood displacement [35, 29], we observed that these methods exhibit instability when scaled to large datasets (approximately 100,000 samples) and trained over multiple epochs (≤ 5). An effective baseline to vanilla DPO should ideally increase $\log \pi(y_w)$ while reducing the DPO loss to a comparable extent. Without achieving a comparable reduction in the DPO loss, it becomes difficult to argue that this method has properly learned the underlying preference distribution.

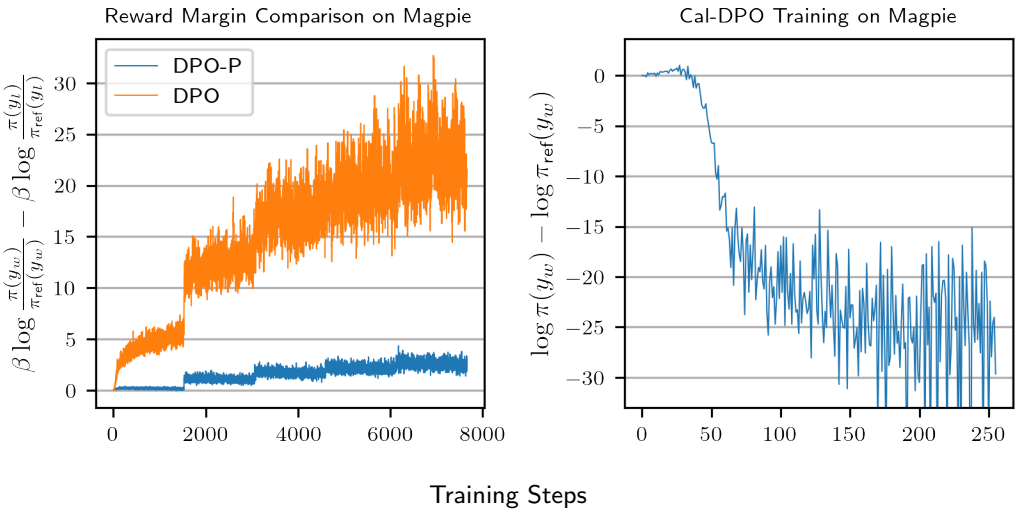

Figure 5: Testing DPO-P [35] and Cal-DPO [29] on Magpie dataset. We found that DPO-P fails to optimize the log-margin as effectively as vanilla DPO. Meanwhile, we found that Cal-DPO is unstable at mitigating log-likelihood displacement.

Despite extensive experiments with various hyper-parameters, we failed to find a setting for both DPO-P [35] and Cal-DPO [29] which met this criterion reliably (Figure 5). This motivated us to design a new method that stably mitigates log-likelihood displacement while ensuring consistent optimization of the DPO loss. The result is DPO-PG, a method grounded in projected gradient descent.

As its name implies, DPO-Projected Gradient (DPO-PG) leverages projected gradient descent [57] to reinforce the policy distribution while optimizing the DPO objective. Specifically, it increases $\log \pi(y_w)$ while maintaining or decreasing $\log \pi(y_l)$. Due to the log-margin term in the DPO loss, DPO-PG is guaranteed to reduce the DPO loss under sufficiently small step sizes (Corollary H.5.1).

The primary advantage of using DPO-PG over other DPO variants (*e.g.*, DPO-P [35], Cal-DPO [29]) is that DPO-PG can reliably optimize the DPO loss while increasing both $\log \pi(y_w)$ and the log-margin $\log \pi(y_w) - \log \pi(y_l)$, all without introducing additional hyper-parameters (*i.e.*, DPO-PG is hyper-parameter free). We plot the log-likelihood change of DPO-PG at Figure 8, and the measured DPO loss of DPO-PG at Figure 9.

Definition H.1. DPO-Projected Gradient (DPO-PG): $\theta_{k+1} = \theta_k - \eta(\nabla L(y_w) - \frac{\alpha}{||\nabla L(y_l)||_2^2} \nabla L(y_l))$, where θ_k denotes the parameter at training step k, $\eta > 0$ is the step size, and $\alpha = \max(0, \nabla L(y_w) \cdot \nabla L(y_l))$.

Here, $L(y)$ is the **negative** log-likelihood loss: $-\frac{1}{M}\sum_{i=1}^{M}\log\pi(y^{(i)})$, where M is the batch size, and $y^{(i)}$ is the i-th element in the batch. In practice, when using any non-SGD optimizer (*e.g.*, Adam [58], RMSprop [59]), we set the parameters' gradient as $\nabla L(y_w) - \frac{\alpha}{||\nabla L(y_l)||_2^2}\nabla L(y_l)$ and update its parameters following the optimizer's algorithm. Similarly, when applying gradient-clipping, we clip the L2 norm of $\nabla L(y_w) - \frac{\alpha}{||\nabla L(y_l)||_2^2}\nabla L(y_l)$.

We now show that DPO-PG decreases $L(y_w)$ while maintaining or increasing $L(y_l)$, for sufficiently small step sizes. We first start with the definition of *descent direction* [57]:

Definition H.2. For some function $f:\mathbb{R}^D\to\mathbb{R}$, and a point $\theta\in\mathbb{R}^D$, a direction $\Delta\theta\in\mathbb{R}^D$ is called a *descent direction* if there exists $\bar{\alpha}>0$ such that $f(\theta+\alpha\Delta\theta)<f(\theta), \forall\alpha\in(0,\bar{\alpha})$.

The following well-known lemma allows one to verify whether some direction is a descent direction of a differentiable objective function f [57].

Lemma H.3. *Consider a point $\theta\in\mathbb{R}^D$. Any direction $\Delta\theta\in\mathbb{R}^D$ satisfying $\Delta\theta\cdot\nabla f(\theta)<0$ is a descent direction.*

We now analyze the properties of the update direction of DPO-PG: $\Delta\theta = \theta_{k+1}-\theta_k = -\eta\{\nabla L(y_w) - \frac{\max(0,\nabla L(y_w)\cdot\nabla L(y_l))}{||\nabla L(y_l)||_2^2}\nabla L(y_l)\}$. The following theorem states that DPO-PG increases the log-likelihood of y_w.

Theorem H.4. *$\Delta\theta$ is a descent direction of the negative log-likelihood of the preferred responses $-\frac{1}{M}\Sigma_{i=1}^{M}\log\pi(y_w^{(i)}) = L(y_w)$.*

Proof. Regardless of the sign value of $\nabla L(y_w)\cdot\nabla L(y_l)$, we can show that $\Delta\theta\cdot\nabla L(y_w)<0$.

Case 1: If we have $\nabla L(y_w)\cdot\nabla L(y_l)>0$, it follows that

$$\begin{aligned}\Delta\theta\cdot\nabla L(y_w) &= -\eta\{||\nabla L(y_w)||_2^2 - \frac{\nabla L(y_w)\cdot\nabla L(y_l)}{||\nabla L(y_l)||_2^2}\nabla L(y_l)\cdot\nabla L(y_w)\}\\ &= -\frac{\eta}{||\nabla L(y_l)||_2^2}\{||\nabla L(y_w)||_2^2\cdot||\nabla L(y_l)||_2^2 - (\nabla L(y_l)\cdot\nabla L(y_w))^2\}<0\end{aligned}$$

where the last inequality follows from the Cauchy-Schwarz inequality: $||\nabla L(y_w)||_2^2\cdot||\nabla L(y_l)||_2^2 > ||\nabla L(y_w)\cdot\nabla L(y_l)||_2^2>0$.

