DIFFERENTIAL INFORMATION DISTRIBUTION: A BAYESIAN PERSPECTIVE ON DIRECT PREFERENCE OPTIMIZATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Direct Preference Optimization (DPO) has been widely used for aligning language models with human preferences in a supervised manner. However, several key questions remain unresolved: the rationale behind its log-ratio reward, how the statistical structure of preference datasets shapes its training dynamics, and how those dynamics impact downstream capabilities. We approach these questions from a Bayesian perspective, interpreting the goal of preference optimization as learning the differential information required to update a reference policy into a target policy. To formalize this view, we introduce the DIFFERENTIAL INFORMATION DISTRIBUTION (DID), defined as the distribution over samples that carry the Bayesian evidence required to update policies. We introduce three complementary insights by viewing preference optimization through the DID. First, we find that DPO's log-ratio reward is uniquely justified when preferences encode the Differential Information needed to update a reference policy into the target policy. Second, we discuss how commonly observed training dynamics in DPO, including changes in log-likelihood and policy exploration, stem from a power-law DID relationship. Finally, we analyze how training dynamics influence downstream performance using the entropy of DID, a principled measure of uncertainty in the learned information. We observe that learning high-entropy DID improves open-ended instruction-following, while low-entropy DID benefits knowledge-intensive QA. Taken together, our results show that DPO's reward design, training dynamics, and downstream capabilities all emerge as natural consequences of learning Differential Information, offering both a principled theoretical foundation and practical guidance for preference-based alignment.[1]

1 INTRODUCTION

Aligning language models to human preferences is essential for both safety and usefulness (Ouyang et al., 2022; Bai et al., 2022). Among various alignment methods, Direct Preference Optimization (DPO) (Rafailov et al., 2023) has gained popularity for its strong empirical performance, training stability, and computational efficiency (Xiao et al., 2024b; Liu et al., 2025b). Despite its widespread use, several fundamental questions remain: what justifies the log-ratio reward beyond its derivation from a KL-regularized RL objective, what statistical structure in preference datasets underlie DPO's training dynamics, and how these dynamics impact downstream capabilities.

To address these gaps, we propose a Bayesian perspective that interprets the goal of preference optimization as learning the information needed to update a reference policy into a target policy. We call this information "Differential Information" as it represents the difference in the information encoded by the two policies. To formalize this, we introduce the DIFFERENTIAL INFORMATION DISTRIBUTION (DID), defined as the distribution of samples containing the Bayesian evidence[2] for this policy update. The DID can be expressed as the normalized ratio of the two policy distributions (Theorem 2.2). By analyzing the DID in preference optimization, we show that DPO's key

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

[2] We consider Bayesian evidence that is conditionally independent of the prior given a sample, ensuring that this evidence reflects a sample's intrinsic content rather than the distribution it was drawn from. This aligns with Shannon information being additive for *independent* events. See Section 2 and Appendix D for details.

components–reward parameterization, training dynamics, and learned capabilities–emerge naturally from this Bayesian perspective.

In Section 3, we present a Bayesian interpretation of the optimality of DPO's log-ratio reward parameterization. We first show how a preference data generation process naturally yields a preference distribution that encodes the Differential Information required to update a reference policy into a target policy. We then prove that the log-ratio reward is the *unique* Bradley-Terry reward that learns this target policy, revealing that DPO's design follows directly from Bayesian principles. We validate these findings through controlled Energy-Based Model experiments, where we simulate a preference dataset encoding Differential Information and demonstrate that only the log-ratio reward converges to the target policy while other objectives (*e.g.*, SimPO (Meng et al., 2024)) fail to do so. We also validate our analysis of the DID relationship in DPO on practical training settings.

In Section 4, we analyze the training dynamics of DPO based on a power-law relationship in the DID imposed by DPO. This DID relationship links the DPO-converged policy to the preference sampling distribution. Because the normalized ratio form of the DID is algebraically tied to the KL-divergence, it explains the consistent changes in the log-likelihood observed during DPO training. The DID power-law further clarifies how policy exploration is jointly influenced by the KL-penalty term and the sharpness of preference data, where sharpness reflects the sparsity (reciprocal of sampling temperature) of distributions over chosen and rejected responses. On both controlled setups and real-world datasets, we confirm these predictions and show that the identified properties of DPO persist under gradient-based stochastic optimization.

In Section 5, we investigate the empirical link between training dynamics and downstream capabilities by analyzing the Shannon entropy of the DID. Recent work finds that preventing log-likelihood displacement (LLD) improves factual accuracy (*e.g.*, MMLU (Hendrycks et al., 2021)) at the cost of open-ended tasks (*e.g.*, Wild-Bench (Lin et al., 2024a)) (Shi et al., 2024; Chen et al., 2024; Xiao et al., 2024a). We hypothesize that this trade-off reflects differences in the DID entropy. To test this, we compare standard DPO against a variant that prevents LLD. We train `Mistral7B-v0.3` (Jiang et al., 2024) and `Qwen3-4B` (Team, 2025) on Magpie-Pro and Magpie-G27 datasets (Xu et al., 2024b). Across all settings, preventing LLD consistently learns a low-entropy DID that improves factual QA, while allowing LLD learns a high-entropy DID that enhances open-ended generation. These results suggest that DID entropy could serve as a useful factor in characterizing whether a model's learned capabilities align with factual precision or open-ended generation.

Taken together, our Bayesian perspective unifies reward design, training dynamics, and learned capabilities of DPO. By explicitly linking preference data to the Differential Information they convey, our approach provides both theoretical grounding and practical guidance for designing and understanding preference-based alignment.

2 Preliminaries

Let $\mathcal{Y}$ denote the sample space of all possible sentences. A policy (*i.e.*, language model) π defines a probability distribution over $\mathcal{Y}$, and we assume full support, *i.e.*, $\pi(y) > 0$ for all $y \in \mathcal{Y}$. Let π^* be the target policy we wish to learn and π_{ref} be a fixed reference policy.

Preferences are ordered pairs (y_w, y_ℓ), where y_w is preferred to y_ℓ, written as $y_w \succ y_\ell$. The Bradley-Terry (BT) model (Bradley & Terry, 1952; Luce et al., 1959) assigns the probability of such a preference under an implicit distribution p^* as

$$p^*(y_w \succ y_\ell) := \frac{p^*(y_w)}{p^*(y_w) + p^*(y_\ell)}.$$

If a latent reward $r : \mathcal{Y} \to \mathbb{R}$ induces a Boltzmann distribution $P(Y = y \mid r) \propto \exp(r(y))$, then the BT preference probability can be expressed via the logistic sigmoid $\sigma(x) = 1/(1 + \exp(-x))$ as

$$p(y_w \succ y_\ell \mid r) := \sigma\big(r(y_w) - r(y_\ell)\big).$$

DPO objective and log-ratio reward. Direct Preference Optimization (DPO) (Rafailov et al., 2023) parameterizes a BT reward derived from the KL-regularized RL objective, the log-ratio between the learned policy and the reference: $r_{\text{DPO}}(y) \coloneqq \beta \log(\pi(y)/\pi_{\text{ref}}(y))$, where $\beta > 0$ corresponds to the KL-penalty strength. Under this parameterization, the preference probability becomes

$$p(y_w \succ y_\ell \mid r_{\text{DPO}}) \coloneqq \sigma\left(\beta \log \frac{\pi(y_w)}{\pi_{\text{ref}}(y_w)} - \beta \log \frac{\pi(y_\ell)}{\pi_{\text{ref}}(y_\ell)}\right).$$

Preference generation assumption. Following prior works (Pan et al., 2025; Higuchi & Suzuki, 2025), we assume preferences (y_w, y_ℓ) are sampled independently from two distributions π_w and π_ℓ respectively. Given an unordered pair of distinct responses (y_1, y_2), the ground-truth probability that y_1 is preferred over y_2 is

$$\begin{aligned} p(y_1 \succ y_2) &\coloneqq P(y_1 \sim \pi_w,\ y_2 \sim \pi_\ell \mid \text{sampled pair is } (y_1, y_2)) \\ &= \frac{\pi_w(y_1)\pi_\ell(y_2)}{\pi_w(y_1)\pi_\ell(y_2) + \pi_w(y_2)\pi_\ell(y_1)}. \end{aligned}$$

We denote by $\mathcal{D} = \{(y_w, y_\ell) \mid y_w \sim \pi_w,\ y_\ell \sim \pi_\ell\}$ a preference dataset consisting of such pairs. We assume $\pi_w \neq \pi_\ell$ and $\pi_{\text{ref}} \neq \pi^*$ in the following discussions.

Preference optimization as distribution matching. Preference optimization maximizes the empirical likelihood of observed preferences under a BT model parameterized by a reward r. Under standard identifiability and coverage assumptions, maximizing the preference likelihood is equivalent to fitting the reward-induced Boltzmann distribution to the implicit preference distribution. We cite the following standard result from Dumoulin et al. (2023) (Proof in Appendix J.1).

Theorem 2.1 (Preference vs. Distribution Matching (Dumoulin et al., 2023))**.** *Let $\mathcal{D} = \{(y_w, y_\ell)\}$ be a sufficiently large preference dataset where the sets of y_w and y_ℓ cover $\mathcal{Y}$. Then preference optimization on $\mathcal{D}$ is equivalent to fitting the reward-induced distribution $P(Y = y \mid r)$ to the implicit preference distribution $p^*(y)$:*

$$\max_r \mathbb{E}_{(y_w, y_\ell)\sim\mathcal{D}} [\log \sigma(r(y_w) - r(y_\ell))] \iff \min_r \mathbb{D}_{\text{KL}} [p^*(y) \| P(Y = y \mid r)].$$

Differential Information Distribution. We now introduce the DIFFERENTIAL INFORMATION DISTRIBUTION (DID) to formalize the information that drives the update from a prior belief π_{ref} into a posterior π^*, which we later use to interpret preference optimization. The term *differential* highlights that the difference in the information contained in π^* and π_{ref} is precisely the additional Bayesian evidence required to update π_{ref} into π^*.

Suppose we start from a prior π_{ref} over sentences and then observe new Bayesian evidence X that updates π_{ref} to π^*. We define the DID as the distribution over sentences that carry this incremental evidence. A central postulate is that X is conditionally independent of the prior distribution given some sentence y. In other words, the probability that y exhibits X does not depend on whether y was sampled from π_{ref}. This ensures that the information X attributed to each sentence reflects its intrinsic features. This also parallels how the difference in Shannon information, $-\log P_A - (-\log P_B) = -\log \frac{P_B}{P_A}$, equals the Shannon information of an *independent* event X that updates P_A into P_B via Bayes' rule: $P_{A,X} = P_A P_X = P_B$. See Appendix D for details.

Definition 2.2 (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^*.

Intuitively, if our initial belief regarding which sentence y is the correct sentence follows the distribution $\pi_{\text{ref}}(y)$, and then learn that "all sentences satisfying X are correct", our updated belief becomes $\pi^*(y)$. The following theorem shows that the DID can be computed directly from the normalized likelihood ratio.

Theorem 2.3 (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:*

$$P(Y = y \mid X) = \frac{\pi^*(y)/\pi_{\text{ref}}(y)}{Z} \coloneqq q_{\pi^*/\pi_{\text{ref}}}(y),$$

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

(Proof in Appendix D.3) Therefore, the normalized ratio distribution $q_{\pi^*/\pi_{\text{ref}}}$ can be understood as the Differential Information Distribution responsible for the update from π_{ref} to π^*.

3 Optimality of DPO's log-ratio reward

In this section, we begin our analysis of preference optimization through the lens of Differential Information. We reframe the goal of preference optimization as learning the Differential Information that updates a reference policy π_{ref} into a target policy π^*. We first show how a preference dataset naturally encodes that Differential Information (Theorem 3.1), then prove that DPO's log-ratio reward is the unique Bradley-Terry reward that learns π^* from such data (Theorem 3.2). Finally, we validate these claims in a controlled Energy-Based Model experiment and also using real-world datasets (Section 3.3).

3.1 How preferences encode Differential Information

Since policy updates in preference optimization are driven by the preference distribution p^* of a dataset, it is essential to characterize precisely *when* p^* contains the Differential Information required to transform π_{ref} into π^*. We find this condition is met when the Differential Information Distributions (DIDs) of the underlying policies are related by a power-law.

Theorem 3.1 (Preferences Encoding Differential Information)**.** *Consider a preference dataset $\mathcal{D} = \{(y_w, y_\ell) \mid y_w \sim \pi_w, y_\ell \sim \pi_\ell\}$. Let π^* be the target policy. If the Differential Information Distribution between policies match up to an exponent $\beta > 0$:*

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

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

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

(Proof in Appendix J.2) The condition requires that the Differential Information that updates the rejected response distribution π_ℓ into the chosen one π_w should align with the DID from π_{ref} to π^*, up to an exponent $\beta > 0$. When this holds, the dataset's preference distribution carries the Bayesian evidence needed to update π_{ref} into π^*.

3.2 Bradley-Terry reward for learning Differential Information

As we have characterized when a preference dataset encodes the Differential Information necessary for learning π^*, we now ask which functional form of the reward parameterization $r(y)$ recovers π^*. We find that the log-ratio form used by DPO is the unique functional form (up to a constant) that makes π^* the global optimizer of the training objective.

Theorem 3.2 (Optimal Reward for Learning Differential Information)**.** *Let $\mathcal{D}$ be a preference dataset satisfying Theorem 3.1, encoding the Differential Information required to learn the target policy π^*. Then, for some constant C, we have*

$$\pi^* = \arg\max_{\pi} \mathbb{E}_{(y_w, y_\ell)\sim\mathcal{D}}\left[\log\sigma(r(y_w) - r(y_\ell))\right] \iff r(y) = \beta \log \frac{\pi(y)}{\pi_{\text{ref}}(y)} + C.$$

(Proof in Appendix J.3) This justifies DPO's log-ratio structure: if preference captures the Differential Information needed to improve π_{ref} toward π^*, then using the log-ratio reward 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. (2023), originally motivated by the KL-regularized RL objective. This highlights the Bayesian structure of DPO in learning Differential Information.

The key results of Theorems 3.1 and 3.2 can be summarized into the following relationship.

Corollary 3.2.1 (DID Power-Law of DPO)**.** *Consider a preference dataset $\mathcal{D} = \{(y_w, y_\ell) \mid y_w \sim \pi_w, y_\ell \sim \pi_\ell\}$ and a policy π^* obtained as a stationary point of preference optimization using the log-ratio reward $r = \beta \log(\pi/\pi_{\text{ref}})$ on $\mathcal{D}$. Then, a power-law relationship between the DID of policies must hold:*

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

(Proof in Appendix J.4) Corollary 3.2.1 can be read two ways: either datasets that satisfy the DID power-law lead DPO to recover π^*; conversely, if DPO has converged to some π^* then the dataset's sampling distributions must satisfy this power-law relation.[3]

3.3 Experiments

We first validate our theoretical findings in a controlled setup using Energy-Based Models.

Setup. We define policies $\pi_\theta(i) = \exp(\theta_i)/\sum_j \exp(\theta_j)$ for class $i \in \{1, \dots, K\}$ 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, to construct the target policy π^*, we set the target logits $\theta^* = \theta_{\text{ref}}/\tau$ with $0 < \tau < 1$ for reinforcing and $\tau > 1$ for smoothing. The logits of π_ℓ are set as $\theta_\ell = 2\theta_{\text{ref}} - \theta^*$, which aligns the DID between policies: $q_{\pi_{\text{ref}}/\pi_\ell} = q_{\pi^*/\pi_{\text{ref}}}$. Finally, preference pairs (y_w, y_ℓ) are constructed by sampling $y_w \sim \pi_{\text{ref}}$ and $y_\ell \sim \pi_\ell$, and labeled as $y_w \succ y_\ell$ (Hyper-parameters in Appendix M.1).

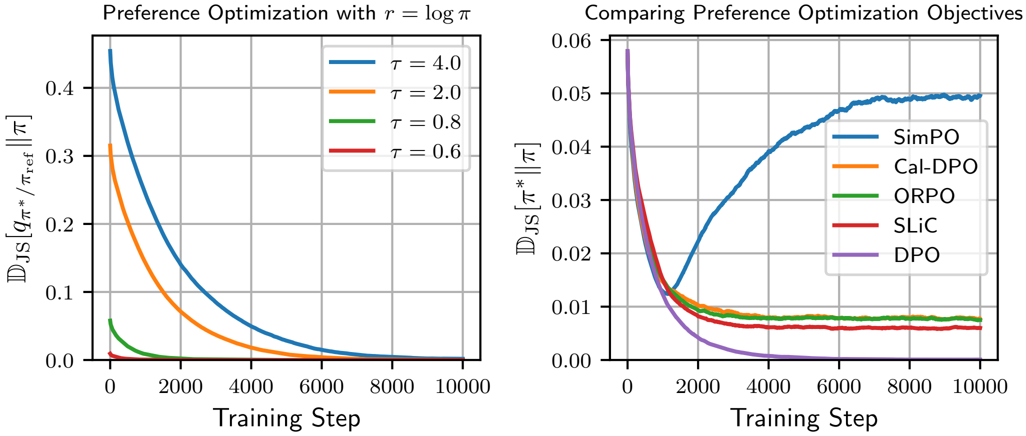

Figure 1: *Left*: Optimization using $r = \log \pi$ on preference data satisfying the DID power-law. The Jensen-Shannon Divergence $\mathbb{D}_{\text{JS}}[q_{\pi^*/\pi_{\text{ref}}} \| \pi]$ converges to 0, confirming Theorem 3.1 that the preference encodes Differential Information. *Right*: Comparison of $\mathbb{D}_{\text{JS}}[\pi^* \| \pi]$ using different objectives on the same data with $\tau = 4$. Standard DPO ($r = \log(\pi/\pi_{\text{ref}})$, purple) uniquely converges to π^*, consistent with Theorem 3.2.

Do preferences encode Differential Information? Theorem 3.1 predicts that the preference distribution encodes 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^* (Dumoulin et al., 2023; Xu et al., 2024a; Liu et al., 2024b). 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.

Figure 1 shows that the JS divergence consistently converges to zero, meaning that the policy trained to directly fit p^* converges to the DID $q_{\pi^*/\pi_{\text{ref}}}$. This confirms Theorem 3.1 in that **sampling chosen and rejected samples each from a distribution satisfying the DID power-law yields a preference distribution encoding the Differential Information required to learn the target policy.**

Is the log-ratio reward optimal? To test Theorem 3.2, we compare optimization with the log-ratio reward $r = \log(\pi/\pi_{\text{ref}})$ against several alternative objectives: SLiC (Zhao et al., 2023b), ORPO (Hong et al., 2024), SimPO (Meng et al., 2024), and Cal-DPO (Xiao et al., 2024a). All methods are trained on the same synthetic dataset and we compare $\mathbb{D}_{\text{JS}}[\pi^* \| \pi]$ for each method.

Figure 9 shows that the DPO log-ratio objective is the *only* method that consistently minimizes $\mathbb{D}_{\text{JS}}[\pi^* \| \pi]$ across various τ values, converging to π^*. This empirical result supports Theorem 3.2,

[3] Corollary 3.2.1 directly yields a closed-form expression of the optimal DPO dataset (Appendix F).

highlighting that **when preferences encode Differential Information, the log-ratio reward succeeds in recovering the target policy while other objectives fail to do so.**

Does the DID power-law hold in real-world DPO training? Corollary 3.2.1 predicts that a DPO-converged policy π^* should satisfy the DID power-law: $q_{\pi_w/\pi_\ell}(y) \propto q_{\pi^*/\pi_{\text{ref}}}(y)^\beta$. In particular, the DID power-law implies that when the reference policy is fine-tuned on the chosen responses ($\pi_{\text{ref}} = \pi_w$), the average reward-margin of the converged π^* should be invariant to the choice of $\beta > 0$ used for DPO training (proof in Appendix G):

$$\mathbb{E}_{y_w \sim \pi_{\text{ref}}}[\beta \log \frac{\pi^*(y_w)}{\pi_{\text{ref}}(y_w)}] - \mathbb{E}_{y_\ell \sim \pi_\ell}[\beta \log \frac{\pi^*(y_\ell)}{\pi_{\text{ref}}(y_\ell)}] = \mathbb{D}_{\text{KL}}\left[\pi_{\text{ref}} \| \pi_\ell\right] + \mathbb{D}_{\text{KL}}\left[\pi_\ell \| \pi_{\text{ref}}\right].$$

To test this on real data, we fine-tuned `Mistral7B-v0.3` and `Qwen3-4B` on the Magpie-Pro and Magpie-G27 instruction-following datasets (Appendix M.2). The reference policy was first fine-tuned for one epoch on the chosen responses and then trained with DPO for one epoch (1,530 steps) using $\beta \in \{0.05, 0.1, 0.2\}$. Final reward- and log-margins were measured over the last 50 steps.

Figure 10 shows that the log-margins of the DPO-converged policy differ substantially across β, approximately scaling inversely with β as predicted. In contrast, Figure 11 shows that the reward-margins remain within a narrow, stable range. **These results provide strong empirical support that the DID power-law of Corollary 3.2.1 holds robustly in practical DPO training setups.** To our knowledge, prior gradient-based analyses (Pal et al., 2024; Feng et al., 2024; Razin et al., 2024) cannot predict such log/reward-margin trends at convergence, which demonstrates the explanatory strength of our DID-based analysis.

4 Training Dynamics of DPO

The power-law structure of the Differential Information Distribution (DID) identified in the previous section (Corollary 3.2.1) determines how DPO updates policies during training. This provides a unified lens through which we can understand the training dynamics of DPO. We discuss how the power-law DID relationship proves general guarantees on the log-likelihood change in DPO (Theorem 4.1). Next, we analyze which factors impact policy exploration during DPO training, based on the power-law DID relationship (Theorem 4.2).

4.1 Log-likelihood change

Corollary 3.2.1 explains the characteristic log-likelihood shifts observed during DPO training. The DID power-law, which DPO must satisfy at convergence, clarifies how the preference sampling distributions and the converged policy are linked together. Combined with Jensen's and Gibbs' inequalities, this relationship predicts the asymmetric shifts in the log-likelihoods of chosen and rejected responses. Unlike prior analyses (Feng et al., 2024; Razin et al., 2024; Cho et al., 2025), our statements derived from this principle place no restrictions on step sizes, gradients, or parameterization methods. We further extend beyond the in-distribution regime (*i.e.*, samples π_{ref} was trained on) considered in some previous work (Rafailov et al., 2024).

Theorem 4.1 (Log-Likelihood Change of DPO)**.** *Consider a preference dataset $\mathcal{D} = \{(y_w, y_\ell) \mid y_w \sim \pi_{\text{ref}}, y_\ell \sim \pi_\ell\}$, and π^* obtained by preference optimization on $\mathcal{D}$ using the log-ratio reward $r = \beta \log \pi/\pi_{\text{ref}}$. Then, for any $\beta > 0$, π^* must decrease the average log-likelihood of y_ℓ:*

$$\mathbb{E}_{y_\ell \sim \pi_\ell}\left[\log \pi^*(y_\ell)\right] < \mathbb{E}_{y_\ell \sim \pi_\ell}\left[\log \pi_{\text{ref}}(y_\ell)\right].$$

Conversely, if π_{ref} was fine-tuned on y_ℓ (i.e., $\pi_{\text{ref}} = \pi_\ell$), then, for any $\beta \geq 1$, π^* must increase the average log-likelihood of y_w:

$$\mathbb{E}_{y_w \sim \pi_w}\left[\log \pi^*(y_w)\right] > \mathbb{E}_{y_w \sim \pi_w}\left[\log \pi_{\text{ref}}(y_w)\right].$$

See Appendix J.5 for proof. The theorem captures a basic asymmetric effect of DPO: when y_w is sampled from π_{ref}, the converged policy π^* must decrease $\log \pi(y_\ell)$, and when $y_\ell \sim \pi_{\text{ref}}$, π^* must increase $\log \pi(y_w)$ for $\beta \geq 1$. The theorem therefore connects the power-law DID to concrete, observable training dynamics and clarifies how and why likelihoods change during DPO training.[4]

[4] See Appendix G for a discussion on how the log-margin $\log \pi(y_w) - \log \pi(y_\ell)$ evolves during DPO training, and how it implies an information-theoretic triangle inequality that must hold at convergence.

4.2 POLICY EXPLORATION

The characteristics of policy exploration in DPO can also be explained using the power-law DID relationship. In particular, this DID structure allows DPO to adaptively adjust its KL-divergence from the reference policy π_{ref} based on the sharpness of the preference data. Intuitively, when preference labels are weak (*i.e.*, $p^*(y_1 \succ y_2)$ being close to 0.5), the power-law DID relationship constrains the DPO-converged policy π^* to remain close to π_{ref}. Conversely, when preference probabilities are stronger (*i.e.*, $p^*(y_1 \succ y_2)$ being close to 0 or 1), the same DID relationship drives the converged policy farther away from π_{ref}, under the same KL-penalty β. This trade-off is formalized below (proof in Appendix J.6).

Theorem 4.2 (Adaptive Policy Exploration of DPO)**.** *Let $\mathcal{D} = \{(y_w, y_\ell) \mid y_w \sim \pi_{\text{ref}},\ y_\ell \sim \pi_\ell\}$ be a preference dataset with an implicit Bradley-Terry preference distribution $p^*_{\mathcal{D}}$. Consider another dataset $\mathcal{D}' = \{(y_w, y_\ell)\}$ whose implicit Bradley-Terry distribution $p^*_{\mathcal{D}'}$ is a "sharpened" version of $p^*_{\mathcal{D}}$, in the sense that there exists $\alpha > 1$ such that for all pairs $(y_w, y_\ell) \in \mathcal{Y} \times \mathcal{Y}$,*

$$p^*_{\mathcal{D}'}(y_w \succ y_\ell) = \frac{\big(p^*_{\mathcal{D}}(y_w)\big)^\alpha}{\big(p^*_{\mathcal{D}}(y_w)\big)^\alpha + \big(p^*_{\mathcal{D}}(y_\ell)\big)^\alpha} = \sigma\big(\alpha \log p^*_{\mathcal{D}}(y_w) - \alpha \log p^*_{\mathcal{D}}(y_\ell)\big).$$

For the same reference policy π_{ref} and any $\beta > 0$, let $\pi^*_{\mathcal{D}}$ and $\pi^*_{\mathcal{D}'}$ denote the policies obtained by preference optimization on $\mathcal{D}$ and $\mathcal{D}'$, respectively, using the log-ratio reward $r = \beta \log \pi/\pi_{\text{ref}}$. Then the strengthened dataset $\mathcal{D}'$ induces a strictly larger divergence from the reference:

$$\mathbb{D}_{\text{KL}}\left[\pi_{\text{ref}} \| \pi^*_{\mathcal{D}'}\right] \ > \ \mathbb{D}_{\text{KL}}\left[\pi_{\text{ref}} \| \pi^*_{\mathcal{D}}\right].$$

Remark 1. *By Theorem 3.1, decreasing the data sampling temperature by a factor of $\alpha > 1$, i.e., drawing y_w and y_ℓ from $\pi_w(y)^\alpha$ and $\pi_\ell(y)^\alpha$, amplifies preference strength by the same factor α.*

Remark 2. *To recover $\pi^*_{\mathcal{D}}$ by optimizing a stronger dataset $\mathcal{D}'$ using the log-ratio reward $r' = \beta' \log \frac{\pi}{\pi_{\text{ref}}}$, one must increase the KL-penalty strength such that $\beta' = \beta \cdot \alpha$ for $\alpha > 1$.*

Thus, the effective KL-budget of DPO depends not only on β, but also on the strength of the preference data. Datasets with weak or noisy preference labels (*i.e.*, $p^*(y_1 \succ y_2) \approx 0.5$) constrain DPO to remain near π_{ref}, where large deviations cannot be justified by the evidence. In contrast, stronger preference labels act as a divisor on the log-ratio reward (Remark 2), enabling larger departures from π_{ref}. Therefore, the DID power-law allows DPO to balance conservatism and exploration, by linking policy deviation directly to the strength or quality of preference data.[5]

4.3 EXPERIMENTS

We first validate Theorems 4.1 and 4.2 using the Energy-Based Model (EBM) experiment from Section 3.3. All policies (reference, rejected, chosen) are parameterized by independent logits drawn from a normal distribution. The dataset is constructed from preference pairs sampled as in Section 3.3, and we train policies using the DPO objective with varying β values.

Verifying log-likelihood change. To test Theorem 4.1, we consider two cases: (1) chosen responses sampled from the reference ($y_w \sim \pi_{\text{ref}}$), and (2) rejected responses sampled from the reference ($y_\ell \sim \pi_{\text{ref}}$). We then track the change in average log-likelihoods under DPO training.

Figure 2 confirms the predictions of Theorem 4.1. When $y_w \sim \pi_{\text{ref}}$, the converged policy π^* decreases $\mathbb{E}[\log \pi(y_\ell)]$ relative to π_{ref}. Conversely, when $y_\ell \sim \pi_{\text{ref}}$, DPO increases $\mathbb{E}[\log \pi(y_w)]$ for $\beta \geq 1$. The same trend can also be observed in real-world settings using `Qwen3-4B` (Figure 3). We find that the direction of log-likelihood change of DPO ($\beta = 1.0$) is consistent with Theorem 4.1 on three datasets (Magpie-Pro, Magpie-G27, Ultra-Feedback; 50k samples each).

These results show that **the consistent log-likelihood shifts observed in DPO can be rigorously explained and precisely predicted based on the power-law DID relationship.**

[5] This is consistent with empirical trends from Wu et al. (2024) where easily discriminated preference pairs require higher β values to prevent overfitting, while noisy pairs require lower β values for aggressive updates.

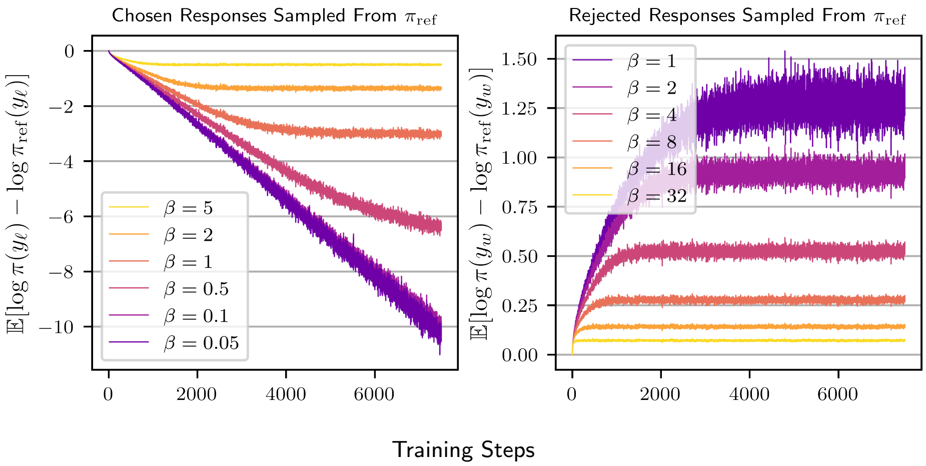

Figure 2: Log-likelihood change during DPO training. When chosen responses y_w are sampled from π_{ref}, the log-likelihood of rejected responses y_ℓ decreases relative to π_{ref} (left plot). When rejected samples y_ℓ are sampled from π_{ref}, the log-likelihood of chosen responses increases for $\beta \geq 1$ (right plot). This confirms the predicted change in log-likelihood of DPO (Theorem 4.1).

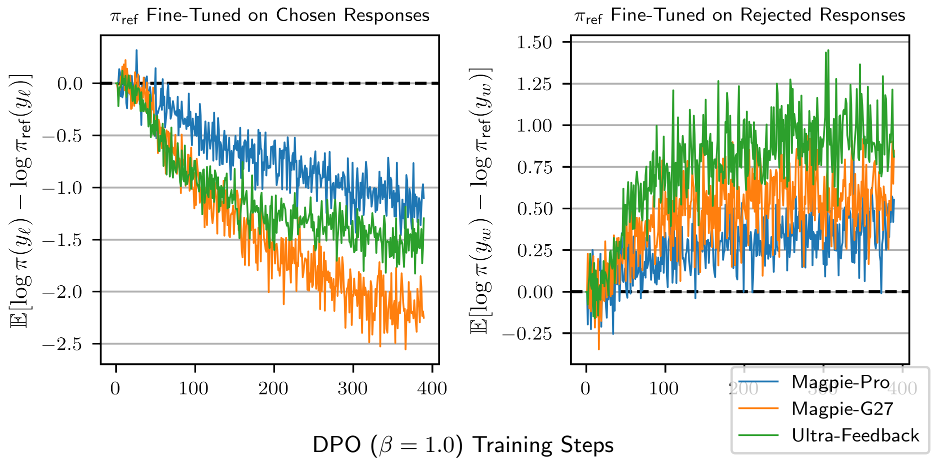

Figure 3: Log-likelihood change of real-world DPO training. On three datasets (Magpie-Pro, Magpie-G27, Ultra-Feedback; 50k samples each) and using `Qwen3-4B`, the reference policy π_{ref} is first fine-tuned for one epoch on either chosen or rejected responses, followed by one epoch of DPO ($\beta = 1.0$). Consistent with Theorem 4.1, DPO decreases the log-probability of rejected responses when π_{ref} is fine-tuned on chosen responses (left), and increases the log-probability of chosen responses when π_{ref} is fine-tuned on rejected responses (right).

Verifying policy exploration. To test Theorem 4.2, we generate preference datasets with varying sharpness. Specifically, we compare a fixed dataset $\mathcal{D}$ with a sharpened version $\mathcal{D}'$ that halves the sampling temperature of preference pairs (Remark 1), increasing the preference strength to $\alpha = 2$. We train policies on both datasets using the DPO objective with various β values, and track the KL-divergence $\mathbb{D}_{\text{KL}}[\pi_{\text{ref}} \| \pi]$ throughout the training process.

Figure 4 shows that a stronger preference signal $(\mathcal{D}', \alpha = 2)$ consistently yields larger divergence from π_{ref} than a weaker one $(\mathcal{D}, \alpha = 1)$, for the same β. Moreover, increasing the KL-penalty to $\beta' = 2\beta$ for the sharpened dataset $\mathcal{D}'$ results in a converged policy that matches the KL-divergence of the original dataset $\mathcal{D}$, consistent with Remark 2. This confirms that **policy exploration in DPO is jointly governed by the KL-penalty weight β and the implicit strength of preference data.**

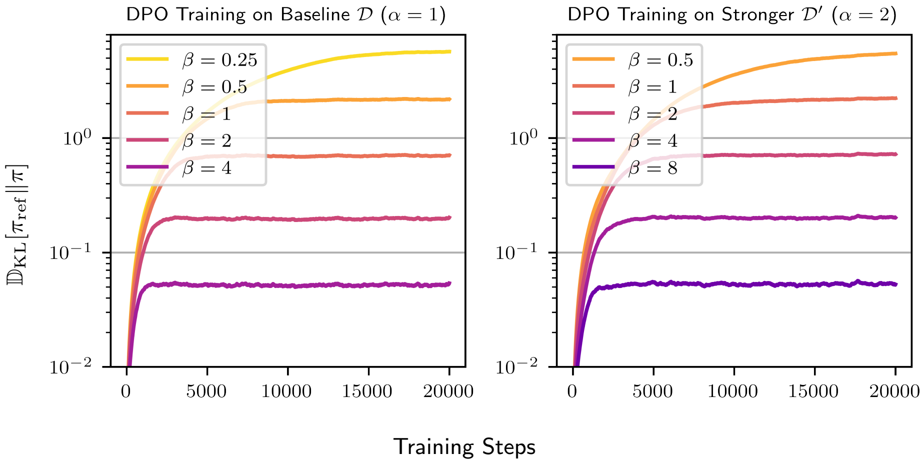

Figure 4: Policy exploration of DPO. Compared to the original dataset $\mathcal{D}$, halving the sampling temperature to form $\mathcal{D}'$ strengthens preferences ($\alpha = 2$) and increases the KL-divergence from π_{ref} under the same KL-penalty β, consistent with Theorem 4.2. Increasing the KL-penalty to 2β when training on $\mathcal{D}'$ restores the divergence to the level obtained with $\mathcal{D}$ using β, in line with Remark 2.

5 Training dynamics and learned capabilities

In this section, we examine how log-likelihood dynamics shape downstream performance by analyzing the properties of the learned Differential Information. We first study the empirical link between the *log-likelihood displacement* (LLD) phenomenon and downstream capabilities (Section 5.1). We then show how the Shannon entropy of the Differential Information Distribution can help explain the observed trade-off, where different training dynamics lead to distinct capabilities (Section 5.2).

5.1 A case study on log-likelihood displacement

Log-likelihood displacement (LLD) refers to the phenomenon where the log-likelihood of chosen response decreases during DPO training, even as alignment improves (Rafailov et al., 2024; Razin et al., 2024; Shi et al., 2024). Preventing LLD has been shown to improve performance on benchmarks such as MMLU (Hendrycks et al., 2021), which requires verifiable, ground-truth answers (Shi et al., 2024; Chen et al., 2024; Xiao et al., 2024a). While prior work investigates the cause of LLD through sample similarity or gradient dynamics (Pal et al., 2024; Razin et al., 2024; Feng et al., 2024), it doesn't fully explain why preventing LLD results in learning different capabilities.

To investigate this gap, we conduct a case study on how LLD affects downstream performance. We compare standard DPO with an another method utilizing projected gradient descent to prevent LLD while still optimizing the DPO objective, which we term **DPO-PG** (Appendix K). All runs start from the same π_{ref} fine-tuned on chosen responses. We train `Mistral7B-v0.3` (Jiang et al., 2024) and `Qwen3-4B` (Team, 2025) on two instruction-following datasets (Magpie-Pro and Magpie-G27), and evaluate on open-ended instruction-following (Arena-Hard (Li* et al., 2024), Wild-Bench (Lin et al., 2024a)) and a suite of eight knowledge-intensive QA tasks (details in Appendix M.2).

As shown in Table 1, across both model architectures, preventing LLD (DPO-PG) consistently yields the strongest performance on knowledge-intensive QA, at the expense of open-ended instruction-following. Standard DPO shows the opposite pattern, excelling on open-ended tasks but underperforming on factual QA. The same trend appears on Magpie-G27 (Table 4, Figure 16). This raises the question: *why is LLD associated with a trade-off between factual QA and open-ended tasks?*

5.2 Connecting policy dynamics with learned capabilities

We hypothesize that this trade-off reflects the properties of the information learned from training. To measure this, we use the Shannon entropy of the DID which quantifies the uncertainty or con-

Table 1: Impact of log-likelihood displacement (LLD) on downstream capabilities using the Magpie-Pro dataset. For open-ended instruction-following, we report Arena-Hard-v0.1 win-rate (AH, [%]) and Wild-Bench-v2 ELO score (WB). For knowledge-intensive QA, we report mean reciprocal rank across 8 QA benchmarks (QA). DID entropy (Ent., [nats]) is estimated via importance sampling (Appendix D.5). Compared to standard DPO, preventing LLD (DPO-PG) learns a low-entropy DID, which enhances factual accuracy but reduces performance on open-ended tasks.