Case 2: Otherwise, we have $\nabla L(y_w)\cdot\nabla L(y_l)\le 0$ and it follows that $\Delta\theta\cdot\nabla L(y_w) = -\eta||\nabla L(y_w)||_2^2<0$. □

Conversely, we can show that DPO-PG decreases or maintains the log-likelihood of y_l.

Theorem H.5. *$\Delta\theta$ is **not** a descent direction of the negative log-likelihood of the dis-preferred responses: $-\frac{1}{M}\Sigma_{i=1}^{M}\log\pi(y_l^{(i)}) = L(y_l)$*

Proof. $\Delta\theta\cdot\nabla L(y_l) = -\eta\{\nabla L(y_w)\cdot\nabla L(y_l) - \max(0,\nabla L(y_w)\cdot\nabla L(y_l))\}\ge 0$ In other words, $\Delta\theta$ is either orthogonal or an ascent direction to the negative log-likelihood of the dis-preferred responses y_l. □

Meanwhile, various offline preference optimization methods can be characterized by solving the following objective [60]:

$$\arg\min_{\theta}\mathbb{E}_{(y_w,y_l)\sim\mu}\left[f(\beta\log\frac{\pi_\theta(y_w)}{\pi_{\text{ref}}(y_w)} - \beta\log\frac{\pi_\theta(y_l)}{\pi_{\text{ref}}(y_l)})\right]$$

where f denotes any valid supervised binary classification loss function [61]. As a consequence of H.4 and H.5, DPO-PG is able to train a policy to learn human preferences [60].

Corollary H.5.1. *For any valid supervised binary classification loss function f with $f'(\cdot)<0$, $\Delta\theta$ is a descent direction to the loss $f(\beta\cdot(\log\frac{\pi_\theta(y_w)}{\pi_{\text{ref}}(y_w)} - \log\frac{\pi_\theta(y_l)}{\pi_{\text{ref}}(y_l)}))$ where $\beta>0$.*

Proof.

$$\Delta\theta \cdot \nabla f(\beta \log \frac{\pi_\theta(y_w)}{\pi_{\text{ref}}(y_w)} - \beta \log \frac{\pi_\theta(y_l)}{\pi_{\text{ref}}(y_l)})$$
$$= \Delta\theta \cdot \beta f' \left(\beta \log \frac{\pi_\theta(y_w)}{\pi_{\text{ref}}(y_w)} - \beta \log \frac{\pi_\theta(y_l)}{\pi_{\text{ref}}(y_l)} \right) (\nabla L(y_w) - \nabla L(y_l))$$
$$= \underbrace{\beta f' \left(\beta \log \frac{\pi_\theta(y_w)}{\pi_{\text{ref}}(y_w)} - \beta \log \frac{\pi_\theta(y_l)}{\pi_{\text{ref}}(y_l)} \right)}_{f'(\cdot)<0} (\underbrace{\Delta\theta \cdot \nabla L(y_w)}_{>0} - \underbrace{\Delta\theta \cdot \nabla L(y_l)}_{\leq 0})$$

From Lemma H.4, we have $\Delta\theta \cdot \nabla L(y_w) > 0$, and from Lemma H.5, we have $\Delta\theta \cdot \nabla L(y_l) \leq 0$. Thus, we have $(\Delta\theta \cdot \nabla L(y_w) - \Delta\theta \cdot \nabla L(y_l)) > 0$. Since $\beta > 0$ and $\beta f'(\cdot) < 0$, it follows that $\Delta\theta \cdot \nabla f(\beta \log \frac{\pi_\theta(y_w)}{\pi_{\text{ref}}(y_w)} - \beta \log \frac{\pi_\theta(y_l)}{\pi_{\text{ref}}(y_l)}) < 0$. □

To summarize, Lemma H.4 ensures that only $\log \pi(y_w)$ (and not $\log \pi(y_l)$) increases during training, for sufficiently small step sizes. This ensures policy reinforcement with respect to π_{ref}. Corollary H.5.1 further ensures that DPO-PG optimizes the DPO loss, too. We empirically validate the log-likelihood dynamics in Figure 8, and verify that DPO-PG successfully optimizes the DPO loss in Figure 9.

I Additional experimental results

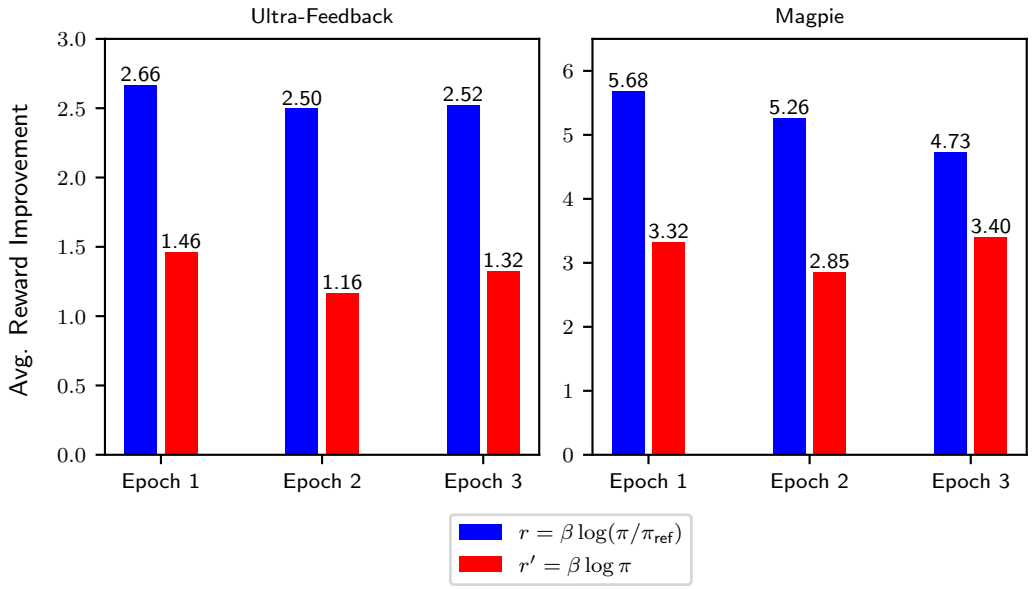

Figure 6: Comparison of expected rewards on 1,000 test prompts (Ultra-Feedback, Magpie) of policies trained with standard DPO $r = \beta \log(\pi/\pi_{\text{ref}})$ vs. $r' = \beta \log \pi$. Overall, the log-ratio reward parameterization consistently yields policies with higher expected rewards. This supports that the preferences of instruction-following datasets tend to encode differential information, rather than directly encoding the target policy. We estimate the reward of completions generated by each model using `Skywork/Skywork-Reward-Gemma-2-27B-v0.2` [32]. At each epoch, we compute the average reward across all prompts, and record the maximum average reward observed across $\beta \in \{0.2, 0.1, 0.05, 0.02\}$. For each estimated reward, we subtract the average reward of π_{ref} to illustrate the relative improvement from the reference policy (*i.e.*, we specify $\mathbb{E}_{y\sim\pi}[R(y)] - \mathbb{E}_{y_{\text{ref}}\sim\pi_{\text{ref}}}[R(y_{\text{ref}})]$, where $R : \mathcal{Y} \to \mathbb{R}$ is the reward model.).

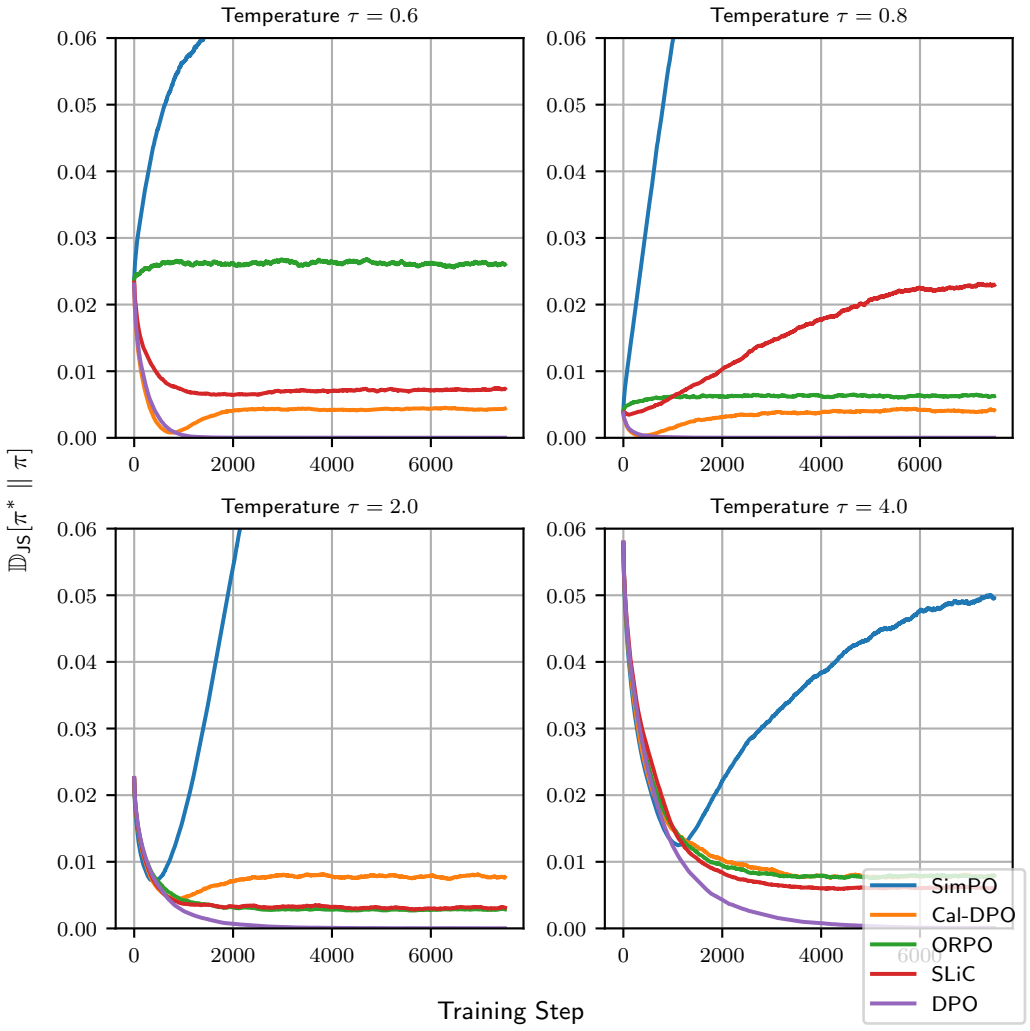

Figure 7: Validation of Theorem 3.4: Comparison of $\mathbb{D}_{\mathrm{JS}}[\pi^* \parallel \pi]$ during training using different objectives on the synthetic dataset. Standard DPO ($r = \log(\pi/\pi_{\mathrm{ref}})$, purple) consistently minimizes the JS divergence to the target policy π^*. This demonstrates its optimality under conditions where preferences encode DID.

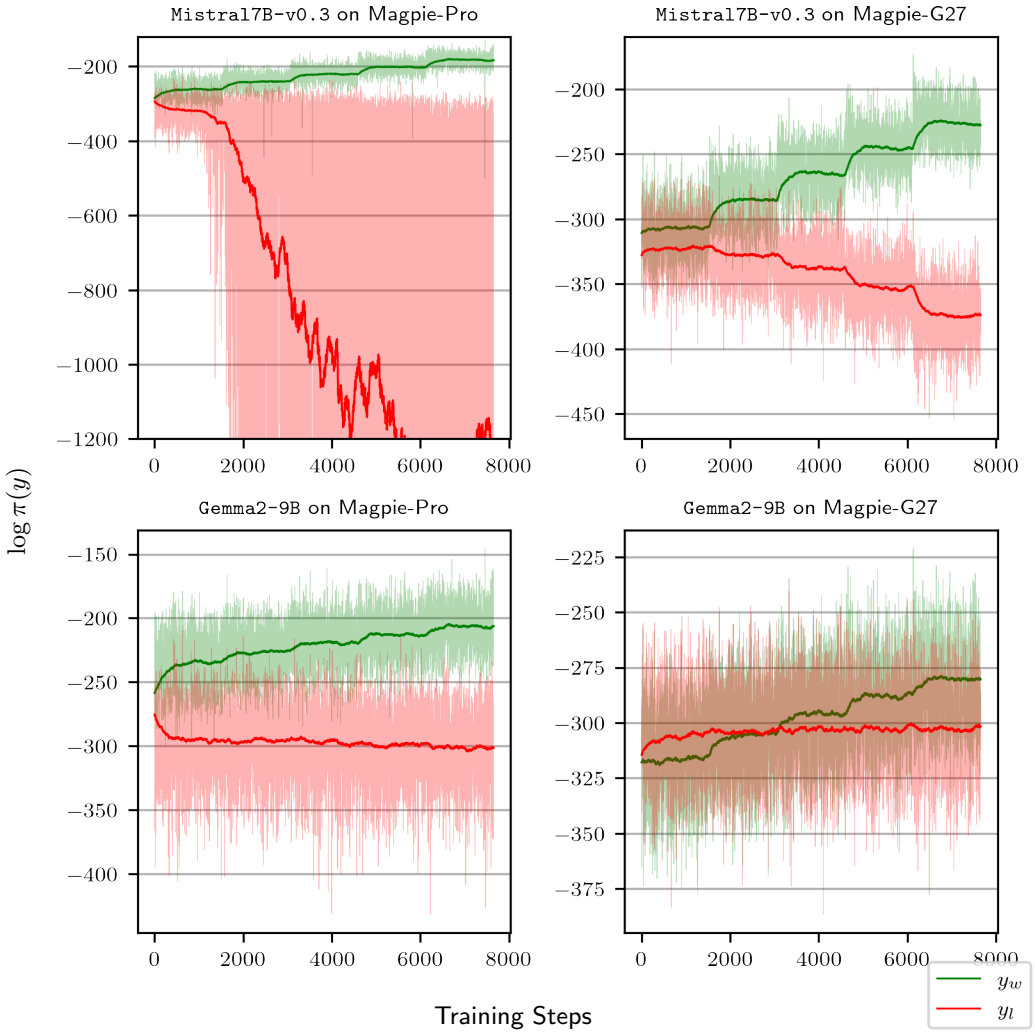

Figure 8: Log-likelihood change for DPO-PG across all experimental configurations. Overall, DPO-PG consistently increases the log-likelihood of y_w, while decreasing or maintaining the log-likelihood of y_l.