		`Mistral7B-v0.3`				`Qwen3-4B`			
Method	β	Ent.	AH (↑)	WB (↑)	QA (↑)	Ent.	AH (↑)	WB (↑)	QA (↑)
DPO	1.0	1123.2	19.1	1141.5	0.53	9158.4	30.7	1134.4	0.49
	0.2	1303.4	18.5	1145.3	0.26	4765.9	28.1	1146.2	0.34
	0.1	1253.9	**23.4**	**1146.6**	0.28	7663.3	27.5	1148.2	0.27
	0.05	970.2	22.4	1145.3	0.30	6801.2	**43.7**	**1164.5**	0.24
DPO-PG		495.1	19.6	1129.9	**0.92**	388.2	37.4	1148.5	**0.94**

centration of Differential Information that drives policy updates. The DID entropy is defined as

$$H(q_{\pi^*/\pi_{\text{ref}}}) = -\sum_{y\in\mathcal{Y}} q_{\pi^*/\pi_{\text{ref}}}(y)\log q_{\pi^*/\pi_{\text{ref}}}(y).$$

Intuitively, low DID entropy indicates that Differential Information is concentrated on a narrow subset of samples, while high entropy suggests it is distributed more broadly. Appendices D.4 and D.5 detail how DID entropy reflects the properties of Differential Information and how it can be estimated using importance sampling.

As shown in Table 1, the DID entropy (column "Ent.") is observed to be significantly lower when preventing LLD (DPO-PG) compared to standard DPO. We hypothesize that LLD reflects changes in the output distribution that increase the entropy of the learned DID. Since chosen responses y_w typically lie in high-probability regions of the reference policy, decreasing $\log \pi(y_w)$ would smooth these probability peaks, yielding a more high-entropy DID. Conversely, increasing $\log \pi(y_w)$ sharpens these peaks, concentrating probability mass and reducing DID entropy (Appendix H).

Comparing the DID entropy and downstream performance in Table 1, we observe that factual QA performance is associated with low-entropy DID, while open-ended task performance is associated with high-entropy DID. This aligns with intuition: factual queries (*e.g.*, "What is the capital of France?") admit only a narrow set of correct answers (Lee et al., 2023; Xiang et al., 2025), concentrating Bayesian evidence on a small subset of $\mathcal{Y}$. In contrast, open-ended prompts (*e.g.*, "Write a story about a dragon.") admit a wide variety of valid responses (Li et al., 2025; Gu et al., 2024), dispersing Bayesian evidence more broadly across $\mathcal{Y}$. These results suggest that learning low-entropy DID enhances factual precision, while learning high-entropy DID improves open-ended tasks.

In summary, we observe that preventing LLD induces the model to learn a low-entropy DID, which improves accuracy on factual QA tasks. In contrast, allowing LLD results in learning a high-entropy DID that enhances open-ended tasks. These observations suggest that log-likelihood dynamics reflect the type of information learned during alignment.

6 Conclusion

We introduced a Bayesian perspective on Direct Preference Optimization (DPO) through the lens of Differential Information Distribution (DID). We showed that DPO's log-ratio reward is the unique Bradley-Terry reward that learns the target policy when preferences encode Differential Information. We further demonstrated that DPO's characteristic training dynamics (log-likelihood shifts and adaptive policy exploration) stem from a power-law DID relationship. We finally introduced DID entropy as a principled measure of uncertainty in the learned information, clarifying the trade-off between log-likelihood displacement and downstream performance: high-entropy DID smooths the output distribution and aids open-ended instruction-following, while low-entropy DID concentrates probability mass and benefits knowledge-intensive QA. Together, our findings provide both a principled theoretical foundation and practical guidance for preference-based alignment.

7 Reproducibility statement

To ensure the reproducibility of our work, we provide comprehensive details on our theoretical and empirical findings. For our theoretical results, detailed proofs for all theorems and corollaries are available in Appendix J. For our empirical validation, Appendix M contains a full description of our experimental setups. Specifically, Appendix M.1 details the setup and hyper-parameters for our controlled experiments using Energy-Based Models. Appendix M.2 describes the details for preparing the Magpie-G27 dataset, the training configurations for both DPO and our DPO-PG method, and the evaluation protocols for the real-data experiments in Section 5.2. In the supplementary material, we include the training code for the EBM experiments, raw evaluation results for the real-data experiments, and a reference Pytorch implementation of the DPO-PG method (Appendix K). We plan to release all model checkpoints and the complete code for training and evaluation upon acceptance to ensure direct replication of our findings.

References

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.

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*, pp. 7432–7439. AAAI Press, 2020. URL https://aaai.org/ojs/index.php/AAAI/article/view/6239.

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/.

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.

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 (eds.), *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.

Jay Hyeon Cho, JunHyeok Oh, Myunsoo Kim, and Byung-Jun Lee. Rethinking dpo: The role of rejected responses in preference misalignment, 2025. URL https://arxiv.org/abs/2506.12725.

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 (eds.), *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)*, pp. 2924–2936, Minneapolis, Minnesota, 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1300. URL https://aclanthology.org/N19-1300.

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.

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.

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.

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.

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.

Zihui Gu, Xingwu Sun, Fengzong Lian, Zhanhui Kang, Cheng-Zhong Xu, and Ju Fan. Diverse and fine-grained instruction-following ability exploration with synthetic data, 2024. URL https://arxiv.org/abs/2407.03942.

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, 2024a. URL https://openreview.net/forum?id=LNLjU5C5dK.

Shangmin Guo, Biao Zhang, Tianlin Liu, Tianqi Liu, Misha Khalman, Felipe Llinares, Alexandre Rame, Thomas Mesnard, Yao Zhao, Bilal Piot, Johan Ferret, and Mathieu Blondel. Direct language model alignment from online ai feedback, 2024b. URL https://arxiv.org/abs/2402.04792.

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

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.

Rei Higuchi and Taiji Suzuki. Direct density ratio optimization: A statistically consistent approach to aligning large language models. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=jDdaysR6bz.

Ari Holtzman, Jan Buys, Maxwell Forbes, Antoine Bosselut, David Golub, and Yejin Choi. Learning to write with cooperative discriminators. In Iryna Gurevych and Yusuke Miyao (eds.), *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1638–1649, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-1152. URL https://aclanthology.org/P18-1152/.

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.

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.

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.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun (eds.), *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.

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 (eds.), *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*, 2022a. URL http://papers.nips.cc/paper_files/paper/2022/hash/67496dfa96afddab795530cc7c69b57a-Abstract-Conference.html.

Tomasz Korbak, Ethan Perez, and Christopher L Buckley. Rl with kl penalties is better viewed as bayesian inference, 2022b. URL https://arxiv.org/abs/2205.11275.

Ben Krause, Akhilesh Deepak Gotmare, Bryan McCann, Nitish Shirish Keskar, Shafiq Joty, richard socher, and Nazneen Rajani. Gedi: Generative discriminator guided sequence generation. In *ACL Rolling Review - May 2021*, 2021. URL https://openreview.net/forum?id=3E3F51B8azH.

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.

Nayeon Lee, Wei Ping, Peng Xu, Mostofa Patwary, Pascale Fung, Mohammad Shoeybi, and Bryan Catanzaro. Factuality enhanced language models for open-ended text generation, 2023. URL https://arxiv.org/abs/2206.04624.

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/.

Ziniu Li, Congliang Chen, Tian Xu, Zeyu Qin, Jiancong Xiao, Zhi-Quan Luo, and Ruoyu Sun. Preserving diversity in supervised fine-tuning of large language models, 2025. URL https://arxiv.org/abs/2408.16673.

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, 2024a. URL https://arxiv.org/abs/2406.04770.

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, 2024b. URL https://arxiv.org/abs/2411.19943.

Alisa Liu, Maarten Sap, Ximing Lu, Swabha Swayamdipta, Chandra Bhagavatula, Noah A. Smith, and Yejin Choi. DExperts: Decoding-time controlled text generation with experts and anti-experts. In Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli (eds.), *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International*

Joint Conference on Natural Language Processing (Volume 1: Long Papers), pp. 6691–6706, Online, August 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-long.522. URL https://aclanthology.org/2021.acl-long.522/.

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, 2024a. URL https://arxiv.org/abs/2410.18451.

Chris Yuhao Liu, Liang Zeng, Yuzhen Xiao, Jujie He, Jiacai Liu, Chaojie Wang, Rui Yan, Wei Shen, Fuxiang Zhang, Jiacheng Xu, Yang Liu, and Yahui Zhou. Skywork-reward-v2: Scaling preference data curation via human-ai synergy. *arXiv preprint arXiv:2507.01352*, 2025a.

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, 2025b. URL https://arxiv.org/abs/2503.11701.

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

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

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.

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 (eds.), *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.

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 (eds.), *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.

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.

Yu Pan, Zhongze Cai, Guanting Chen, Huaiyang Zhong, and Chonghuan Wang. What matters in data for dpo?, 2025. URL https://arxiv.org/abs/2508.18312.

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

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 (eds.), *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.

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.

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.

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

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.

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.

Gemma Team. Gemma. 2024. doi: 10.34740/KAGGLE/M/3301. URL https://www.kaggle.com/m/3301.

Qwen Team. Qwen3 technical report, 2025. URL https://arxiv.org/abs/2505.09388.

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.

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.

Junkang Wu, Yuexiang Xie, Zhengyi Yang, Jiancan Wu, Jinyang Gao, Bolin Ding, Xiang Wang, and Xiangnan He. β-DPO: Direct preference optimization with dynamic β. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=ZfBuhzE556.

Violet Xiang, Charlie Snell, Kanishk Gandhi, Alon Albalak, Anikait Singh, Chase Blagden, Duy Phung, Rafael Rafailov, Nathan Lile, Dakota Mahan, Louis Castricato, Jan-Philipp Franken, Nick Haber, and Chelsea Finn. Towards system 2 reasoning in llms: Learning how to think with meta chain-of-thought, 2025. URL https://arxiv.org/abs/2501.04682.

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 (eds.), *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024a. URL http://papers.nips.cc/paper_files/paper/2024/hash/cf8b2205e39f81726a8d828ecbe00ad0-Abstract-Conference.html.

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, 2024b. URL https://arxiv.org/abs/2410.15595.

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, 2024a. URL https://openreview.net/forum?id=51iwkioZpn.

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, 2024b. URL https://arxiv.org/abs/2406.08464.

Kevin Yang and Dan Klein. FUDGE: Controlled text generation with future discriminators. In Kristina Toutanova, Anna Rumshisky, Luke Zettlemoyer, Dilek Hakkani-Tur, Iz Beltagy, Steven Bethard, Ryan Cotterell, Tanmoy Chakraborty, and Yichao Zhou (eds.), *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 3511–3535, Online, June 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.naacl-main.276. URL https://aclanthology.org/2021.naacl-main.276/.

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 (eds.), *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 4791–4800, Florence, Italy, 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1472. URL https://aclanthology.org/P19-1472.

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, 2023a. URL https://arxiv.org/abs/2304.11277.

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, 2023b. URL https://arxiv.org/abs/2305.10425.

CONTENTS OF APPENDIX

A LLM USAGE DISCLOSURE

In accordance with the ICLR 2026 policy,[6] we disclose that large language models were used for minor writing assistance and polishing. All research ideations, technical derivations, and experiments were planned and carried out by the human authors.

B LIMITATIONS

While our perspective offers novel insights, we acknowledge limitations for future work. First, Theorem 2.1, established from prior work (Dumoulin et al., 2023), assumes sufficient data coverage and train-test generalization. Second, the connection between DID entropy and policy dynamics (Claim H.1) is qualitative and based on information-theoretic intuition (Appendix H); despite experimental support (Section 5.2), a formal treatment would strengthen this aspect of our work.

Assumptions on the Reference Policy. Our general Bayesian formulation of Direct Preference Optimization (DPO) in Section 3 applies to any reference policy with full-support. However, our analytical derivation of training dynamics in Section 4 assumes the setting where the reference policy π_{ref} matches either the chosen or rejected response distribution (*i.e.*, $\pi_{\text{ref}} = \pi_w$ or $\pi_{\text{ref}} = \pi_\ell$). This assumption reflects standard practice: fine-tuning π_{ref} on chosen/rejected responses (Rafailov et al., 2023; 2024) or using on-policy generations for preference data (Meng et al., 2024; Guo et al., 2024b). Importantly, our theoretical predictions match the empirical patterns observed in real-life DPO training (Figures 3, 10, 11), confirming that our analysis accurately captures practical behavior. While scenarios may arise where π_{ref} differs substantially from the underlying preference distributions, extending our analysis to such cases and studying their convergence properties is a promising direction for future work. For such alternative choices of reference policy or different preference-sampling assumptions, the power-law DID result in Corollary 3.2.1 still holds, as long as closed-form expressions for the reference policy or the preference-sample distribution under π_w and π_ℓ can be obtained. This allows the mathematical analysis in Section 4 to be applied in a similar manner.

Real-World Optimality of DPO. It is worth noting that Theorem 3.2 should not be interpreted as claiming universal optimality of DPO on real data; rather, it highlights DPO's unique optimality in learning Differential Information, which may help explain its widespread adoption across domains. While verifying DPO's optimality on real-world data is infeasible without access to the global optimal policy, **we do not claim that DPO is optimal for arbitrary datasets**. In our framework, π^* represents a target policy we aim to learn, which is not necessarily the globally optimal policy. Furthermore, Theorem 3.2 establishes DPO's optimality within a Bayesian framework where preferences encode the *Differential Information* required to update π_{ref} into π^*. When real-world data deviates from this assumption, DPO is not expected to recover a globally optimal policy. Instead, it converges to a policy satisfying the power-law DID relationship (Corollary 3.2.1). If this converged policy empirically improves over π_{ref}, it indicates that the preference data encodes the Differential Information necessary for policy improvement. DPO's strong empirical performance across diverse domains (Xiao et al., 2024b) suggests that this condition often holds in practice.

C RELATED WORK

Direct Preference Optimization. Direct Preference Optimization (DPO) (Rafailov et al., 2023) is widely used to align LMs with human preferences in a supervised manner (Xiao et al., 2024b; Liu et al., 2025b). Recent research investigates the theoretical foundations of preference optimization, connecting it to distribution matching (Korbak et al., 2022a; Dumoulin et al., 2023; Xu et al., 2024a; Liu et al., 2024b; Ji et al., 2024), and analyzing the optimization dynamics of log-likelihood

[6] https://iclr.cc/Conferences/2026/AuthorGuide

displacement (Pal et al., 2024; Feng et al., 2024; Mao et al., 2024). While Chen et al. (2024) reinterpret the DPO objective from a noise contrastive estimation perspective, their approach relies on the optimal policy of the KL-regularized RL objective and leaves its justification open for discussion.

We complement prior work by offering a Bayesian perspective on the justification for the reward parameterization of DPO, linking its optimality to the Differential Information captured by the preference data. This perspective explains the training properties of DPO, and also yields a novel interpretation of log-likelihood displacement, relating it to the entropy of the learned DID.

Bayesian perspective of KL-regularized RL. A prior work done by Korbak et al. (2022b) interprets the optimal policy of the KL-regularized RL objective $\pi^*(y) \propto \pi_{\text{ref}}(y)\exp(r(y)/\beta)$ from a Bayesian perspective, showing how the reward-induced distribution $P(Y = y \mid r) \propto \exp(r(y))$ can be viewed as carrying the Bayesian-evidence towards a target policy. Because DPO learns the same optimal policy using supervised learning, there exists an inherent connection between this view and our DID perspective. Our work builds on that connection but provides a Bayesian account of DPO, characterizing the statistical structure of preference datasets, the optimality of the DPO log-ratio reward, and the resulting policy-dynamics phenomena.

D Interpretation of Differential Information Distribution

This section provides a probabilistic interpretation of the Differential Information Distribution (DID). Our goal is to illustrate the intuition that the DID $q_{\pi^*/\pi_{\text{ref}}}$ represents the distribution over samples y that carry the Differential Information needed to update the reference policy π_{ref} into the target policy π^* through Bayesian conditioning.

D.1 Information as an abstract event

We begin by establishing a Bayesian framework to reason about information associated with sentences. Consider the sample space $\mathcal{Y}$ of all possible sentences, assuming a uniform prior distribution $P(Y = y) = 1/|\mathcal{Y}|$.

Now, consider 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. For instance:

1. If $P(X \mid Y = y)$ measures the probability of "y being a mathematically correct sentence", then the probabilities will be either 0 or 1.
   - $P(X \mid Y =$"`1+1=2`"$) = 1$
   - $P(X \mid Y =$"`1+0=1`"$) = 1$
   - $P(X \mid Y =$"`2+2=5`"$) = 0$
2. If $P(X \mid Y = y)$ measures the probability of "y being a safe sentence", then the probabilities can be in the range of $0 \leq P(X \mid Y = y) \leq 1$.
   - $P(X \mid Y =$"`Apples are red.`"$) = 0.99$
   - $P(X \mid Y =$"`Alcohol is good for relaxation.`"$) = 0.3$
   - $P(X \mid Y =$"`Let's promote violence!`"$) = 0$

A crucial assumption in our analysis 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 `Qwen3-4B`; 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 $Y = y$:

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

This 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.

Note that similar independence assumptions are standard in adjacent areas. For example, in Controlled Text Generation, *Weighted Decoding* methods (Holtzman et al., 2018; Krause et al., 2021; Liu et al., 2021; Yang & Klein, 2021) factor the controlled distribution into the base model's prior distribution and a control-dependent likelihood, effectively mirroring our conditional independence formulation.

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

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 that holds X is directly proportional to the likelihood that sentence y possesses the property X. Sentences that strongly exhibit property X (*i.e.*, high $P(X \mid Y = y)$) are more likely to be sampled from $P(Y = y \mid X)$.

D.3 Information difference between policies

We now focus on comparing two language models, π^* and π_{ref}, both assumed to have full support over $\mathcal{Y}$. We are interested in the difference in the information contained in these two models. We characterize this information difference as the Bayesian evidence required to update π_{ref} into π^*. We represent such information by an abstract event X which we will call the Differential Information that updates π_{ref} into π^*. We seek an X such that conditioning $\pi_{\text{ref}}(y)$ on X yields $\pi^*(y)$. 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 prior π_{ref} given the sentence y:

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

In other words, the probability that a sentence y holds the information X does not depend on whether it was sampled from π_{ref}.

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:

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

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 samples 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:

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

□

Therefore, sampling a sentence from the normalized ratio distribution $q_{\pi^*/\pi_{\text{ref}}}$ is equivalent to sampling a sentence that carries the Differential Information required to update π_{ref} into π^* via Bayes' rule.

D.4 Uncertainty of Differential Information

The normalized ratio form of the DID $q_{\pi^*/\pi_{\text{ref}}}$ naturally admits an information-theoretic characterization. In particular, we can measure the uncertainty of the Differential Information by the Shannon entropy:

$$H(q_{\pi^*/\pi_{\text{ref}}}) = -\sum_y q_{\pi^*/\pi_{\text{ref}}}(y) \log q_{\pi^*/\pi_{\text{ref}}}(y).$$

This entropy quantifies how broadly the Bayesian evidence required to update π_{ref} into π^* is distributed across the sample space $\mathcal{Y}$.