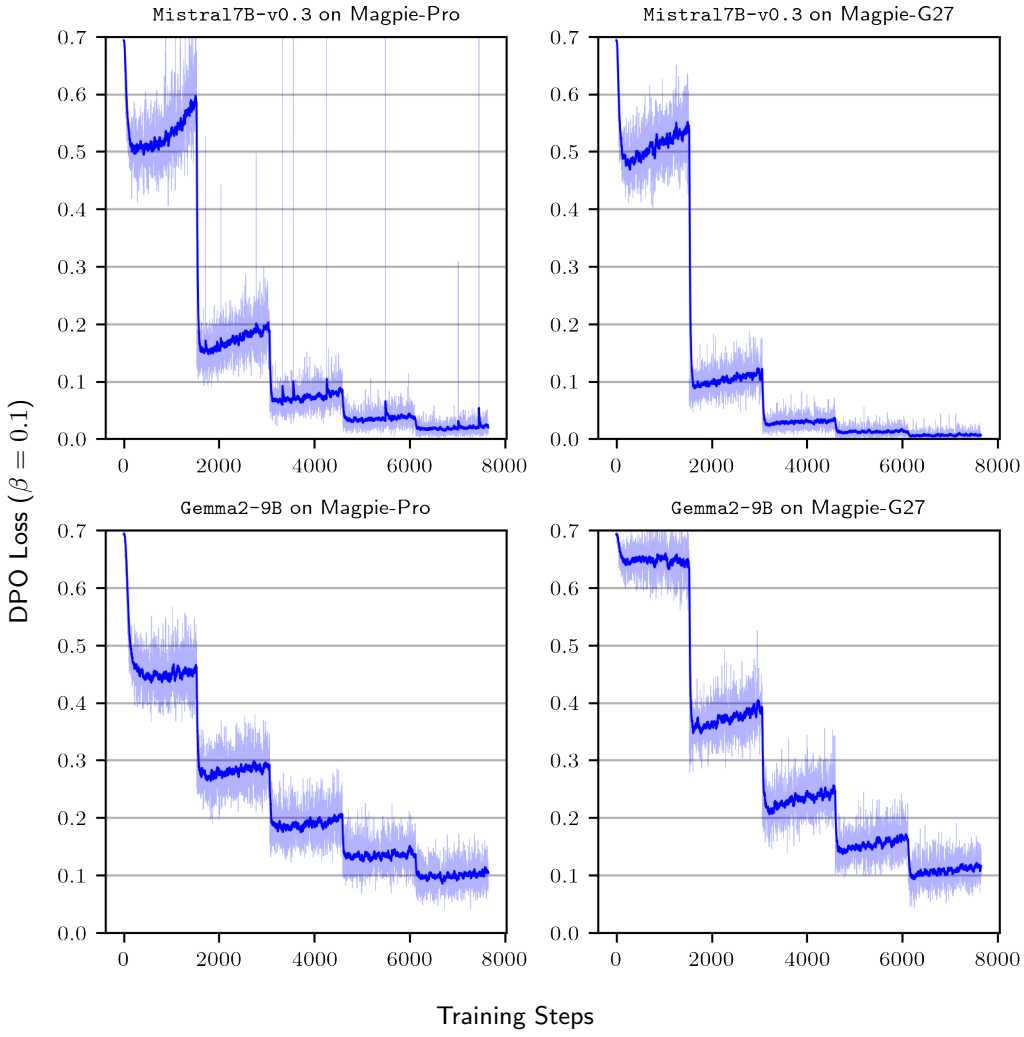

Figure 9: DPO loss for DPO-PG across all experimental configurations. The DPO loss is computed using $\beta = 0.1$. DPO-PG is able to optimize the DPO loss regardless of the architecture or dataset, validating Corollary H.5.1.

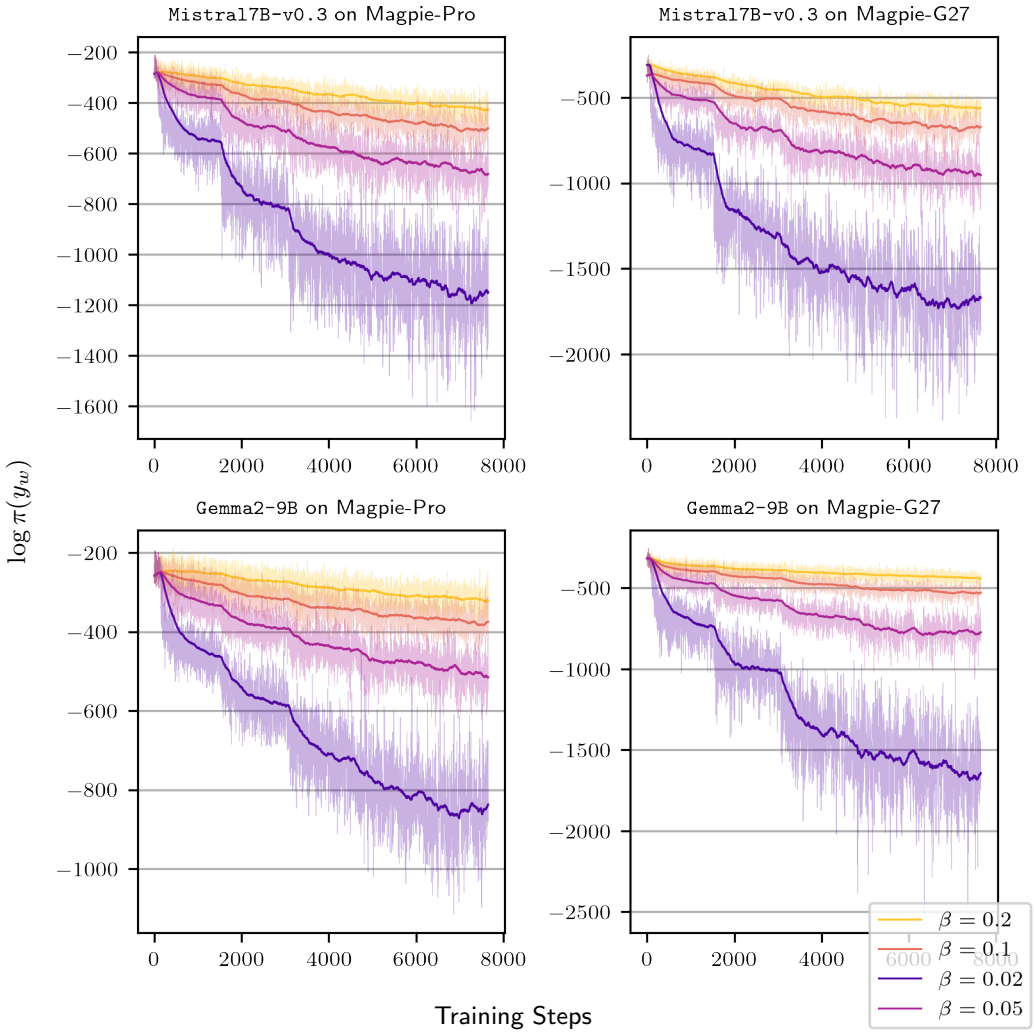

Figure 10: Log-likelihood change of y_w for DPO across all experimental configurations. The log-likelihood of chosen responses decreases throughout the training process, indicating log-likelihood displacement.