A low-entropy DID $H(q_{\pi^*/\pi_{\text{ref}}})$ describes a deterministic Bayesian evidence that drives the update from π_{ref} to π^*. Intuitively, if only a few samples hold that Bayesian evidence, then the DID $q_{\pi^*/\pi_{\text{ref}}}$ will be highly concentrated on a few samples that have a large enough value of $P(X \mid Y = y)$ (*i.e.*, the probability of y having X, Appendix D.2). Therefore, the policy update from π_{ref} to π^* can effectively be explained by a few characteristic samples. This corresponds to information that is specific and localized, such as factual knowledge (*e.g.*, the birthplace of George Washington) where only a narrow subset of $\mathcal{Y}$ strongly supports the relevant property.

Conversely, a high-entropy DID $H(q_{\pi^*/\pi_{\text{ref}}})$ describes an uncertain Bayesian evidence that drives the update from π_{ref} to π^*. If the evidence is spread across many possible samples, then the DID $q_{\pi^*/\pi_{\text{ref}}}$ will also be more spread-out and flatter. No single sample dominantly holds a high enough value of $P(X \mid Y = y)$, and the policy update requires a Bayesian evidence from a wide variety of samples. This corresponds to information that is general and broadly distributed, such as open-ended instruction-following (*e.g.*, writing a story about dragons) where many different completions may plausibly express the property.

Therefore, the DID entropy provides a principled measure of how uncertain or spread-out the Differential Information is across the sample space.

D.5 Estimation of DID entropy

To measure the Shannon entropy of the Differential Information Distribution (DID):

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

we can first estimate the log-partition function $\log Z = \log \sum_{y\in\mathcal{Y}} \frac{\pi(y)}{\pi_{\text{ref}}(y)} = \log \mathbb{E}_{y\sim\pi}[\frac{1}{\pi_{\text{ref}}(y)}]$, and then estimate the remainder term via self-normalized importance sampling. For Tables 1 and 4, we estimate the two terms in the following steps:

- To estimate $\log Z$, we sample $K = 32$ completions from π and use the log-sum-exp trick to directly estimate $\log Z \approx \log \sum_{i=1}^{K} \exp(-\log \pi_{\text{ref}}(y_i)) - \log K$.
- To estimate $\mathbb{E}_{y\sim q_{\pi/\pi_{\text{ref}}}}[\log \frac{\pi(y)}{\pi_{\text{ref}}(y)}]$, we draw 32 samples from π and re-weight them by $1/\pi_{\text{ref}}(y)$, which is proportional to the importance weight $q_{\pi/\pi_{\text{ref}}}(y)/\pi(y)$.

The 95% confidence intervals for the estimated DID entropy values are listed in Table 5. Overall, the difference in DID entropy across methods are statistically significant.

Note that naive auto-regressive sampling from the token-level ratio distribution is ineffective due to out-of-distribution prefixes, leading to degenerate outputs. While this method is sound for tractable output spaces, it does not scale to LLMs. This is the very reason behind our importance-sampling based approach for estimating the DID which is proportional to the sequence-level probability ratio. Alternative estimators (*e.g.*, token-level or length-normalized DID entropy) have an unclear theoretical interpretation and is not straightforward to empirically estimate. For instance, consider the length-normalized DID entropy. Normalizing the $\log Z$ term or the log-ratio term $\mathbb{E}_{y\sim q_{\pi/\pi_{\text{ref}}}}[\log \frac{\pi(y)}{\pi_{\text{ref}}(y)}]$ by token length has no clear interpretation within the DID framework and, most crucially, does not accurately measure the uncertainty in the DID defined over token *sequences*.

E Empirical comparison of DPO and reference-free rewards

In this section, we test whether real-world preference datasets align more closely with the DPO log-ratio reward $r = \beta \log(\pi/\pi_{\text{ref}})$ or with the reference-free reward $r = \beta \log \pi$ commonly used in SimPO (Meng et al., 2024) or CPO (Xu et al., 2024a).

For a Bradley-Terry preference distribution p^* and the target policy π^*, the reference-free reward assumes that preferences directly encode the target policy:

$$p^*(y) \propto \pi^*(y)^\beta.$$

In contrast, the DPO log-ratio reward assumes that preferences encode *Differential Information* needed to update π_{ref} into π^*:

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

Experimental Setup We trained `Mistral-7B-v0.3` on both reward forms using the Magpie-Pro and Ultra-Feedback datasets. For evaluation, we measured the average reward of sampled completions on 1,000 unseen prompts across a broad range of hyper-parameters, $\beta \in \{0.2, 0.1, 0.05, 0.02\}$, and over three training epochs. The reward scores were estimated using an off-the-shelf reward model `Skywork-Reward-V2-Llama-3.1-8B` (Liu et al., 2025a).

Results Table 2 shows the expected reward improvements in each setting, computed as the difference between the average reward of a policy and that under the reference policy. Formally, for a reward model $R : \mathcal{Y} \to \mathbb{R}$, we report $\mathbb{E}_{y\sim\pi}[R(y)] - \mathbb{E}_{y_{\text{ref}}\sim\pi_{\text{ref}}}[R(y_{\text{ref}})]$ to highlight the relative

Table 2: Comparison of expected reward improvements over π_{ref} for policies trained with the DPO log-ratio reward $r = \beta \log(\pi/\pi_{\text{ref}})$ versus the reference-free reward $r = \beta \log \pi$. The largest reward improvement within the same reward parameterization form is highlighted in bold. The log-ratio reward consistently produces policies with higher expected rewards, indicating that instruction-following preferences primarily encode Differential Information rather than the target policy itself.

Dataset	Reward	β	Epoch-1	Epoch-2	Epoch-3
Ultra-Feedback	$r = \beta \log \pi/\pi_{\text{ref}}$	0.2	5.08	5.73	6.81
		0.1	6.31	6.04	6.28
		0.05	7.53	7.95	8.15
		0.02	**8.20**	8.07	6.04
	$r = \beta \log \pi$	0.2	2.65	2.60	**3.77**
		0.1	1.83	1.13	1.92
		0.05	0.34	0.54	1.50
		0.02	0.50	-1.39	-2.37
Magpie-Pro	$r = \beta \log \pi/\pi_{\text{ref}}$	0.2	6.30	6.25	5.62
		0.1	**7.41**	6.67	6.17
		0.05	7.17	6.85	6.13
		0.02	7.27	6.78	2.60
	$r = \beta \log \pi$	0.2	1.11	0.80	0.19
		0.1	1.81	1.81	2.93
		0.05	-1.01	2.13	-4.52
		0.02	**3.41**	-0.87	-5.67

improvement over π_{ref}. Overall, the DPO log-ratio reward consistently produces responses with higher expected reward than the reference policy, even outperforming the reference-free reward. **This provides empirical evidence that real-world preference data frequently encodes the *Differential Information* needed to improve π_{ref} towards π^*, rather than directly encoding the target policy π^* itself.**

F Optimal Dataset for DPO

A central design choice when building DPO datasets is *how* to sample the chosen and rejected responses. Prior work has advocated opposing strategies: *strong contrasts* that maximize quality gaps (Meng et al., 2024; Xu et al., 2024b) versus *fine-grained distinctions* with minimal differences (Lin et al., 2024b; Tunstall et al., 2023; Guo et al., 2024a). We resolve this tension by showing that what matters is not *absolute* gap size but the *Differential Information* encoded by the pair (y_w, y_ℓ). In particular, the optimal rejection distribution should make the dataset's Differential Information distribution reflect the Differential Information between the reference and target policies. Using Corollary 3.2.1 we obtain the following closed-form characterization:

Theorem F.1 (Optimal Distribution of Chosen and Rejected Responses)**.** *Given a preference dataset $\mathcal{D} = \{(y_w, y_\ell) \mid y_w \sim \pi_w, y_\ell \sim \pi_\ell\}$, if $\pi_{\text{ref}} = \pi_w$, then preference optimization on $\mathcal{D}$ using the log-ratio reward $r = \beta \log \pi/\pi_{\text{ref}}$ converges to π^* if and only if the rejected sample distribution π_ℓ satisfies*

$$\pi_\ell(y) \propto \pi_{\text{ref}}(y) \left(\frac{\pi_{\text{ref}}(y)}{\pi^*(y)}\right)^\beta, \quad \forall y \in \mathcal{Y}.$$

Likewise, if $\pi_{\text{ref}} = \pi_\ell$, then optimizing $\mathcal{D}$ using the log-ratio reward converges to π^* if and only if the chosen sample distribution π_w satisfies

$$\pi_w(y) \propto \pi_{\text{ref}}(y) \left(\frac{\pi^*(y)}{\pi_{\text{ref}}(y)}\right)^\beta, \quad \forall y \in \mathcal{Y}.$$

Intuitively, Theorem F.1 states that the correct construction of preference data depends on matching the dataset's DID to the log-ratio reward used in DPO. Thus both "strong" and "fine-grained" constructions can be optimal, given that the DID from π_ℓ to π_w aligns with the DID from π_{ref} to π^*, up to the exponent $\beta > 0$.

Proof. This directly follows from Corollary 3.2.1. For any general preference dataset $\mathcal{D} = \{(y_w, y_\ell) \mid y_w \sim \pi_w, y_\ell \sim \pi_\ell\}$, the Bradley-Terry preference distribution p^* must exactly follow q_{π_w/π_ℓ}. Corollary 3.2.1 states that a power-law DID structure involving the converged policy π^* must hold:

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

When $\pi_{\text{ref}} = \pi_w$, for all $y \in \mathcal{Y}$, we have

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

Conversely, when $\pi_{\text{ref}} = \pi_\ell$, for all $y \in \mathcal{Y}$, we have

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

□

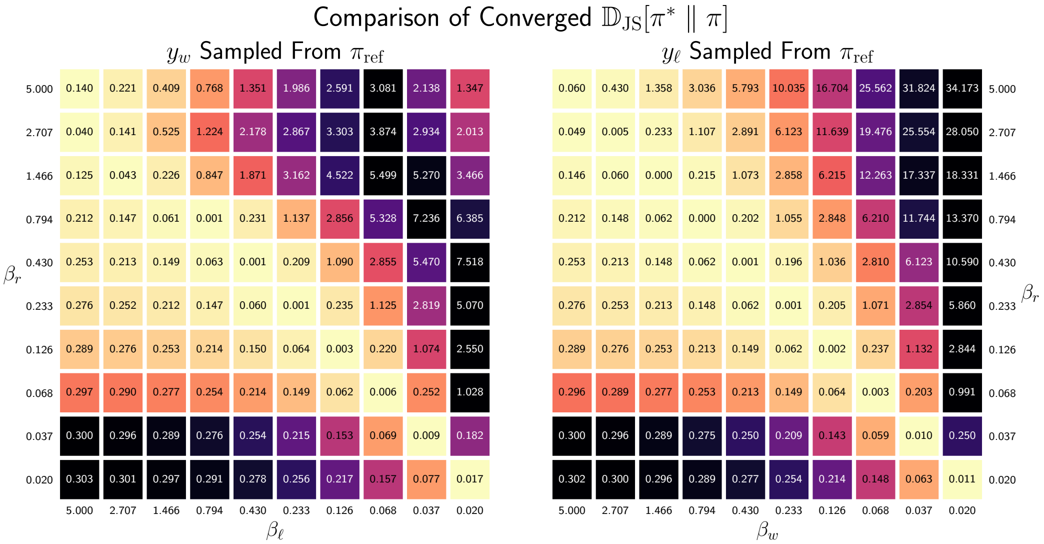

Figure 5: Convergence quality (Jensen-Shannon divergence) between the target π^* and the converged policy π under varying dataset exponents β_ℓ and β_w (controlling π_ℓ and π_w respectively), and reward scale β_r. Consistent with Theorem F.1, the best convergence occurs near the diagonal $\beta_\ell = \beta_r$ and $\beta_w = \beta_r$.

We can validate Theorem F.1 using the EBM experiments described in Section 3.3. To test Theorem F.1 we disentangle the exponent used to construct rejected samples from the scaling factor used in the DPO reward. We sample chosen responses y_w from π_{ref} and draw rejected responses y_ℓ from $\pi_\ell(y) \propto \pi_{\text{ref}}(y) \left(\frac{\pi_{\text{ref}}(y)}{\pi^*(y)}\right)^{\beta_\ell}$, while training with reward $r(y) = \beta_r \log \frac{\pi(y)}{\pi_{\text{ref}}(y)}$. We also test the setup where y_ℓ comes from π_{ref} and y_w is drawn from $\pi_w(y) \propto \pi_{\text{ref}}(y) \left(\frac{\pi^*(y)}{\pi_{\text{ref}}(y)}\right)^{\beta_w}$. Sweeping $\beta \in [0.02, 5]$, we measure $\mathbb{D}_{\text{JS}}[\pi^* \| \pi]$ for the converged policy. Figure 5 shows the minimum divergence concentrated along $\beta_\ell = \beta_r$ and $\beta_w = \beta_r$, showing that the optimal DPO dataset requires the DID from π_ℓ to π_w to align with that from π_{ref} to π^*, up to a positive exponent β.

G Log-margin dynamics of DPO

Based on the power-law DID relationship in DPO (Corollary 3.2.1), we can prove how a policy ordering $\pi^* \succ \pi_{\text{ref}} \succ \pi_\ell$ must exist based on increasing log-margins:

Theorem G.1 (Log-Margin Ordered Policies of DPO)**.** *Under the same setup as Theorem F.1, if $\pi_{\text{ref}} = \pi_w$, then the following ordering of policies based on increasing log-margins must hold:*

$$\begin{aligned}\mathbb{E}_{y_w \sim \pi_{\text{ref}}}\left[\log \pi^*(y_w)\right] - \mathbb{E}_{y_\ell \sim \pi_\ell}\left[\log \pi^*(y_\ell)\right] &> \mathbb{E}_{y_w \sim \pi_{\text{ref}}}\left[\log \pi_{\text{ref}}(y_w)\right] - \mathbb{E}_{y_\ell \sim \pi_\ell}\left[\log \pi_{\text{ref}}(y_\ell)\right] \\ &> \mathbb{E}_{y_w \sim \pi_{\text{ref}}}\left[\log \pi_\ell(y_w)\right] - \mathbb{E}_{y_\ell \sim \pi_\ell}\left[\log \pi_\ell(y_\ell)\right].\end{aligned}$$

Proof. Since we assume that $\pi_{\text{ref}} = \pi_w$, we have $\pi^*(y) \propto \pi_{\text{ref}}(y) \cdot (q_{\pi_{\text{ref}}/\pi_\ell}(y))^{\frac{1}{\beta}}$. Therefore, it follows that

$$\begin{aligned}&\mathbb{E}_{y_w \sim \pi_{\text{ref}}}\left[\log \pi^*(y_w)\right] - \mathbb{E}_{y_\ell \sim \pi_\ell}\left[\log \pi^*(y_\ell)\right] - \Big(\mathbb{E}_{y_w \sim \pi_{\text{ref}}}\left[\log \pi_{\text{ref}}(y_w)\right] - \mathbb{E}_{y_\ell \sim \pi_\ell}\left[\log \pi_{\text{ref}}(y_\ell)\right]\Big) \\ &\quad= \frac{1}{\beta}\mathbb{E}_{y_w \sim \pi_{\text{ref}}}\left[\log \pi_{\text{ref}}(y_w) - \log \pi_\ell(y_w)\right] - \frac{1}{\beta}\mathbb{E}_{y_\ell \sim \pi_\ell}\left[\log \pi_{\text{ref}}(y_\ell) - \log \pi_\ell(y_\ell)\right] \\ &\quad= \frac{1}{\beta}\mathbb{D}_{\text{KL}}\left[\pi_{\text{ref}} \| \pi_\ell\right] + \frac{1}{\beta}\mathbb{D}_{\text{KL}}\left[\pi_\ell \| \pi_{\text{ref}}\right] > 0.\end{aligned}$$

Thus we have proven the top inequality (1). Next, the bottom inequality (2) can be shown by the following fact:

$$\begin{gathered}\mathbb{D}_{\text{KL}}\left[\pi_{\text{ref}} \| \pi_\ell\right] > 0 > -\mathbb{D}_{\text{KL}}\left[\pi_\ell \| \pi_{\text{ref}}\right] \\ \Rightarrow \\ \mathbb{E}_{y_w \sim \pi_{\text{ref}}}\left[\log \pi_{\text{ref}}(y_w) - \log \pi_\ell(y_w)\right] > \mathbb{E}_{y_\ell \sim \pi_\ell}\left[\log \pi_{\text{ref}}(y_\ell) - \log \pi_\ell(y_\ell)\right].\end{gathered}$$

□

This directly yields an information-theoretic triangle inequality within the DPO framework.

Corollary G.1.1 (Information-Theoretic Triangle Inequality of DPO)**.** *Under the conditions of Theorem F.1, the following inequality holds:*

$$\mathbb{D}_{\text{KL}}\left[\pi_{\text{ref}} \| \pi_\ell\right] + \mathbb{D}_{\text{KL}}\left[\pi_\ell \| \pi^*\right] > \mathbb{D}_{\text{KL}}\left[\pi_{\text{ref}} \| \pi^*\right].$$

Proof. From Theorem G.1, it directly follows that

$$\begin{gathered}\mathbb{E}_{y_w \sim \pi_{\text{ref}}}\left[\log \pi^*(y_w)\right] - \mathbb{E}_{y_\ell \sim \pi_\ell}\left[\log \pi^*(y_\ell)\right] > \mathbb{E}_{y_w \sim \pi_{\text{ref}}}\left[\log \pi_\ell(y_w)\right] - \mathbb{E}_{y_\ell \sim \pi_\ell}\left[\log \pi_\ell(y_\ell)\right] \\ \iff \\ \mathbb{D}_{\text{KL}}\left[\pi_{\text{ref}} \| \pi_\ell\right] - \mathbb{D}_{\text{KL}}\left[\pi_{\text{ref}} \| \pi^*\right] > -\mathbb{D}_{\text{KL}}\left[\pi_\ell \| \pi^*\right] \\ \iff \\ \mathbb{D}_{\text{KL}}\left[\pi_{\text{ref}} \| \pi_\ell\right] + \mathbb{D}_{\text{KL}}\left[\pi_\ell \| \pi^*\right] > \mathbb{D}_{\text{KL}}\left[\pi_{\text{ref}} \| \pi^*\right].\end{gathered}$$

□

Although the KL-divergence does not generally satisfy a triangle inequality, Corollary G.1.1 shows that DPO enforces this specific triangle inequality for the trio $(\pi_{\text{ref}}, \pi_\ell, \pi^*)$.

Corollary G.1.1 establishes a fundamental lower bound in the information "cost" (KL-divergence) of learning π^* by contrasting π_{ref} against π_ℓ. It shows that the cost of updating π^* back into π_{ref} via π_ℓ must be larger than that of directly updating π^* into π_{ref}.

H Log-likelihood displacement and DID entropy

In this section, we provide a qualitative argument regarding the relationship between log-likelihood displacement (LLD) and DID entropy discussed in Section 5.2. In particular, we present the following informal claim.

Informal Claim H.1. *Consider a policy π derived from π_{ref} such that $\mathbb{D}_{\text{KL}}\left[\pi \| \pi_{\text{ref}}\right]$ is bounded. 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 Distribution: $H(q_{\pi/\pi_{\text{ref}}}) < H(\pi_{\text{ref}})$.*

- *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 Distribution: $H(q_{\pi/\pi_{\text{ref}}}) > H(\pi_{\text{ref}})$.*

Our assumptions is as follows:

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 $\mathbb{D}_{\text{KL}}\left[\pi_{\text{ref}} \| \pi\right] < c$.