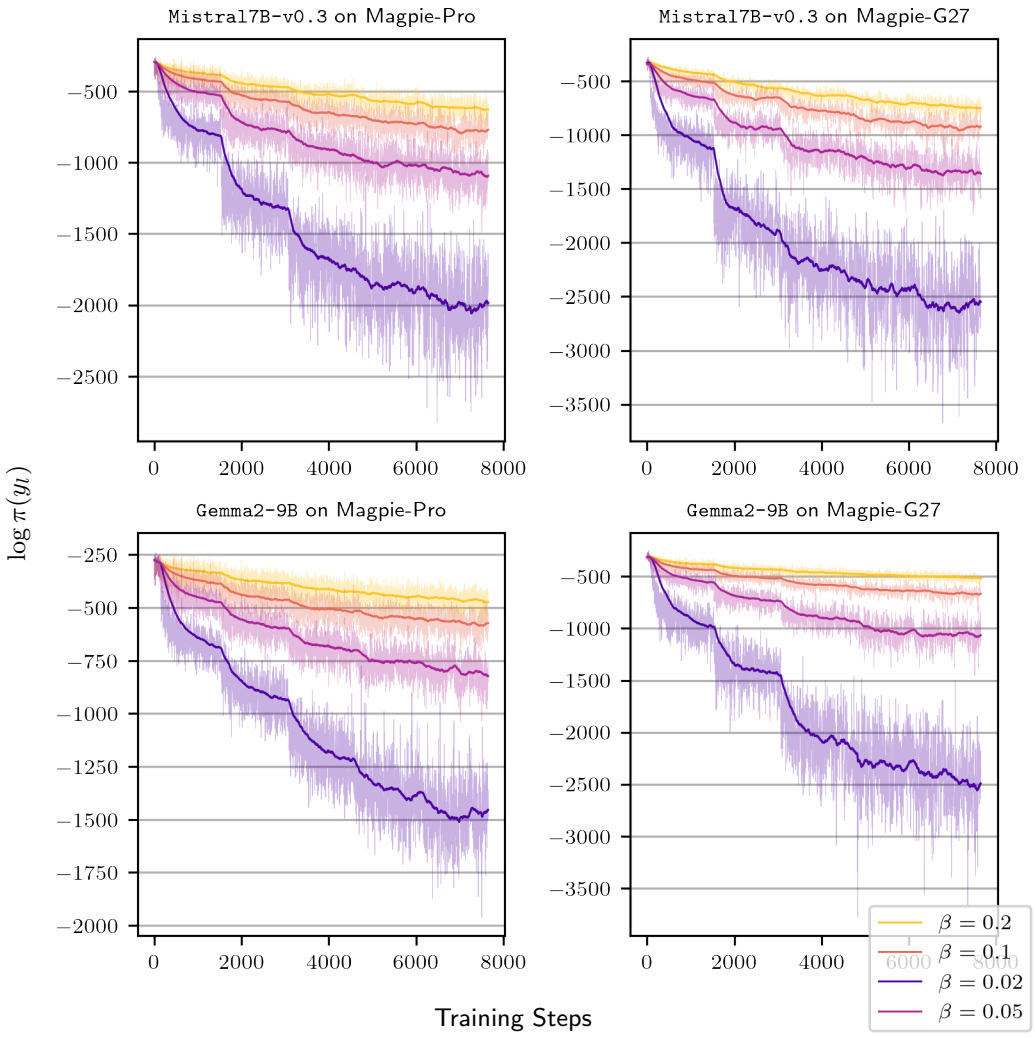

Figure 11: Log-likelihood change of y_l for DPO across all experimental configurations.

Table 3: Evaluation results for general instruction-following. We report the win-rate [%] for Arena-Hard (AH) and ELO score for Wild-Bench (WB), with the 95% confidence interval. We also specify the selected best epoch following the procedure in Appendix J.2. Standard DPO (policy smoothing) generally outperforms DPO-PG (policy reinforcing) on these benchmarks, suggesting general instruction-following is mainly associated with learning higher-entropy DID.

(a) `Mistral7B-v0.3` trained on Magpie-Pro

Method	Best Epoch	Arena-Hard	Wild-Bench
DPO $\beta = 0.2$	4	18.5 (-2.1, 1.7)	1145.70 (-10.14, 11.54)
DPO $\beta = 0.1$	1	23.4 (-1.9, 2.0)	1146.95 (-13.54, 11.42)
DPO $\beta = 0.05$	1	22.4 (-1.7, 1.9)	1145.02 (-12.67, 11.61)
DPO $\beta = 0.02$	1	20.1 (-1.8, 2.1)	1141.69 (-16.03, 12.44)
DPO-PG	5	19.6 (-1.9, 1.6)	1130.10 (-14.83, 13.30)

(b) `Mistral7B-v0.3` trained on Magpie-G27

Method	Best Epoch	Arena-Hard	Wild-Bench
DPO $\beta = 0.2$	2	30.0 (-2.6, 1.9)	1144.63 (-11.68, 10.82)
DPO $\beta = 0.1$	2	27.7 (-2.0, 2.4)	1144.89 (-11.60, 12.56)
DPO $\beta = 0.05$	1	27.0 (-2.3, 2.0)	1148.70 (-15.79, 13.12)
DPO $\beta = 0.02$	1	25.4 (-2.2, 2.1)	1136.91 (-14.56, 13.56)
DPO-PG	5	24.0 (-1.9, 1.4)	1130.08 (-18.00, 13.69)

(c) `Gemma2-9B` trained on Magpie-Pro

Method	Best Epoch	Arena-Hard	Wild-Bench
DPO $\beta = 0.2$	2	34.0 (-1.8, 2.4)	1166.95 (-11.04, 10.49)
DPO $\beta = 0.1$	4	33.8 (-2.8, 1.9)	1173.28 (-11.67, 11.45)
DPO $\beta = 0.05$	4	34.5 (-1.9, 2.6)	1170.93 (-10.94, 12.47)
DPO $\beta = 0.02$	1	36.7 (-2.0, 1.5)	1174.43 (-09.87, 14.16)
DPO-PG	4	24.9 (-2.3, 1.4)	1152.03 (-12.88, 14.60)

(d) `Gemma2-9B` trained on Magpie-G27

Method	Best Epoch	Arena-Hard	Wild-Bench
DPO $\beta = 0.2$	3	40.8 (-2.3, 2.6)	1174.26 (-12.09, 12.42)
DPO $\beta = 0.1$	3	38.4 (-2.6, 2.1)	1177.03 (-13.07, 11.75)
DPO $\beta = 0.05$	4	40.1 (-2.3, 2.5)	1174.88 (-12.92, 13.37)
DPO $\beta = 0.02$	2	32.1 (-1.8, 1.6)	1169.20 (-11.86, 10.50)
DPO-PG	5	31.2 (-1.9, 1.9)	1152.24 (-13.73, 15.33)

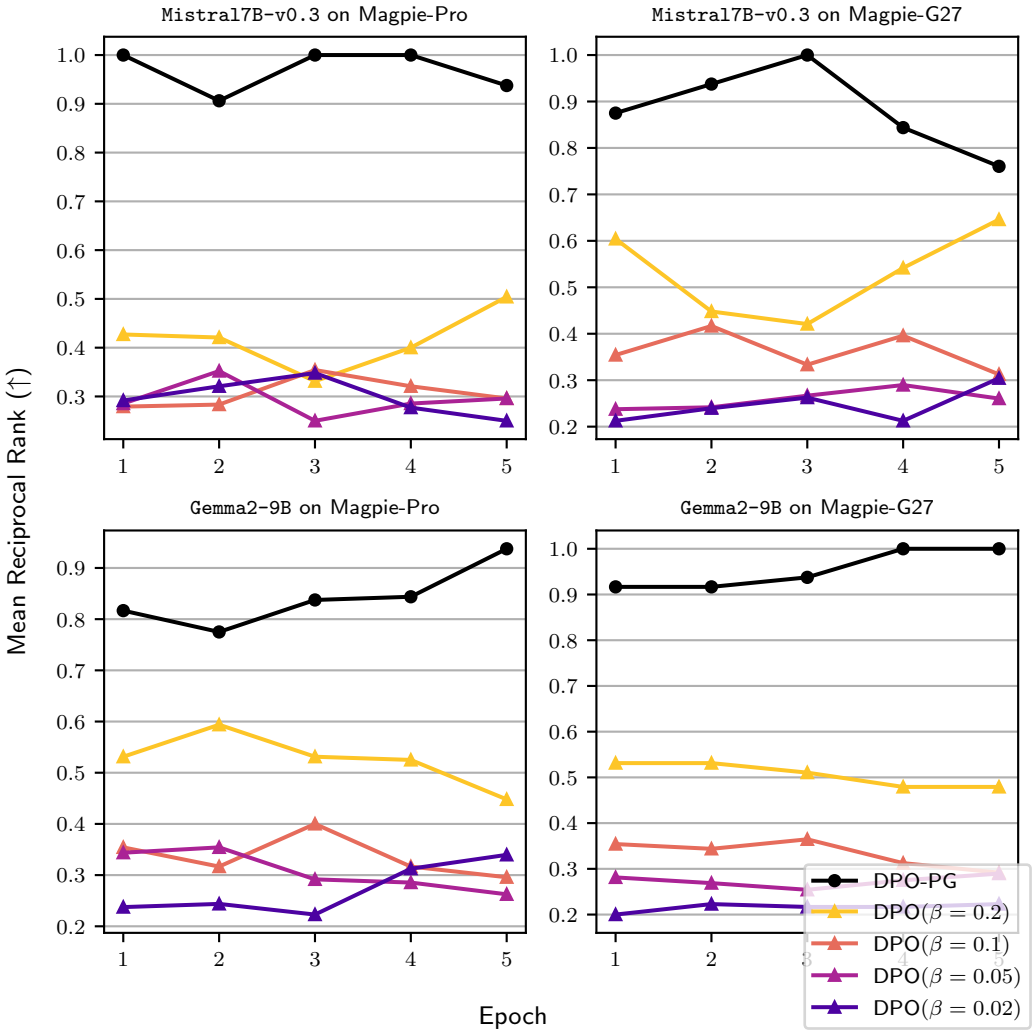

Figure 12: Mean Reciprocal Rank (MRR) across 8 QA benchmarks during training. The reinforcing method (DPO-PG) outperforms standard DPO which induces policy smoothing. This suggests QA capabilities are mainly associated with learning lower-entropy DID.

J Experimental setup

J.1 Synthetic setting

We conduct synthetic experiments involving Energy Based Models (EBMs) in a free-tier Google Colaboratory[5] CPU environment, using PyTorch [62]. For synthetic experiments, we use `torch.float32` as the default data type, and fix the training seed to 42 for reproducibility. We set the total class size as 32, and use a batch size of 512. We fix the learning rate to 0.001, and utilize the RMSprop [59] optimizer with gradient clipping at maximum norm of 1.0.

For fair comparison, we follow [12] in extensively searching the hyper-parameters for the following baseline methods:

- SLiC [23]: $\beta \in \{0.1, 0.5, 1.0, 2.0\}, \lambda \in \{0.1, 0.5, 1.0, 10.0\}$
- ORPO [10]: $\beta \in \{0.1, 0.5, 1.0, 2.0\}$
- SimPO [12]: $\beta \in \{2.0, 2.5\}, \gamma \in \{0.3, 0.5, 1.0, 1.2, 1.4, 1.6\}$
- Cal-DPO [29]: $\beta \in \{0.001, 0.002, 0.003, 0.01, 0.1\}$

The best hyper-parameter is chosen based on the minimum value of $\mathbb{D}_{\text{JS}}[\pi^* \parallel \pi]$ achieved through-out the training process.

J.2 Real-world setting

Magpie-G27 Dataset Magpie-G27 is an instruction-following preference dataset built from the prompts of Magpie-Air[6] and completed with responses generated by a stronger model (`google/gemma-2-27b-it`) [52]. Prompts in Magpie-G27 are disjoint from those in the Magpie-Pro dataset[7]. For each prompt, we sample five completions via `vLLM` [63] using the following sampling configuration:

`{n=5, temperature=0.9, top_p=1, max_tokens=4096, seed=42}`.

We then score these completions with `Skywork/Skywork-Reward-Gemma-2-27B-v0.2` [32] and select the highest- and lowest-scoring responses as y_w and y_l, respectively.

Training Setup To isolate the impact of alignment methods, we use pre-trained base models (*i.e.*, not instruction-tuned) paired with the official chat templates of their instruction-tuned counterparts.

Supervised Fine-Tuning (SFT) runs for one epoch with an effective batch size of 256, optimized via Adam [58] (default β_0, β_1; weight decay $= 0$). Training proceeds with a constant learning-rate of 5×10^{-6} and a linear warm-up over the first 10% of steps. The objective is standard cross-entropy loss applied to the full token sequence (including prompts and special template tokens). We fix the random seed to 0.

Preference Optimization During the alignment phase, we train for five epochs with an effective batch size of 64 using RMSprop [59] (no weight decay). We adopt a constant learning rate of 1×10^{-6} with 150-step linear warm-up and compute loss only over generated completions. We fix the random seed to 1. Models checkpoints are saved after each epoch and trained in `bfloat16` precision.