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 $\mathbb{D}_{\text{KL}}\left[\pi_{\text{ref}} \| \pi\right]$ is bounded.

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

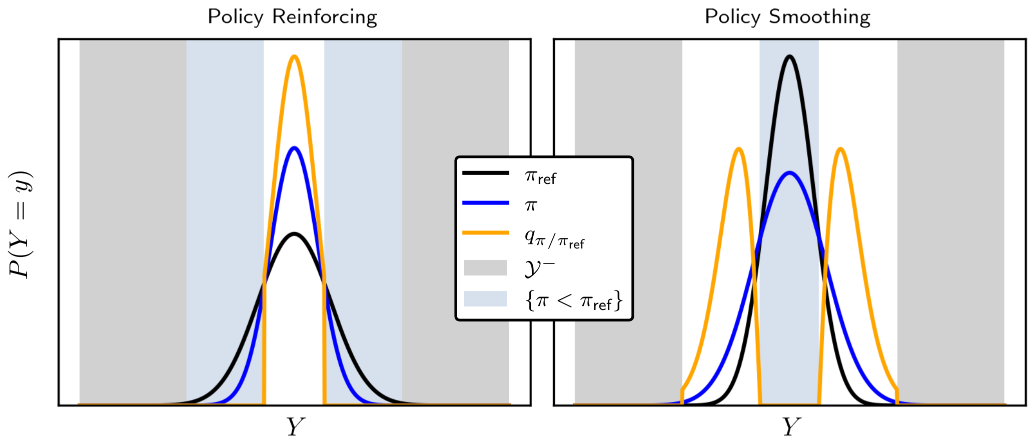

Figure 6: Illustration of policy reinforcement (*left*) and smoothing (*right*). The gray region corresponds to $\mathcal{Y}^- = \{y' \in \mathcal{Y} \mid \pi_{\text{ref}}(y') \approx 0\}$, and the light-blue region $\{\pi < \pi_{\text{ref}}\}$ corresponds to $\{\tilde{y} \in \mathcal{Y} \mid \pi(\tilde{y}) < \pi_{\text{ref}}(\tilde{y})\}$. *This plot serves only as an illustrative example and does not represent the true DID $q_{\pi^*/\pi_{\text{ref}}}$.*

Case 1: 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 far 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 6.

Case 2: 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:

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

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

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

See the right plot in Figure 6 for a visual intuition. When π smooths its distribution with respect to π_{ref}, we can expect $\mathbb{D}_{\text{KL}}\left[\pi_{\text{ref}} \| q_{\pi/\pi_{\text{ref}}}\right] > \mathbb{D}_{\text{KL}}\left[\pi_{\text{ref}} \| \pi\right]$. This results in the relationship: $H(q_{\pi/\pi_{\text{ref}}}) > H(\pi_{\text{ref}})$.

I DID ENTROPY AND PERFORMANCE TRENDS OF SIMPO

Table 3: Performance trends of `Qwen3-4B` trained on Magpie-Pro with the addition of SimPO. The DID entropy ($H(q_{\pi^*/\pi_{\text{ref}}})$, [nats]) is estimated by importance sampling (Appendix D.5). We report the win-rate [%] for Arena-Hard-v0.1 and ELO score for Wild-Bench-v2. For knowledge-intensive QA, we report mean reciprocal rank across 8 QA benchmarks (QA). Overall, SimPO closely follows DPO's behavior in learning high-entropy DID and achieving strong performance on open-ended tasks, while exhibiting lower factual precision (QA) than DPO-PG.

Method	$H(q_{\pi^*/\pi_{\text{ref}}})$	Arena-Hard-v0.1 (↑)	Wild-Bench-v2 (↑)	QA (↑)
DPO ($\beta = 1.0$)	9158.4	30.7	1134.4	0.37
DPO ($\beta = 0.2$)	4765.9	28.1	1146.2	0.30
DPO ($\beta = 0.1$)	7663.3	27.5	1148.2	0.23
DPO ($\beta = 0.05$)	6801.2	**43.7**	1164.5	0.20
SimPO ($\beta = 10, \gamma = 5$)	1071.4	42.8	**1177.5**	0.50
DPO-PG	388.2	37.4	1148.5	**0.85**

This section provides additional analysis on whether the Differential Information Distribution (DID) entropy trends identified in Section 5 extend to alternative objectives such as SimPO (Meng et al., 2024). Before presenting the results, we briefly note that comparisons with other LLD-mitigation objectives (DPOP (Pal et al., 2024), Cal-DPO (Xiao et al., 2024a)) were attempted but was shown to be infeasible at our scale: for 4-7B models trained on 100K preference samples over 5 epochs, these objectives exhibited high sensitivity to hyper-parameters and failure to mitigate LLD (see Appendix K). This instability was one of the motivations behind DPO-PG (Appendix K), which stably avoids LLD via projected gradient descent without introducing new hyper-parameters.

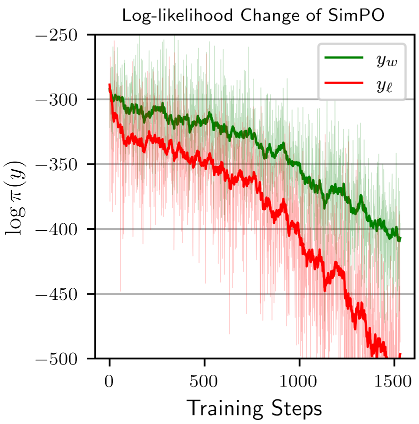

Figure 7: Likelihood change of SimPO when training `Qwen3-4B` on the Magpie-Pro dataset. SimPO also decreases the likelihood of chosen responses (LLD).

Experimental Setup. We trained `Qwen3-4B` using the SimPO objective on the Magpie-Pro dataset. During the experiments, SimPO also exhibited instability and was prone to over-optimization. Due to limited compute, we trained SimPO for 2 epochs and saved model checkpoints every 20,000 samples (10 checkpoints total) with a learning rate of 8e-7. Following the same model-selection protocol used for DPO and DPO-PG (Appendix M.2), we selected the best-performing checkpoint for evaluation.

Results. Figure 7 shows that SimPO also exhibits LLD, similar to the likelihood trends of DPO (Figure 14). Table 3 reports the DID entropy and downstream performance trends from Table 1 with the addition of SimPO. As with DPO, SimPO learns a high-entropy DID with strong performance on open-ended tasks (Arena-Hard-v0.1 and Wild-Bench-v2), at the cost of weaker factual accuracy (QA) compared to DPO-PG.[7]

These findings demonstrate that our DID entropy based analysis provides a comprehensive explanation of training dynamics and downstream behavior across diverse preference-learning methods.

[7] QA MRR values differ slightly from Table 1 because MRR is a relative metric which shifts when additional methods (SimPO) are included.

J Derivations and proofs

In this section we provide the detailed proofs supporting our theoretical findings.

J.1 Proof for equivalence of preference optimization

Theorem (Preference vs. Distribution Matching (Dumoulin et al., 2023))**.** *Let $\mathcal{D} = \{(y_w, y_\ell)\}$ be a sufficiently large preference dataset where the sets of y_w and y_ℓ cover $\mathcal{Y}$. Then preference optimization on $\mathcal{D}$ is equivalent to fitting the reward-induced distribution $P(Y = y \mid r)$ to the implicit preference distribution $p^*(y)$:*

$$\max_r \mathbb{E}_{(y_w,y_\ell)\sim\mathcal{D}} \left[\log \sigma(r(y_w) - r(y_\ell))\right] \iff \min_r \mathbb{D}_{\mathrm{KL}} \left[p^*(y) \| P(Y = y \mid r)\right].$$

We restate the proof in Dumoulin et al. (2023) for reference.

Proof. Recall from Section 2 that we model the ground truth probability of y_1 being preferred over y_2 as

$$p^*(y_1 \succ y_2) = \frac{\pi_w(y_1)\pi_\ell(y_2)}{\pi_w(y_1)\pi_\ell(y_2) + \pi_w(y_2)\pi_\ell(y_1)}.$$

Now, for a sufficiently large preference dataset $\mathcal{D}$, we can show that preference optimization is equivalent to minimizing the KL-divergence between the preference distributions. First, observe the following relationship:

$$\mathbb{E}_{(y_w,y_\ell)\sim\mathcal{D}} \left[\log \sigma(r(y_w) - r(y_\ell))\right] = \sum_{(y_w,y_\ell)\in\mathcal{Y}\times\mathcal{Y}} \pi_w(y_w)\pi_\ell(y_\ell) \log \sigma(r(y_w) - r(y_\ell))$$

$$= \sum_{(y_w,y_\ell)\in\mathcal{I}} \Big(\pi_w(y_w)\pi_\ell(y_\ell) + \pi_w(y_\ell)\pi_\ell(y_w)\Big) \Big[\frac{\pi_w(y_w)\pi_\ell(y_\ell)}{\pi_w(y_w)\pi_\ell(y_\ell) + \pi_w(y_\ell)\pi_\ell(y_w)} \log \sigma(r(y_w) - r(y_\ell))$$

$$+ \frac{\pi_w(y_\ell)\pi_\ell(y_w)}{\pi_w(y_w)\pi_\ell(y_\ell) + \pi_w(y_\ell)\pi_\ell(y_w)} \log \sigma(r(y_\ell) - r(y_w))\Big]$$

$$= -\frac{1}{2}\mathbb{E}_{(y_w,y_\ell)\sim\mathcal{D}} \Big[- p^*(y_w \succ y_\ell) \log p(y_w \succ y_\ell \mid r) - p^*(y_\ell \succ y_w) \log p(y_\ell \succ y_w \mid r)\Big]$$

$$= -\frac{1}{2}\mathbb{E}_{(y_w,y_\ell)\sim\mathcal{D}} \Big[\mathbb{D}_{\mathrm{KL}} \left[p^*(y_w \succ y_\ell) \| p(y_w \succ y_\ell \mid r)\right]\Big] + C,$$

where $\mathcal{I} = \{(y_i, y_j) \in \mathcal{Y} \times \mathcal{Y} : i > j\}$ is the set of ordered distinct pairs (y_i, y_j), and C is a constant term independent of r. Therefore, preference optimization is equivalent to minimizing the KL-divergence between preference distributions:

$$\arg\max_r \mathbb{E}_{(y_w,y_\ell)\sim\mathcal{D}} \left[\log \sigma(r(y_w) - r(y_\ell))\right]$$
$$= \arg\min_r \mathbb{E}_{(y_w,y_\ell)\sim\mathcal{D}} \Big[\mathbb{D}_{\mathrm{KL}} \left[p^*(y_w \succ y_\ell) \| p(y_w \succ y_\ell \mid r)\right]\Big].$$

Now, for any two reward parameterizations r_1 and r_2, $\mathbb{D}_{\mathrm{KL}} \left[p(y_w \succ y_\ell \mid r_1) \| p(y_w \succ y_\ell \mid r_2)\right]$ is minimized to 0 if and only if $r_1(y) = r_2(y) + C$ for all $y \in \mathcal{Y}$ and for some constant C. If we let $r_1(y) = \log p^*(y)$, we have $p(y_w \succ y_\ell \mid r_1) = p^*(y_w \succ y_\ell)$. Next, set $r(y) = r_2(y)$ and the following holds:

$$\begin{aligned}
\mathbb{E}_{(y_w,y_\ell)\sim\mathcal{D}} \left[\mathbb{D}_{\mathrm{KL}} \left[p(y_w \succ y_\ell \mid r_1) \| p(y_w \succ y_\ell \mid r_2)\right]\right] = 0 &\iff \\
\mathbb{E}_{(y_w,y_\ell)\sim\mathcal{D}} \left[\mathbb{D}_{\mathrm{KL}} \left[p^*(y_w \succ y_\ell) \| p(y_w \succ y_\ell \mid r)\right]\right] = 0 &\iff \\
\forall y \in \mathcal{Y} : \log p^*(y) = r(y) + C &\iff \\
\forall y \in \mathcal{Y} : p^*(y) \propto \exp(r(y)) &\iff \\
\forall y \in \mathcal{Y} : p^*(y) = P(Y = y \mid r) &\iff \\
\mathbb{D}_{\mathrm{KL}} \left[p^*(y) \| P(Y = y \mid r)\right] = 0.&
\end{aligned}$$

Therefore, for any reward parameterization $r : \mathcal{Y} \to \mathbb{R}$, the preference optimization objective is optimized only when the reward induced distribution $P(Y = y \mid r) := \frac{\exp(r(y))}{\sum_{y'\in\mathcal{Y}} \exp(r(y'))}$ is exactly the same as the ground truth preference distribution $p^*(y)$. □

J.2 Proof for preferences encoding Differential Information

Theorem (Preferences Encoding Differential Information)**.** *Consider a preference dataset $\mathcal{D} = \{(y_w, y_\ell) \mid y_w \sim \pi_w, y_\ell \sim \pi_\ell\}$. Let π^* be the target policy. If the Differential Information Distribution between policies match up to an exponent $\beta > 0$:*

$$q_{\pi_w/\pi_\ell}(y) \propto q_{\pi^*/\pi_{\mathrm{ref}}}(y)^\beta, \quad \forall y \in \mathcal{Y},$$

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

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

Proof. The relationship follows by directly applying the power-law DID relationship to the ground-truth preference probability.

$$\begin{aligned}
p^*(y_1 \succ y_2) &= \frac{\pi_w(y_1)\pi_\ell(y_2)}{\pi_w(y_1)\pi_\ell(y_2) + \pi_w(y_2)\pi_\ell(y_1)} \\
&= \frac{\frac{\pi_w(y_1)}{\pi_\ell(y_1)}}{\frac{\pi_w(y_1)}{\pi_\ell(y_1)} + \frac{\pi_w(y_2)}{\pi_\ell(y_2)}} \\
&= \sigma\left(\log \frac{\pi_w(y_1)}{\pi_\ell(y_1)} - \log \frac{\pi_w(y_2)}{\pi_\ell(y_2)}\right) \\
&= \sigma\left(\log q_{\pi_w/\pi_\ell}(y_1) - \log q_{\pi_w/\pi_\ell}(y_2)\right) \\
&= \sigma\left(\beta \log q_{\pi^*/\pi_{\mathrm{ref}}}(y_1) - \beta \log q_{\pi^*/\pi_{\mathrm{ref}}}(y_2)\right).
\end{aligned}$$

□

J.3 Proof for optimal reward for learning Differential Information

Theorem (Optimal Reward for Learning Differential Information)**.** *Let $\mathcal{D}$ be a preference dataset satisfying Theorem 3.1, encoding the Differential Information required to learn the target policy π^*. Then, for some constant C, we have*

$$\pi^* = \arg\max_\pi \mathbb{E}_{(y_w, y_\ell)\sim\mathcal{D}}\left[\log \sigma(r(y_w) - r(y_\ell))\right] \iff r(y) = \beta \log \frac{\pi(y)}{\pi_{\mathrm{ref}}(y)} + C.$$

for some constant C.

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

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

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

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

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

Therefore, in order to have the following equivalence:

$$\mathbb{E}_{(y_w, y_\ell)\sim\mathcal{D}}\left[p^*(y_w \succ y_\ell) \| p(y_w \succ y_\ell \mid r)\right] = 0 \iff \pi^* = \pi,$$

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

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

for some constant C. □

J.4 Proof for power-law structure of DPO

Corollary (DID Power-Law of DPO). *Consider a preference dataset $\mathcal{D}=\{(y_w,y_\ell)\mid y_w\sim\pi_w, y_\ell\sim\pi_\ell\}$ and a policy π^* obtained as a stationary point of preference optimization using the log-ratio reward $r=\beta\log(\pi/\pi_{\mathrm{ref}})$ on $\mathcal{D}$. Then, a power-law relationship between the DID of policies must hold:*

$$q_{\pi_w/\pi_\ell}(y)\propto q_{\pi^*/\pi_{\mathrm{ref}}}(y)^{\beta},\quad\forall y\in\mathcal{Y}.$$

Proof. According to Theorem 2.1, the converged policy π^* obtained by optimizing $\mathcal{D}$ with $r_{\mathrm{DPO}}=\beta\log\pi/\pi_{\mathrm{ref}}$ must follow $\pi^*(y)\propto\pi_{\mathrm{ref}}(y)\cdot(p^*(y))^{\frac{1}{\beta}}$ due to the following:

$$\begin{aligned}p^*(y)=P(Y=y\mid r_{\mathrm{DPO}})\propto q_{\pi^*/\pi_{\mathrm{ref}}}(y)^{\beta},\quad\forall y\in\mathcal{Y}\\ \iff (p^*(y))^{\frac{1}{\beta}}\propto q_{\pi^*/\pi_{\mathrm{ref}}}(y),\quad\forall y\in\mathcal{Y}\\ \iff \pi^*(y)\propto\pi_{\mathrm{ref}}(y)\cdot(p^*(y))^{\frac{1}{\beta}}.\quad\forall y\in\mathcal{Y}\end{aligned}$$

Meanwhile, it can also be shown that $p^*=q_{\pi_w/\pi_\ell}$. This because the reward $r'=\log\pi_w/\pi_\ell$ perfectly fits the ground-truth preference distribution. For all $y_1,y_2\in\mathcal{Y}\times\mathcal{Y}$,

$$\begin{aligned}p^*(y_1\succ y_2)&=\frac{\pi_w(y_1)\pi_\ell(y_2)}{\pi_w(y_1)\pi_\ell(y_2)+\pi_w(y_2)\pi_\ell(y_1)}\\ &=\sigma\left(\log\frac{\pi_w(y_1)}{\pi_\ell(y_1)}-\log\frac{\pi_w(y_2)}{\pi_\ell(y_2)}\right)\\ &=\sigma\left(r'(y_1)-r'(y_2)\right)\\ &\Rightarrow p^*(y)=P(Y=y\mid r')=q_{\pi_w/\pi_\ell}(y),\quad\forall y\in\mathcal{Y}\qquad\text{(Theorem 2.1)}.\end{aligned}$$

Since $\pi^*(y)\propto\pi_{\mathrm{ref}}(y)\cdot(p^*(y))^{\frac{1}{\beta}}$ and $p^*=q_{\pi_w/\pi_\ell}$, the power-law DID relationship $q_{\pi_w/\pi_\ell}(y)\propto q_{\pi^*/\pi_{\mathrm{ref}}}(y)^{\beta}$ follows directly.

Note that this result recovers the findings of Pan et al. (2025), where the authors derive the power-law DID relationship from the functional derivative of the DPO loss. In contrast, our proof takes an alternative approach by leveraging the distribution matching result of Theorem 2.1 (Dumoulin et al., 2023). □

J.5 Proof for log-likelihood changes in DPO

Theorem (Log-Likelihood Change of DPO). *Consider a preference dataset $\mathcal{D}=\{(y_w,y_\ell)\mid y_w\sim\pi_{\mathrm{ref}}, y_\ell\sim\pi_\ell\}$, and π^* obtained by preference optimization on $\mathcal{D}$ using the log-ratio reward $r=\beta\log\pi/\pi_{\mathrm{ref}}$. Then, for any $\beta>0$, π^* must decrease the average log-likelihood of y_ℓ:*

$$\mathbb{E}_{y_\ell\sim\pi_\ell}\left[\log\pi^*(y_\ell)\right]<\mathbb{E}_{y_\ell\sim\pi_\ell}\left[\log\pi_{\mathrm{ref}}(y_\ell)\right].$$

Conversely, if π_{ref} was fine-tuned on y_ℓ (i.e., $y_\ell\sim\pi_{\mathrm{ref}}$), then, for any $\beta\geq1$, π^* must increase the average log-likelihood of y_w:

$$\mathbb{E}_{y_w\sim\pi_w}\left[\log\pi^*(y_w)\right]>\mathbb{E}_{y_w\sim\pi_w}\left[\log\pi_{\mathrm{ref}}(y_w)\right].$$

Proof.