Infrastructure and Throughput All experiments use PyTorch FSDP [64] on A100 or H100 GPUs, with prompt lengths capped at 2,048 tokens and total sequence lengths at 4,096 tokens. Training `Mistral7B-v0.3` with DPO on 8 A100 GPUs takes approximately 3 hours for 1 Epoch on Magpie-Pro/G27, while `Gemma2-9B` on 8 H100 GPUs requires about 2 hours and 43 minutes for the same data size.

[5] https://colab.google/

[6] Available at `https://huggingface.co/datasets/Magpie-Align/Magpie-Air-DPO-100K-v0.1`.

[7] Available at `https://huggingface.co/datasets/Magpie-Align/Magpie-Llama-3.1-Pro-DPO-100K-v0.1`.

Evaluation We select the best checkpoint by absolute win-rate on **Arena-Hard** using `gpt-4.1-nano-2025-04-14`. Final performance on **Arena-Hard** is reported with `gpt-4.1-2025-04-14` (to reduce evaluation costs), following [22]. For **Wild-Bench v0.2**, we use `gpt-4o-2024-08-06` as recommended in the official repository[8]. During inference, we greedy-decode up to 4,096 tokens with `vLLM`. QA benchmarks are evaluated via the `lm-evaluation-harness` [65], with a minor modification: we prefix the system prompt as the first sentence of each user query.

K Entropy estimation

We measure the entropy of a language model π using a Monte-Carlo estimation:

$$H(\pi) \approx -\frac{1}{N}\sum_{i=1}^{N} \log \pi(y_i)$$

where we sample $y_i \sim \pi$ independently N-times. Note that this estimate converges to the true entropy almost surely as $N \to \infty$ [66].

Table 1 estimates the entropy on the training set. We sub-sample $1,000$ prompts from the training data, and estimate the entropy by sampling $N = 4$ responses for each prompt. We sample responses with temperature 1.0 and a max token length of $4,096$, with a fixed seed of 42.

A difficulty arises when measuring the entropy of the DID: $q_{\pi/\pi_{\text{ref}}}$

$$\begin{aligned} H(q_{\pi/\pi_{\text{ref}}}) &= -\sum_{y} q_{\pi/\pi_{\text{ref}}}(y) \log q_{\pi/\pi_{\text{ref}}}(y) \\ &= -\mathbb{E}_{y\sim q_{\pi/\pi_{\text{ref}}}}\left[\log \frac{\pi(y)}{\pi_{\text{ref}}(y)Z}\right] \end{aligned}$$

which requires estimating the partition function: $Z = \sum_y \frac{\pi(y)}{\pi_{\text{ref}}(y)}$ over all $y \in \mathcal{Y}$. We found that estimating Z by sampling $y \sim \pi$ and averaging $1/\pi_{\text{ref}}(y)$ to be unsuitable, due to high-variance.

We instead use the following procedure to estimate the DID entropy without estimating the partition function: First, we sample $y_w \sim \pi$ and $y_l \sim \pi_{\text{ref}}$, and construct a preference dataset $\mathcal{D} = \{(y_w, y_l)\}$. We then train another policy $\tilde{\pi}$ on $\mathcal{D}$ using $r = \log \tilde{\pi}$. As shown from Appendix F.2, the preference probability should follow:

$$p^*(y_w \succ y_l) = \sigma(\log q_{\pi/\pi_{\text{ref}}}(y_w) - \log q_{\pi/\pi_{\text{ref}}}(y_l)).$$

According to Theorem 2.1, optimizing $\tilde{\pi}$ with $r = \log \tilde{\pi}$ should converge the policy to $p^* = q_{\pi/\pi_{\text{ref}}}$. Therefore, we measure the entropy of $\tilde{\pi}$ to estimate the entropy of $q_{\pi/\pi_{\text{ref}}}$.

To measure the DID entropy of `Mistral7B-v0.3` in Table 2, we sub-sample $40,000$ prompts $\mathcal{D}'$ from the full training data $\mathcal{D}_0$, and greedy-decode a completion for each prompts, with a max token length of $4,096$. We then initialize $\tilde{\pi} \leftarrow \pi_{\text{ref}}$ and optimize $\tilde{\pi}$ using $r = \log \pi$ on $\mathcal{D}'$. We train for 1 epoch with a batch size 64 and RMSprop optimizer without any weight decay. The learning rate is set to 1e-6 with a linear warm-up of 150 steps. We then estimate the entropy of $\tilde{\pi}$ on $1,000$ unseen prompts from $\mathcal{D}_0 - \mathcal{D}'$. We sample $N = 4$ responses for each prompts with temperature 1.0, max token length $4,096$, and a fixed seed of 42.

K.1 Verification of potential outliers in DID entropy

While the estimated DID entropy values presented in Table 2 are mainly consistent with Claim 4.1, we further examine whether any of these values may be considered as outliers.

Notably, the measured DID entropy for `Mistral7B-v0.3` trained on Magpie-Pro with DPO $\beta = 0.1$ appears anomalous when compared to other values for the same model under different DPO settings. We argue that this specific value is an outlier, likely due to a suboptimal convergence of its corresponding policy $\tilde{\pi}$.

[8] `https://github.com/allenai/WildBench`.

Our reasoning is grounded in the theoretical expectation from Theorem 2.1 and 3.1. We expect the learned policy $\tilde{\pi}$ to approximate $q_{\pi^*/\pi_{\text{ref}}}$. Consequently, for samples $\tilde{y}$ drawn from an ideally converged $\tilde{\pi}$, there should exist some positive coefficient $\beta' > 0$ such that:

$$\tilde{\pi}(\tilde{y}) \propto \left(\frac{\pi^*(\tilde{y})}{\pi_{\text{ref}}(\tilde{y})}\right)^{\beta'}$$

To verify if $\tilde{\pi}$ adheres to this expected form, we estimate β' for completions generated by $\tilde{\pi}$. For each prompt, we use 4 completions sampled from $\tilde{\pi}$. The relationship can be expressed linearly in log-space:

$$\log \tilde{\pi}(\tilde{y}) = \beta' \left(\log \frac{\pi^*(\tilde{y})}{\pi_{\text{ref}}(\tilde{y})}\right) + C,$$

where C is a constant. With 4 samples per prompt, we can estimate β' (and C) via ordinary least squares linear regression by minimizing the sum of squared residuals [67].

A well-converged $\tilde{\pi}$ that accurately reflects $q_{\pi^*/\pi_{\text{ref}}}$ should consistently yield non-negative β' estimates. We quantify this consistency by measuring the proportion of prompts for which the estimated β' is non-negative; we term this metric "β'-accuracy". A low β'-accuracy indicates that $\tilde{\pi}$ significantly deviates from the target distribution $q_{\pi^*/\pi_{\text{ref}}}$, which in turn could render its estimated DID entropy unreliable.

For `Mistral7B-v0.3` trained on Magpie-Pro with DPO $\beta = 0.1$, the β'-accuracy is merely 28.1%, the lowest observed among all tested configurations. For comparison, `Mistral7B-v0.3` trained on Magpie-G27 with DPO $\beta = 0.02$ achieves a β'-accuracy of 80.4%. The markedly low β'-accuracy for the DPO $\beta = 0.1$ setting strongly suggests its $\tilde{\pi}$ is poorly aligned with the theoretical target. Therefore, we classify its estimated DID entropy as an outlier. We hypothesize that this poor alignment and the resulting outlier entropy value stem from instability during the optimization of $\tilde{\pi}$.

To support this hypothesis, Figure 13 illustrates the log-likelihood trend ($\log \tilde{\pi}(y)$) during training for `Mistral7B-v0.3` on Magpie-Pro with DPO ($\beta = 0.1$) and, for comparison, DPO-PG. We observe that the log-likelihood for DPO ($\beta = 0.1$) exhibits considerably larger fluctuations compared to DPO-PG, suggesting instability issues in the optimization process of $\tilde{\pi}$ in the former case.

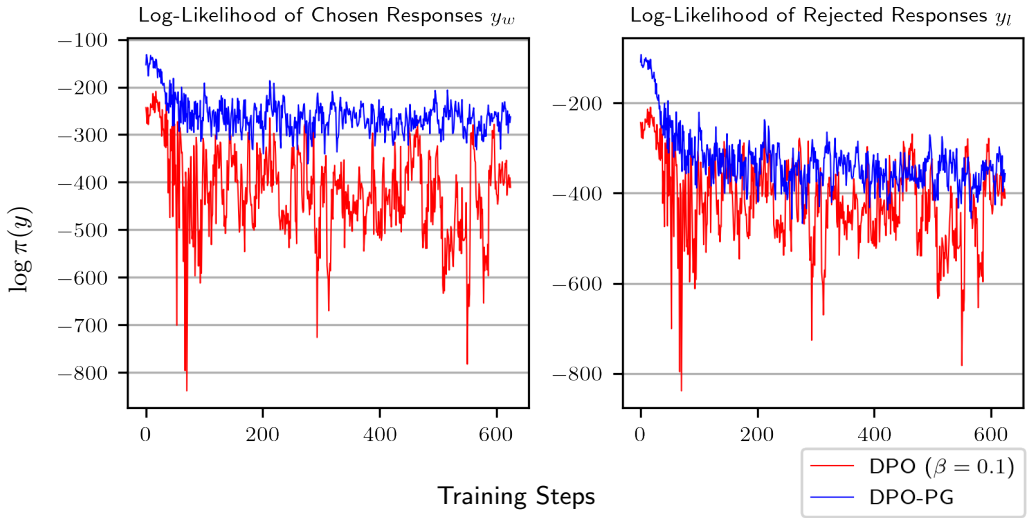

Figure 13: Log-likelihood $\log \tilde{\pi}(y)$ during $\tilde{\pi}$ training for DPO ($\beta = 0.1$) and DPO-PG with `Mistral7B-v0.3` on Magpie-Pro. The $\tilde{\pi}$ trained to approximate the DID of DPO ($\beta = 0.1$) shows greater log-likelihood instability, indicative of an unreliable convergence, which supports treating its corresponding DID entropy estimate as an outlier.

L Uniqueness of the optimal distribution of rejected responses

In this section, we derive the optimal distribution for sampling rejected responses under the DPO framework.

Theorem L.1 (Optimal Distribution For Sampling Rejected Responses)**.** *Given a reference policy π_{ref}, a target policy π^*, and a preference dataset $\mathcal{D} = \{(y_w, y_l) \mid y_w \sim \pi_{\text{ref}}, y_l \sim \pi_l\}$, we have the following relationship:*

$$\arg\max_{\pi} \mathbb{E}_{(y_w,y_l)\sim\mathcal{D}}[\log p(y_w \succ y_l \mid r)] = \arg\min_{\pi} \mathbb{D}_{\text{KL}}[\pi^*(y) \parallel \pi(y)]$$
$$\iff \pi_l(y) \propto \pi_{\text{ref}}(y) \left(\frac{\pi_{\text{ref}}(y)}{\pi^*(y)}\right)^{\beta}, \forall y \in \mathcal{Y}$$

where $r(y) = \beta \log \frac{\pi(y)}{\pi_{\text{ref}}(y)}$.

Proof. **Case 1** First, according to Theorem 2.1, preference optimization with $r = \beta \log \pi/\pi_{\text{ref}}$ leads to $\mathbb{D}_{\text{KL}}[p^*(y) \parallel P(Y = y \mid r^*)] = 0$ in which $P(Y = y \mid r^*) \propto \left(\frac{\pi^*(y)}{\pi_{\text{ref}}(y)}\right)^{\beta}$. Therefore, it can be shown that DPO training with $r = \beta \log \pi/\pi_{\text{ref}}$ converges the policy to the following target policy:

$$\pi^*(y) \propto \pi_{\text{ref}}(y) p^*(y)^{\frac{1}{\beta}}.$$

Since we have $\pi^*(y) \propto \pi_{\text{ref}}(y) p^*(y)^{\frac{1}{\beta}}$, for all $(y_w, y_l) \in \mathcal{D}$, the preference probability must follow:

$$p^*(y_w \succ y_l) = \sigma(\beta \log q_{\pi^*/\pi_{\text{ref}}}(y_w) - \beta \log q_{\pi^*/\pi_{\text{ref}}}(y_l)).$$

However, as shown from Appendix F.2, the preference probability must also follow:

$$p^*(y_w \succ y_l) = \sigma(\log q_{\pi_{\text{ref}}/\pi_l}(y_w) - \log q_{\pi_{\text{ref}}/\pi_l}(y_l)).$$

If we assume that $\mathcal{D}$ is sufficiently large such that its outcomes cover $\mathcal{Y}$, then for all $y \in \mathcal{Y}$, we must have the following:

$$q_{\pi_{\text{ref}}/\pi_l}(y) \propto q_{\pi^*/\pi_{\text{ref}}}(y)^{\beta} \iff \pi_l(y) \propto \pi_{\text{ref}}(y) \left(\frac{\pi_{\text{ref}}(y)}{\pi^*(y)}\right)^{\beta}.$$

Case 2 Now, consider the inverse case. For all $y \in \mathcal{Y}$, assume the following:

$$\pi_l(y) \propto \pi_{\text{ref}}(y) \left(\frac{\pi_{\text{ref}}(y)}{\pi^*(y)}\right)^{\beta}.$$

This immediately yields the power-law relationship: $q_{\pi_{\text{ref}}/\pi_l}(y) \propto q_{\pi^*/\pi_{\text{ref}}}(y)^{\beta}$. Applying Theorem 3.4, it follows that preference optimization with $r = \beta \log \pi/\pi_{\text{ref}}$ yields $\pi = \pi^*$.

Therefore, given a reference policy, target policy, and the β-parameter used for DPO training, the distribution over rejected responses is uniquely determined as $\pi_l(y) \propto \pi_{\text{ref}}(y) \left(\frac{\pi_{\text{ref}}(y)}{\pi^*(y)}\right)^{\beta}$, provided that the chosen responses y_w are sampled from π_{ref}. □