Case $\pi_{\text{ref}} = \pi_w$**:** Assume $\beta > 0$. Let $Z = \sum_{y\in\mathcal{Y}} \pi_{\text{ref}}(y) \cdot (\frac{\pi_{\text{ref}}(y)}{\pi_\ell(y)})^{\frac{1}{\beta}}$. It can be shown that $\log Z > 0$ due to the following:

$$\begin{aligned}
\log Z &= \log \sum_{y\in\mathcal{Y}} \pi_{\text{ref}}(y) \cdot (\frac{\pi_{\text{ref}}(y)}{\pi_\ell(y)})^{\frac{1}{\beta}} \\
&= \log \sum_{y\in\mathcal{Y}} \pi_\ell(y) \cdot (\frac{\pi_{\text{ref}}(y)}{\pi_\ell(y)})^{1+\frac{1}{\beta}} \\
&= \log \mathbb{E}_{y\sim\pi_\ell}\left[(\frac{\pi_{\text{ref}}(y)}{\pi_\ell(y)})^{1+\frac{1}{\beta}}\right] \\
&> \log \left(\mathbb{E}_{y\sim\pi_\ell}\left[\frac{\pi_{\text{ref}}(y)}{\pi_\ell(y)}\right]\right)^{1+\frac{1}{\beta}} \qquad \text{(Jensen's Inequality)} \\
&= (1+\frac{1}{\beta}) \log 1 = 0.
\end{aligned}$$

Since $\pi^*(y) \propto \pi_{\text{ref}}(y) \cdot p^*(y)^{\frac{1}{\beta}}$ and $p^* = q_{\pi_w/\pi_\ell} = q_{\pi_{\text{ref}}/\pi_\ell}$, it follows that $\pi^*(y) \propto \pi_{\text{ref}}(y) \cdot (q_{\pi_{\text{ref}}/\pi_\ell}(y))^{\frac{1}{\beta}}$. Therefore, we have

$$\begin{aligned}
&\mathbb{E}_{y_\ell\sim\pi_\ell}[\log \pi^*(y_\ell) - \log \pi_{\text{ref}}(y_\ell)] \\
&= \frac{1}{\beta} \sum_{y\in\mathcal{Y}} \pi_\ell(y) \log \frac{\pi_{\text{ref}}(y)}{\pi_\ell(y) Z} \\
&= -\frac{1}{\beta}\mathbb{D}_{\text{KL}}\left[\pi_\ell \| \pi_{\text{ref}}\right] - \log Z < 0 \quad \because \mathbb{D}_{\text{KL}}\left[\pi_\ell \| \pi_{\text{ref}}\right] > 0 \text{ and } \log Z > 0.
\end{aligned}$$

Case $\pi_{\text{ref}} = \pi_\ell$**:** Assume $\beta \geq 1$ and $\pi_{\text{ref}} \neq \pi_w$. Let $Z = \sum_{y\in\mathcal{Y}} \pi_{\text{ref}}(y) \cdot (\frac{\pi_w(y)}{\pi_{\text{ref}}(y)})^{\frac{1}{\beta}}$. It can be shown that $\log Z < 0$ due to the following:

$$\begin{aligned}
\log Z &= \log \sum_{y\in\mathcal{Y}} \pi_{\text{ref}}(y) \cdot (\frac{\pi_w(y)}{\pi_{\text{ref}}(y)})^{\frac{1}{\beta}} \\
&= \log \sum_{y\in\mathcal{Y}} \pi_w(y)^{\frac{1}{\beta}} \cdot (\pi_{\text{ref}}(y))^{1-\frac{1}{\beta}} \\
&< \log \left(\left(\sum_{y\in\mathcal{Y}} \pi_w(y)\right)^{\frac{1}{\beta}} \cdot \left(\sum_{y\in\mathcal{Y}} \pi_{\text{ref}}(y)\right)^{1-\frac{1}{\beta}}\right) \qquad \text{(Hölder's Inequality)} \\
&= \log(1 \cdot 1) = 0.
\end{aligned}$$

Since $\pi^*(y) \propto \pi_{\text{ref}}(y) \cdot p^*(y)^{\frac{1}{\beta}}$ and $p^* = q_{\pi_w/\pi_\ell} = q_{\pi_w/\pi_{\text{ref}}}$, it follows that $\pi^*(y) \propto \pi_{\text{ref}}(y) \cdot (q_{\pi_w/\pi_{\text{ref}}}(y))^{\frac{1}{\beta}}$. Therefore, we have

$$\begin{aligned}
&\mathbb{E}_{y_w\sim\pi_w}[\log \pi^*(y_\ell) - \log \pi_{\text{ref}}(y_\ell)] \\
&= \frac{1}{\beta} \sum_{y\in\mathcal{Y}} \pi_w(y) \log \frac{\pi_w(y)}{\pi_{\text{ref}}(y) Z} \\
&= \frac{1}{\beta}\mathbb{D}_{\text{KL}}\left[\pi_w \| \pi_{\text{ref}}\right] - \log Z > 0 \quad \because \mathbb{D}_{\text{KL}}\left[\pi_w \| \pi_{\text{ref}}\right] > 0 \text{ and } \log Z < 0.
\end{aligned}$$

□

J.6 Proof for preference data strength

Theorem (Adaptive Policy Exploration of DPO)**.** *Let $\mathcal{D} = \{(y_w, y_\ell) \mid y_w \sim \pi_{\mathrm{ref}},\ y_\ell \sim \pi_\ell\}$ be a preference dataset with an implicit Bradley-Terry preference distribution $p^*_{\mathcal{D}}$. Consider another dataset $\mathcal{D}' = \{(y_w, y_\ell)\}$ whose implicit Bradley-Terry distribution $p^*_{\mathcal{D}'}$ is a "sharpened" version of $p^*_{\mathcal{D}}$, in the sense that there exists $\alpha > 1$ such that for all pairs $(y_w, y_\ell) \in \mathcal{Y} \times \mathcal{Y}$,*

$$p^*_{\mathcal{D}'}(y_w \succ y_\ell) = \frac{\big(p^*_{\mathcal{D}}(y_w)\big)^\alpha}{\big(p^*_{\mathcal{D}}(y_w)\big)^\alpha + \big(p^*_{\mathcal{D}}(y_\ell)\big)^\alpha} = \exp\big(\alpha \log p^*_{\mathcal{D}}(y_w) - \alpha \log p^*_{\mathcal{D}}(y_\ell)\big).$$

For the same reference policy π_{ref} and any $\beta > 0$, let $\pi^*_{\mathcal{D}}$ and $\pi^*_{\mathcal{D}'}$ denote the policies obtained by preference optimization on $\mathcal{D}$ and $\mathcal{D}'$, respectively, using the log-ratio reward $r = \beta \log \pi/\pi_{\mathrm{ref}}$. Then the strengthened dataset $\mathcal{D}'$ induces a strictly larger divergence from the reference:

$$\mathbb{D}_{\mathrm{KL}}\left[\pi_{\mathrm{ref}} \| \pi^*_{\mathcal{D}'}\right] \ > \ \mathbb{D}_{\mathrm{KL}}\left[\pi_{\mathrm{ref}} \| \pi^*_{\mathcal{D}}\right].$$

Proof. Let us denote $Z_{\mathcal{D}} = \sum_y \pi_{\mathrm{ref}}(y)(\frac{\pi_{\mathrm{ref}}(y)}{\pi_\ell(y)})^{\frac{1}{\beta}}$ and $Z_{\mathcal{D}'} = \sum_y \pi_{\mathrm{ref}}(y)(\frac{\pi_{\mathrm{ref}}(y)}{\pi_\ell(y)})^{\frac{\alpha}{\beta}}$. Observe the following:

$$\pi^*_{\mathcal{D}}(y) = \frac{\pi_{\mathrm{ref}}(y) \cdot (\frac{\pi_{\mathrm{ref}}(y)}{\pi_\ell(y)})^{\frac{1}{\beta}}}{Z_{\mathcal{D}}}, \quad \pi^*_{\mathcal{D}}(y) = \frac{\pi_{\mathrm{ref}}(y) \cdot (\frac{\pi_{\mathrm{ref}}(y)}{\pi_\ell(y)})^{\frac{\alpha}{\beta}}}{Z_{\mathcal{D}'}}.$$

Therefore, we can express the difference in the KL-divergence as

$$\begin{aligned}
\mathbb{D}_{\mathrm{KL}}\left[\pi_{\mathrm{ref}} \| \pi^*_{\mathcal{D}'}\right] - \mathbb{D}_{\mathrm{KL}}\left[\pi_{\mathrm{ref}} \| \pi^*_{\mathcal{D}}\right] &= \sum_{y \in \mathcal{Y}} \pi_{\mathrm{ref}}(y) \log \frac{\pi_{\mathrm{ref}}(y)}{\pi^*_{\mathcal{D}'}(y)} - \sum_{y \in \mathcal{Y}} \pi_{\mathrm{ref}}(y) \log \frac{\pi_{\mathrm{ref}}(y)}{\pi^*_{\mathcal{D}}(y)} \\
&= \sum_{y \in \mathcal{Y}} \pi_{\mathrm{ref}}(y) \log \frac{\pi^*_{\mathcal{D}}(y)}{\pi^*_{\mathcal{D}'}(y)} \\
&= \log \frac{Z_{\mathcal{D}'}}{Z_{\mathcal{D}'}} + \frac{1-\alpha}{\beta} \mathbb{D}_{\mathrm{KL}}\left[\pi_{\mathrm{ref}} \| \pi_\ell\right].
\end{aligned}$$

Now, let $r(y) = \frac{\pi_{\mathrm{ref}}(y)}{\pi_\ell(y)}$ and $X = \log r(y)$. Also, define the cumulant-generating function $K(t)$:

$$K(t) = \log \mathbb{E}_{\pi_{\mathrm{ref}}}[e^{tX}] = \log \sum_{y \in \mathcal{Y}} \pi_{\mathrm{ref}}(y) r(y)^t.$$

Then, we have the following:

$$\log \frac{Z_{\mathcal{D}'}}{Z_{\mathcal{D}}} = K(\frac{\alpha}{\beta}) - K(\frac{1}{\beta}), \quad \mathbb{D}_{\mathrm{KL}}\left[\pi_{\mathrm{ref}} \| \pi_\ell\right] = \mathbb{E}_{\pi_{\mathrm{ref}}}[X] = K'(0),$$

where $K'(t) = \frac{d}{dt} K(t)$.

Therefore, we obtain the following expression:

$$\mathbb{D}_{\mathrm{KL}}\left[\pi_{\mathrm{ref}} \| \pi^*_{\mathcal{D}'}\right] - \mathbb{D}_{\mathrm{KL}}\left[\pi_{\mathrm{ref}} \| \pi^*_{\mathcal{D}}\right] = K(\frac{\alpha}{\beta}) - K(\frac{1}{\beta}) + \frac{1-\alpha}{\beta} K'(0).$$

Since the cumulant-generating function $K(t)$ is convex and twice differentiable, we have

$$\begin{aligned}
& K(\frac{\alpha}{\beta}) \geq K(\frac{1}{\beta}) + \left(\frac{\alpha}{\beta} - \frac{1}{\beta}\right) K'(\frac{1}{\beta}) \\
\iff \ & K(\frac{\alpha}{\beta}) - K(\frac{1}{\beta}) \geq \left(\frac{\alpha}{\beta} - \frac{1}{\beta}\right) K'(\frac{1}{\beta}).
\end{aligned}$$

Meanwhile, since $K'(t)$ is non-decreasing due to convexity, we have $K'(\frac{1}{\beta}) > K'(0)$. Therefore, we arrive at the final relationship:

$$\mathbb{D}_{\mathrm{KL}}\left[\pi_{\mathrm{ref}} \| \pi^*_{\mathcal{D}'}\right] - \mathbb{D}_{\mathrm{KL}}\left[\pi_{\mathrm{ref}} \| \pi^*_{\mathcal{D}}\right] \geq \frac{\alpha - 1}{\beta} \left(K'(\frac{1}{\beta}) - K'(0) \right) > 0,$$

where the strict inequality comes from $\pi_{\mathrm{ref}} \neq \pi_\ell$. □

K DPO-Projected Gradient (DPO-PG)

While several variants of DPO have been proposed to address log-likelihood displacement (Pal et al., 2024; Xiao et al., 2024a), we observed that these methods exhibit instability when scaled to large datasets (approximately 100,000 samples) and trained over multiple epochs (*e.g.*, 5 epochs in our experiments of Section 5.2). A proper alternative that prevents LLD should 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.

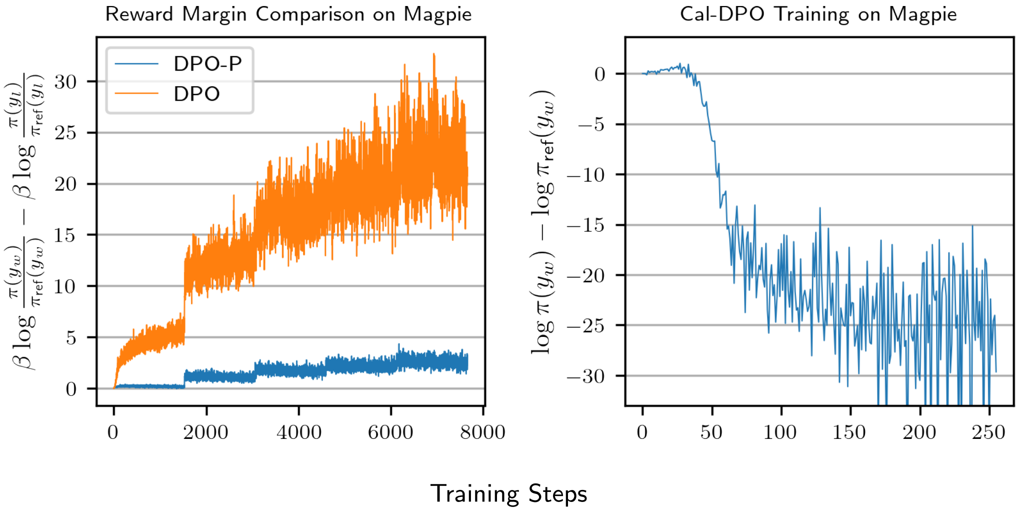

Figure 8: Testing DPOP (Pal et al., 2024) and Cal-DPO (Xiao et al., 2024a) on Magpie dataset. We found that DPOP fails to optimize the log-margin as effectively as vanilla DPO. Meanwhile, we found that Cal-DPO is unstable at preventing log-likelihood displacement.

Despite extensive experiments with various hyper-parameters, we failed to find a setting for both DPOP (Pal et al., 2024) and Cal-DPO (Xiao et al., 2024a) which met this criterion reliably (Figure 8). This motivated us to design a new method that reliably prevents log-likelihood displacement while ensuring 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 (Boyd & Vandenberghe, 2014) to reinforce the policy distribution while optimizing the DPO objective. Specifically, it increases $\log \pi(y_w)$ while maintaining or decreasing $\log \pi(y_\ell)$. 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 K.5.1).

The primary advantage of using DPO-PG over other DPO variants (*e.g.*, DPOP (Pal et al., 2024), Cal-DPO (Xiao et al., 2024a)) is that DPO-PG can reliably optimize the DPO loss while increasing both $\log \pi(y_w)$ and the log-margin $\log \pi(y_w) - \log \pi(y_\ell)$, all without introducing any additional hyper-parameters. We empirically confirm that DPO-PG prevents LLD from Figure 12, and that it also optimizes the DPO loss to a comparable extent in Figure 13.

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

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 (Kingma & Ba, 2015), RMSprop (Tieleman & Hinton, 2012)), we set the parameters' gradient as $\nabla L(y_w) - \frac{\alpha}{||\nabla L(y_\ell)||_2^2} \nabla L(y_\ell)$ and update its parameters following the optimizer's algorithm. For gradient-clipping, we clip the L2 norm of $\nabla L(y_w) - \frac{\alpha}{||\nabla L(y_\ell)||_2^2} \nabla L(y_\ell)$.

We now show that DPO-PG decreases $L(y_w)$ while maintaining or increasing $L(y_\ell)$, for sufficiently small step sizes. We begin with the definition of *descent direction* (Boyd & Vandenberghe, 2014):

Definition K.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 a direction is a descent direction of some differentiable objective function f (Boyd & Vandenberghe, 2014).

Lemma K.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: $\Delta\theta = \theta_{k+1} - \theta_k = -\eta\{\nabla L(y_w) - \frac{\max(0, \nabla L(y_w) \cdot \nabla L(y_\ell))}{||\nabla L(y_\ell)||_2^2} \nabla L(y_\ell)\}$. The following theorem states that DPO-PG increases the log-likelihood of y_w.

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

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

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

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

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

Case 2: Otherwise, we have $\nabla L(y_w) \cdot \nabla L(y_\ell) \leq 0$ 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 K.5. *$\Delta\theta$ is **not** a descent direction of the negative log-likelihood of the rejected responses $-\frac{1}{M}\Sigma_{i=1}^M \log \pi(y_\ell^{(i)}) = L(y_\ell)$.*

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

Meanwhile, various offline preference optimization methods can be characterized as solving the following objective (Tang et al., 2024):

$$\arg\min_\theta \mathbb{E}_{(y_w, y_\ell) \in \mathcal{D}} \left[f(\beta \log \frac{\pi_\theta(y_w)}{\pi_{\text{ref}}(y_w)} - \beta \log \frac{\pi_\theta(y_\ell)}{\pi_{\text{ref}}(y_\ell)}) \right],$$

where f denotes any valid supervised binary classification loss function (Hastie, 2009). As a consequence of Theorems K.4 and K.5, DPO-PG is able to optimize a wide variety of preference optimization objectives including DPO (Tang et al., 2024).

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

Proof.

$$\begin{aligned}
&\Delta\theta \cdot \nabla f(\beta \log \frac{\pi_\theta(y_w)}{\pi_{\text{ref}}(y_w)} - \beta \log \frac{\pi_\theta(y_\ell)}{\pi_{\text{ref}}(y_\ell)}) \\
&= \Delta\theta \cdot \beta f' \left(\beta \log \frac{\pi_\theta(y_w)}{\pi_{\text{ref}}(y_w)} - \beta \log \frac{\pi_\theta(y_\ell)}{\pi_{\text{ref}}(y_\ell)}\right) (\nabla L(y_w) - \nabla L(y_\ell)) \\
&= \underbrace{\beta\, f' \left(\beta \log \frac{\pi_\theta(y_w)}{\pi_{\text{ref}}(y_w)} - \beta \log \frac{\pi_\theta(y_\ell)}{\pi_{\text{ref}}(y_\ell)}\right)}_{f'(\cdot)<0} (\underbrace{\Delta\theta \cdot \nabla L(y_w)}_{>0} - \underbrace{\Delta\theta \cdot \nabla L(y_\ell)}_{\leq 0}).
\end{aligned}$$

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

To summarize, Lemma K.4 ensures that only $\log \pi(y_w)$ (and not $\log \pi(y_\ell)$) increases during training, for sufficiently small step sizes. This ensures policy reinforcement with respect to π_{ref}. Corollary K.5.1 further ensures that DPO-PG optimizes the DPO loss, too. We empirically validate that DPO prevents LLD in Figure 12, and also confirm that DPO-PG successfully optimizes the DPO loss in Figure 13.

L Additional experimental results

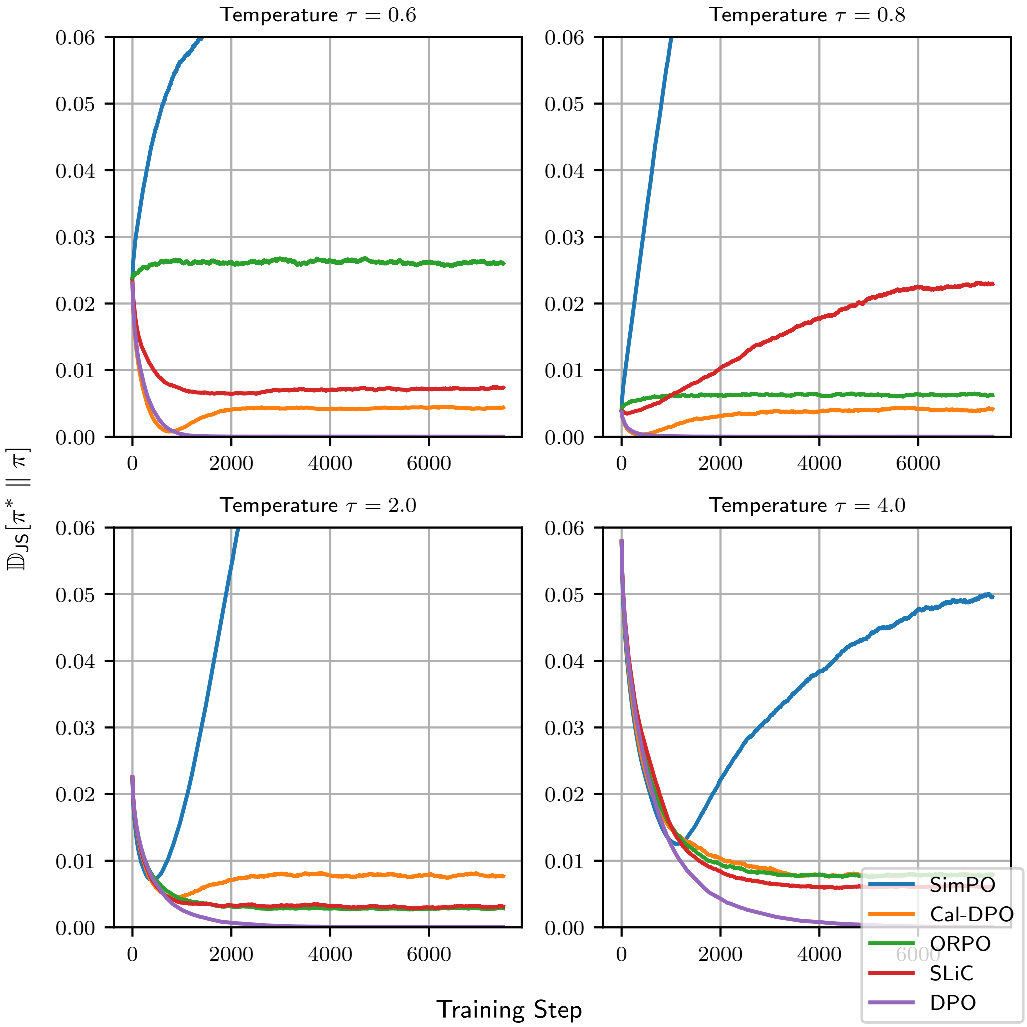

Figure 9: Validation of Theorem 3.2: Comparison of the Jensen-Shannon Divergence $\mathbb{D}_{\text{JS}}[\pi^* \| \pi]$ during training using different objectives on the synthetic dataset of Section 3.3. Standard DPO ($r = \log(\pi/\pi_{\text{ref}})$, purple) consistently minimizes the divergence to the target policy π^*. This demonstrates its optimality when preferences encode the Differential Information required to update the reference policy π_{ref} into the target policy π^*. The non-convergence of SimPO (Meng et al., 2024) to π^* also follows from Theorem 3.2, which states that the DPO log-ratio reward *uniquely* recovers π^* when preferences encode Differential Information (Theorem 3.1). We hypothesize that SimPO's trajectory exhibits a close "fly-by" of π^* before settling in a different region, though a principled explanation for this specific trajectory remains open.

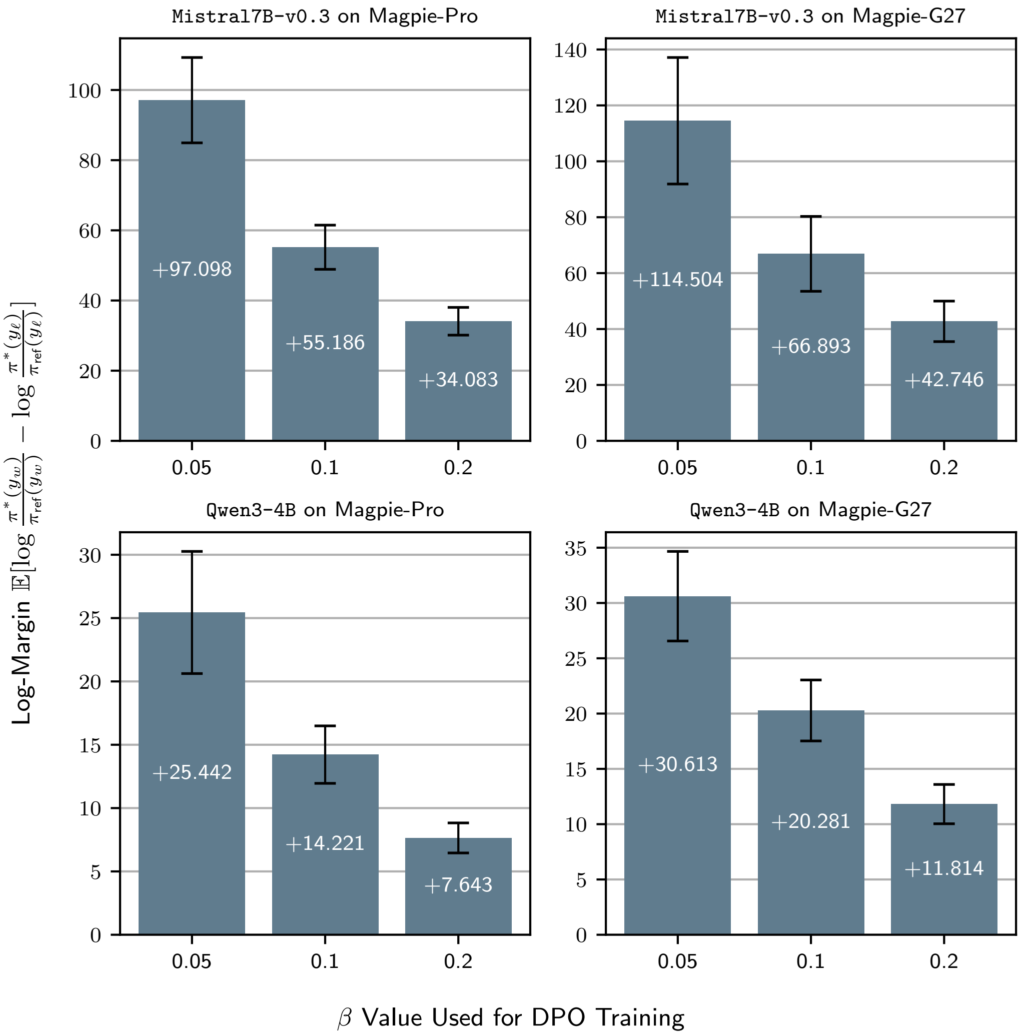

Figure 10: Log-margin [nats] of DPO-converged policies trained on real data, with error bars denoting ±1 standard deviation. The resulting values vary substantially with β, exhibiting statistically significant differences. In particular, the log-margin approximately scales proportionally to $1/\beta$, consistent with the DID power-law (Corollary 3.2.1).

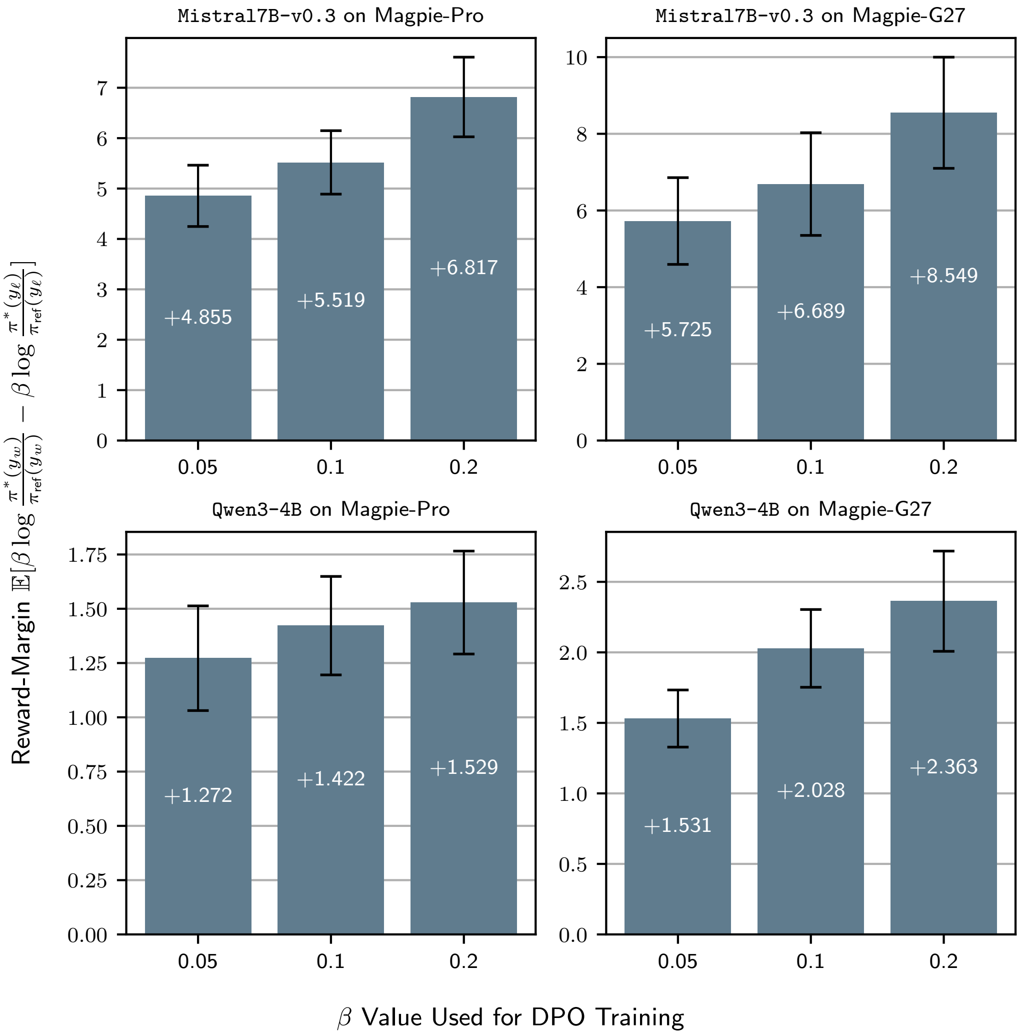

Figure 11: Reward-margin [nats] of DPO-converged policies trained on real data, with error bars denoting ±1 standard deviation. In contrast to the log-margin patterns from Figure 10, these values remain within a stable range across all $\beta \in \{0.05, 0.1, 0.2\}$. Although not perfectly identical due to finite data and imperfect optimization, the reward-margins exhibit strikingly low variation compared to the fluctuations observed in Figure 10, aligning with Corollary 3.2.1. Taken together with the log-margin trends, this β-invariant reward-margin behavior directly supports the DID power-law in real DPO training.

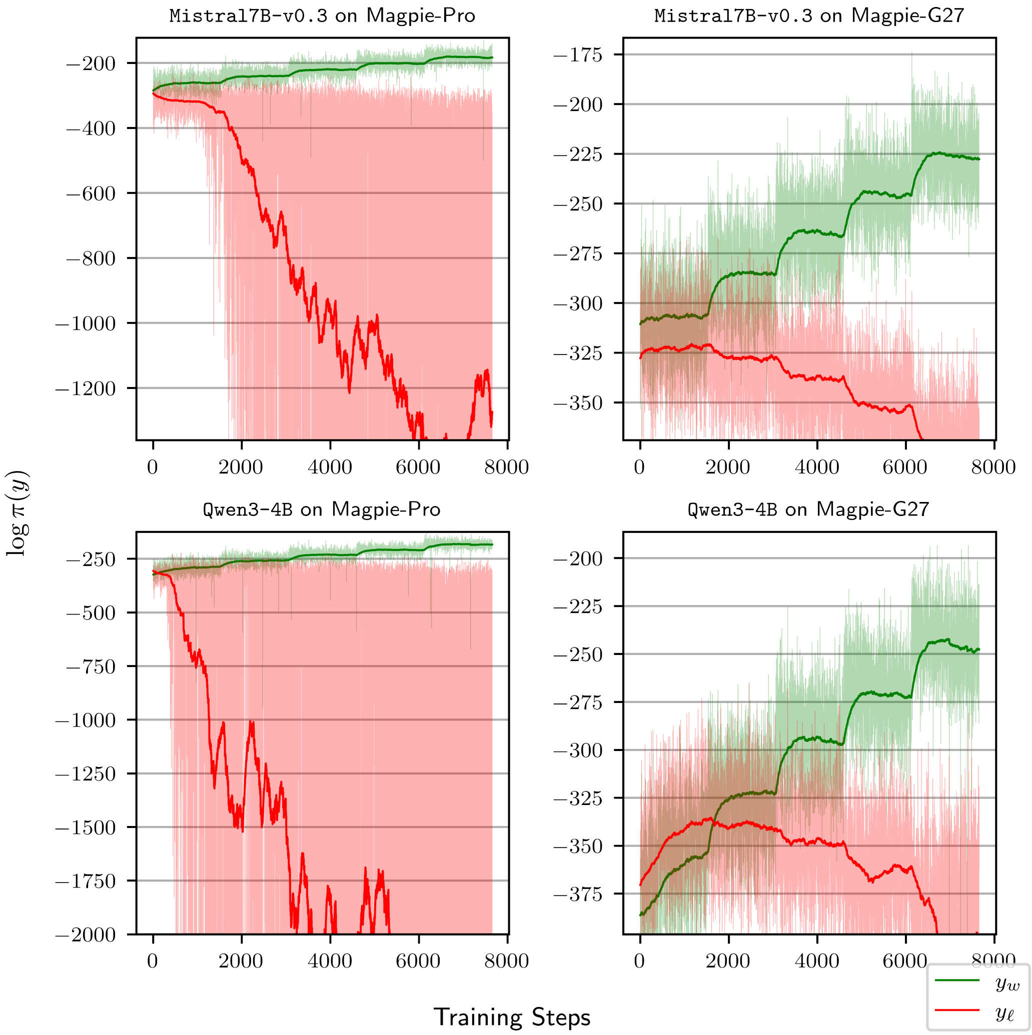

Figure 12: Log-likelihood change for DPO-PG across all experimental configurations. Overall, DPO-PG consistently increases the log-likelihood of y_w, while decreasing or maintaining the log-likelihood of y_ℓ.

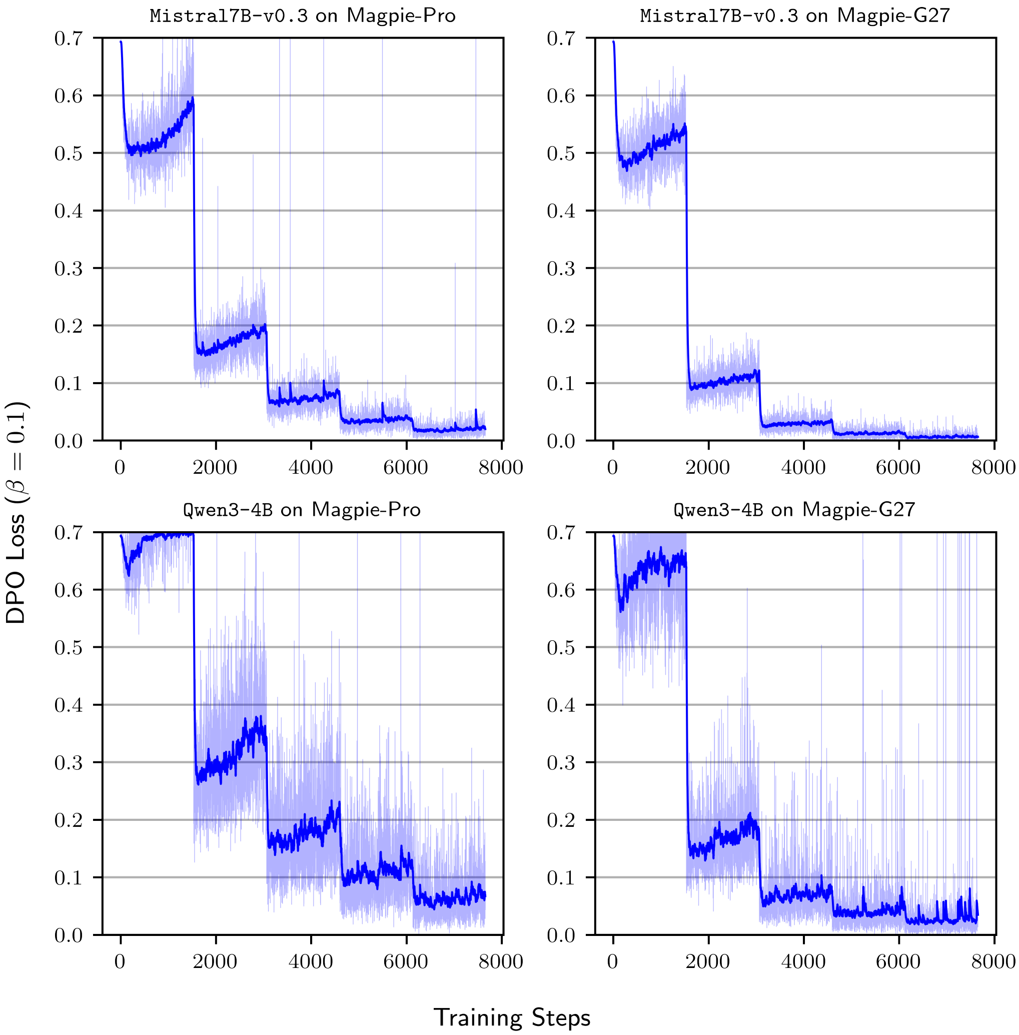

Figure 13: DPO loss for DPO-PG across all experimental configurations. The DPO loss is computed using $\beta = 0.1$. DPO-PG is able to optimize the DPO loss regardless of the model architecture or dataset, validating Corollary K.5.1. In conjunction with Figure 12, DPO-PG is able to prevent log-likelihood displacement while still optimizing the DPO objective.

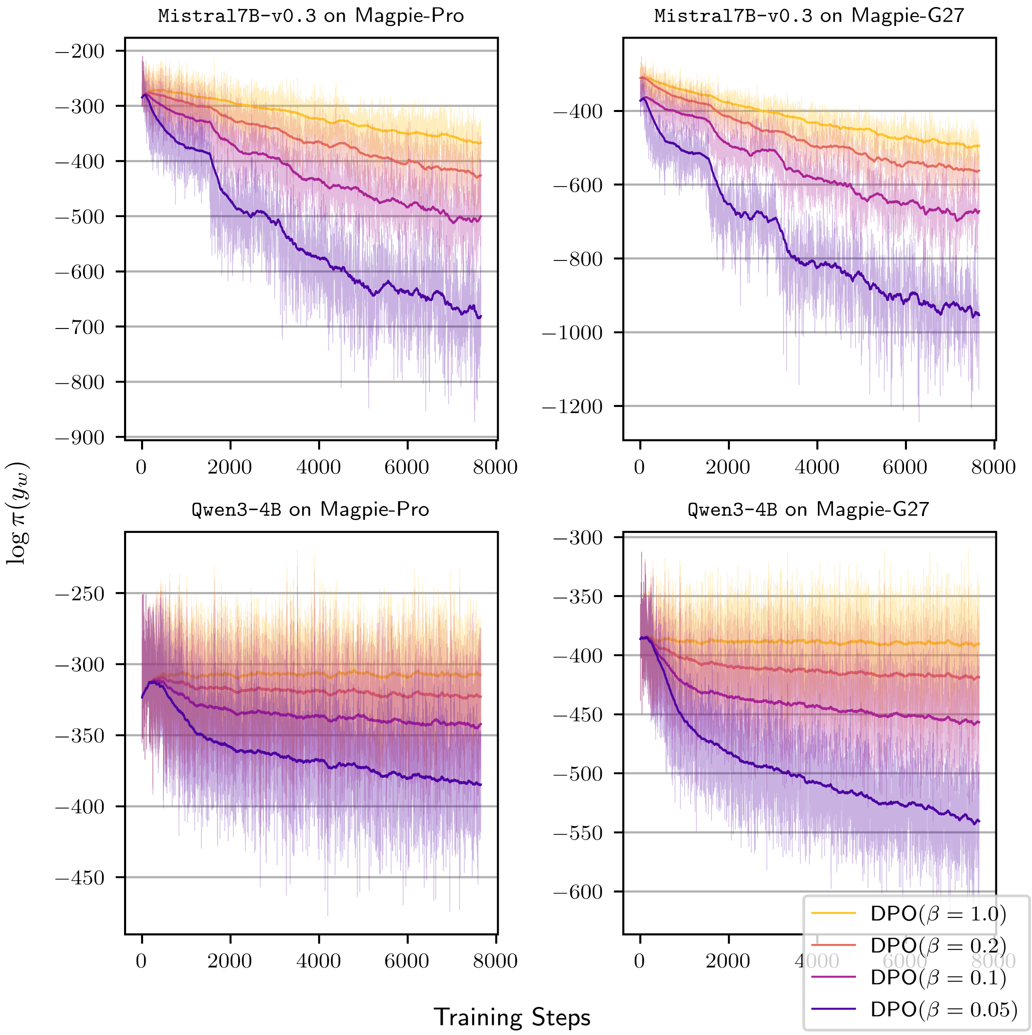

Figure 14: Log-likelihood change of y_w for DPO across all experimental configurations. The log-likelihood of chosen responses decreases throughout the training process, indicating log-likelihood displacement.

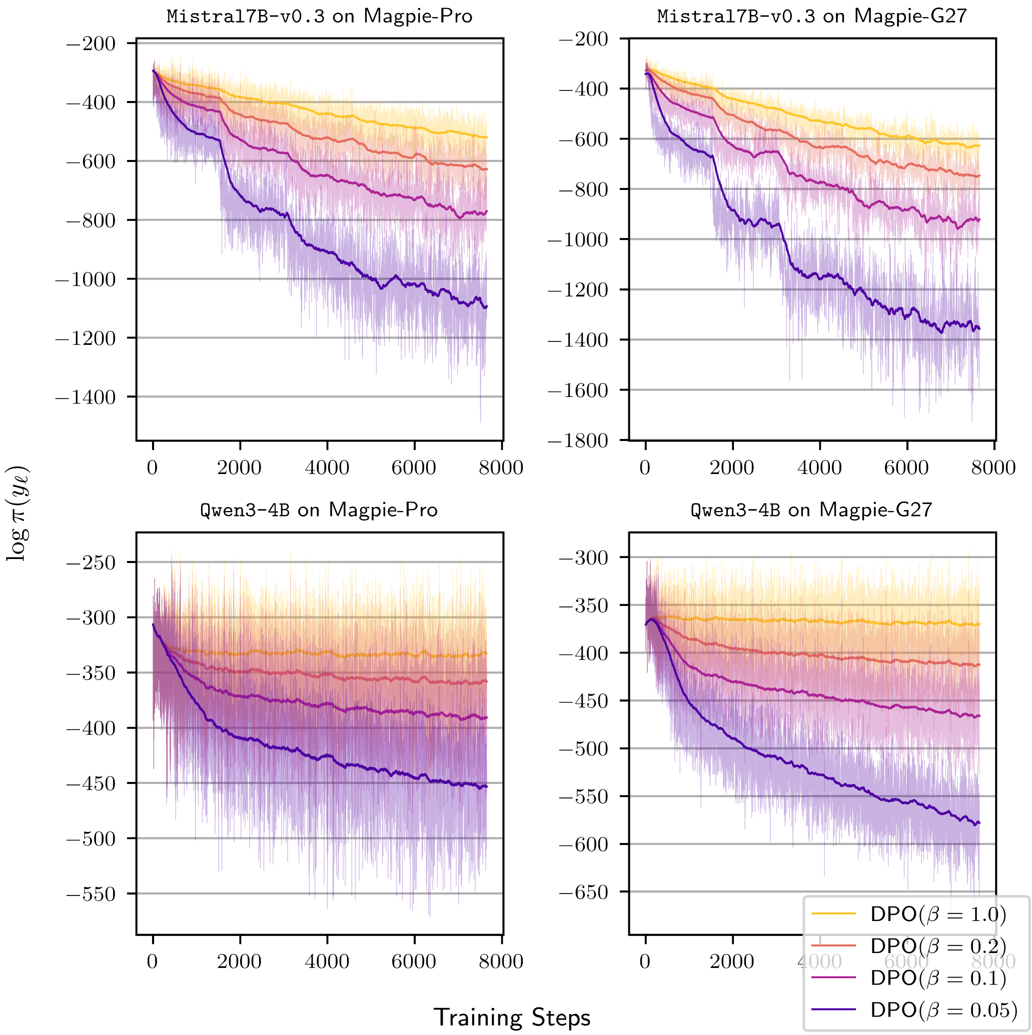

Figure 15: Log-likelihood change of y_ℓ for DPO across all experimental configurations.

Table 4: Evaluation results for open-ended instruction-following. We report the win-rate [%] for Arena-Hard-v0.1 and ELO score for Wild-Bench-v2, with the 95% confidence interval. We also specify the selected best epoch following the procedure in Appendix M.2, and highlight the model with the best Arena-Hard win-rate in **bold**. The DID entropy ($H(q_{\pi^*/\pi_{\mathrm{ref}}})$, [nats]) is estimated by importance sampling (Appendix D.5). Standard DPO, which exhibits LLD and learns a high-entropy DID, outperforms DPO-PG, which prevents LLD and learns a lower-entropy DID. This suggests that knowledge required for such open-ended tasks is associated with high-entropy DID.

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

Method	Best Epoch	$H(q_{\pi^*/\pi_{\mathrm{ref}}})$	Arena-Hard-v0.1	Wild-Bench-v2
DPO $\beta = 1.0$	3	1123.23	19.1 (-1.4, 2.0)	1141.47 (-10.17, 10.36)
DPO $\beta = 0.2$	4	1303.44	18.5 (-1.4, 1.6)	1145.34 (-9.79, 11.10)
DPO $\beta = 0.1$	1	**1253.89**	**23.4 (-1.9, 2.0)**	**1146.63 (-11.99, 9.52)**
DPO $\beta = 0.05$	1	970.21	22.4 (-1.7, 1.9)	1145.32 (-14.83, 12.52)
DPO-PG	5	495.12	19.6 (-1.9, 1.6)	1129.92 (-12.92, 11.83)

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

Method	Best Epoch	$H(q_{\pi^*/\pi_{\mathrm{ref}}})$	Arena-Hard-v0.1	Wild-Bench-v2
DPO $\beta = 1.0$	4	**836.56**	**30.4 (-2.2, 2.0)**	**1140.56 (-12.72, 12.72)**
DPO $\beta = 0.2$	2	576.85	30.0 (-2.6, 1.9)	1145.25 (-13.79, 13.36)
DPO $\beta = 0.1$	2	509.92	27.7 (-2.0, 2.4)	1146.87 (-13.20, 10.76)
DPO $\beta = 0.05$	1	694.96	27.0 (-2.3, 2.0)	1149.51 (-14.38, 14.72)
DPO-PG	5	378.07	24.0 (-1.9, 1.4)	1130.62 (-15.76, 15.07)

(c) `Qwen3-4B` trained on Magpie-Pro

Method	Best Epoch	$H(q_{\pi^*/\pi_{\mathrm{ref}}})$	Arena-Hard-v0.1	Wild-Bench-v2
DPO $\beta = 1.0$	2	9158.43	30.7 (-1.4, 2.0)	1134.39 (-15.73, 12.99)
DPO $\beta = 0.2$	3	4765.94	28.1 (-2.2, 2.1)	1146.17 (-11.92, 11.28)
DPO $\beta = 0.1$	4	7663.28	27.5 (-1.9, 1.9)	1148.22 (-8.99, 9.43)
DPO $\beta = 0.05$	4	**6801.18**	**43.7 (-2.9, 2.2)**	**1164.49 (-9.68, 13.13)**
DPO-PG	5	388.24	37.4 (-2.2, 2.4)	1148.50 (-15.64, 11.74)

(d) `Qwen3-4B` trained on Magpie-G27

Method	Best Epoch	$H(q_{\pi^*/\pi_{\mathrm{ref}}})$	Arena-Hard-v0.1	Wild-Bench-v2
DPO $\beta = 1.0$	2	6606.27	43.1 (-2.9, 2.5)	1157.95 (-14.05, 16.92)
DPO $\beta = 0.2$	5	14744.58	48.8 (-2.5, 2.8)	1165.78 (-11.16, 11.72)
DPO $\beta = 0.1$	3	3048.57	53.1 (-2.5, 2.3)	1173.52 (-15.24, 13.22)
DPO $\beta = 0.05$	4	**11705.65**	**54.0 (-2.7, 2.4)**	**1177.44 (-12.57, 13.84)**
DPO-PG	4	400.75	40.0 (-2.8, 2.1)	1160.97 (-11.34, 12.47)

Table 5: 95% confidence interval (CI) for the estimated DID entropy ($H(q_{\pi^*/\pi_{\text{ref}}})$, [nats]) values from Table 4. We compute the 95% bootstrap confidence intervals with 10,000 resamples. Overall, the DID entropy estimates are stable and the differences across methods are statistically significant.

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

Method	$H(q_{\pi^*/\pi_{\text{ref}}})$	95% Confidence Interval
DPO $\beta = 1.0$	1123.23	(1065.74, 1186.46)
DPO $\beta = 0.2$	1303.44	(1227.55, 1381.41)
DPO $\beta = 0.1$	1253.89	(1172.91, 1337.54)
DPO $\beta = 0.05$	970.21	(910.44, 1030.44)
DPO-PG	495.12	(477.12, 513.06)

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

Method	$H(q_{\pi^*/\pi_{\text{ref}}})$	95% Confidence Interval
DPO $\beta = 1.0$	836.56	(818.00, 855.87)
DPO $\beta = 0.2$	576.85	(563.49, 590.49)
DPO $\beta = 0.1$	509.92	(495.02, 525.25)
DPO $\beta = 0.05$	694.96	(664.22, 732.05)
DPO-PG	378.07	(367.27, 389.20)

(c) `Qwen3-4B` trained on Magpie-Pro

Method	$H(q_{\pi^*/\pi_{\text{ref}}})$	95% Confidence Interval
DPO $\beta = 1.0$	9158.43	(8753.45, 9577.95)
DPO $\beta = 0.2$	4765.94	(4538.30, 5004.31)
DPO $\beta = 0.1$	7663.28	(7515.45, 7810.67)
DPO $\beta = 0.05$	6801.18	(6684.91, 6921.41)
DPO-PG	388.24	(374.38, 402.53)

(d) `Qwen3-4B` trained on Magpie-G27

Method	$H(q_{\pi^*/\pi_{\text{ref}}})$	95% Confidence Interval
DPO $\beta = 1.0$	6606.27	(6239.81, 6983.42)
DPO $\beta = 0.2$	14744.58	(14546.10, 14948.17)
DPO $\beta = 0.1$	3048.57	(2862.17, 3239.40)
DPO $\beta = 0.05$	11705.65	(11468.98, 11946.71)
DPO-PG	400.75	(387.85, 413.72)

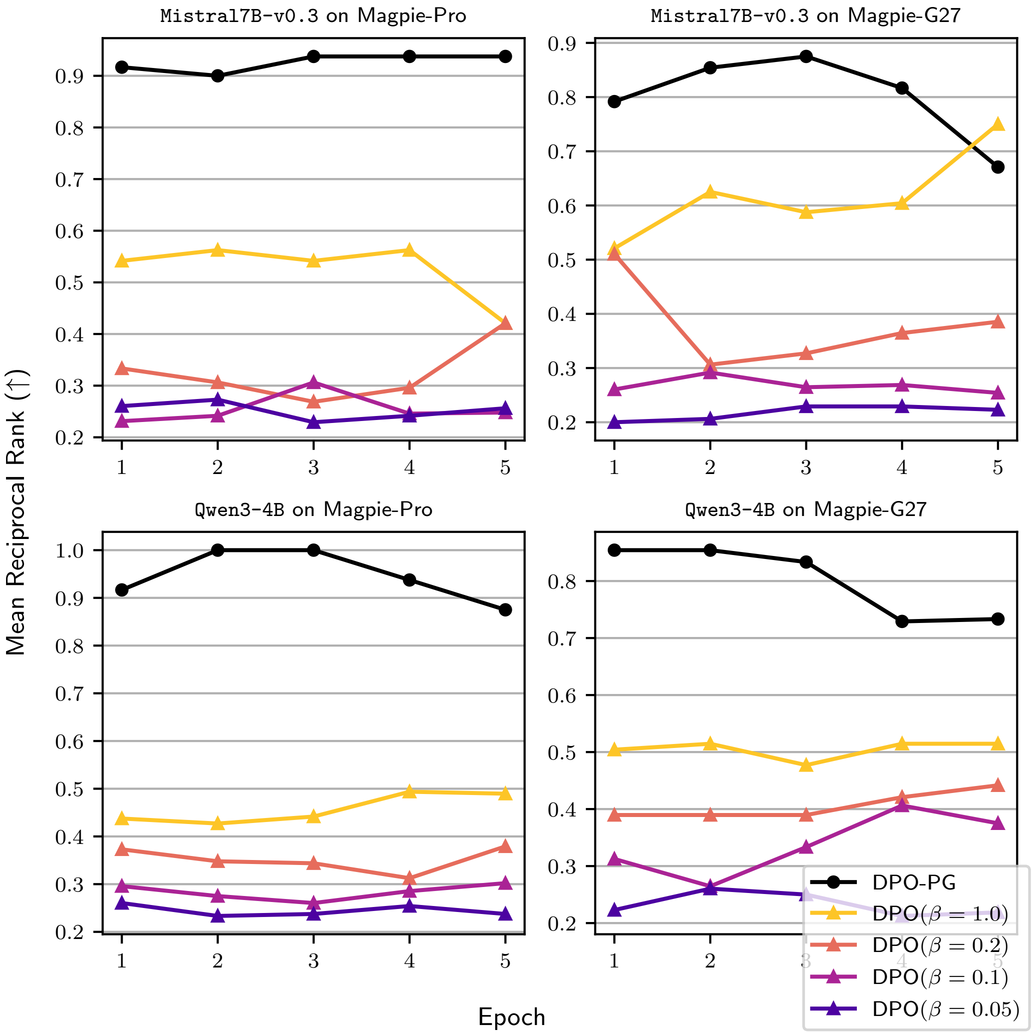

Figure 16: Mean reciprocal rank (MRR) across 8 knowledge-intensive QA benchmarks during training. The MRR is computed following the procedure in Appendix M.2. Preventing LLD (DPO-PG) outperforms standard DPO which exhibits LLD. As DPO-PG learns a low-entropy DID compared to standard DPO (Table 4), this suggests that the knowledge for factual QA is mainly associated with low-entropy DID.

M Experimental setup

M.1 Controlled setting

We conduct controlled experiments involving Energy Based Models (EBMs) in a free-tier Google Colaboratory[8] CPU environment, using PyTorch (Paszke et al., 2019). We use `torch.float32` as the default data type. We set the total class size as 32, and use a batch size of 512 and fix the training seed to 42 for reproducibility. We utilize the RMSprop (Tieleman & Hinton, 2012) optimizer with gradient clipping at maximum norm of 1.0. We use a constant learning rate of 0.001.

Figures 1 and 9. For fair comparison, we follow Meng et al. (2024) in extensively searching the hyper-parameters for the following baseline methods:

- SLiC (Zhao et al., 2023b): $\beta \in \{0.1, 0.5, 1.0, 2.0\}, \lambda \in \{0.1, 0.5, 1.0, 10.0\}$.
- ORPO (Hong et al., 2024): $\beta \in \{0.1, 0.5, 1.0, 2.0\}$.
- SimPO (Meng et al., 2024): $\beta \in \{2.0, 2.5\}, \gamma \in \{0.3, 0.5, 1.0, 1.2, 1.4, 1.6\}$.
- Cal-DPO (Xiao et al., 2024a): $\beta \in \{0.001, 0.002, 0.003, 0.01, 0.1\}$.

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

For Figure 1 (left) we train for a total of 10,000 steps. For Figure 1 (right) and 9, we train for a total of 7,500 steps due to the large number of training configurations as listed above.

Figure 2. We train for a total of 7,500 training steps. The left plot tests $\beta \in \{0.05, 0.1, 0.5, 1, 2, 5\}$, and the right plot tests $\beta \in \{1, 2, 4, 8, 16, 32\}$ following the β conditions in Theorem 4.1.

Figure 4. We train for a total of 20,000 training steps. For the baseline dataset $\mathcal{D}$ ($\alpha = 1$), we test $\beta \in \{0.25, 0.5, 1, 2, 4\}$, and for the strengthened dataset $\mathcal{D}'$ ($\alpha = 2$), we test $\beta \in \{0.5, 1, 2, 4, 8\}$.

Figure 5. We train for a total of 5,000 training steps, averaging over five training seeds: [42, 43, 44, 45, 46]. We measure the converged JS-divergence by averaging the $\mathbb{D}_{\text{JS}}[\pi^* \| \pi]$ of the last 50 training steps.

M.2 Real-world setting

Magpie-G27 dataset. Magpie-G27 is an instruction-following preference dataset built from the prompts of Magpie-Air[9] and completed with responses generated by a stronger model (`google/gemma-2-27b-it`) (Team, 2024). Prompts in Magpie-G27 are disjoint from those in the Magpie-Pro dataset[10]. For each prompt, we sample five completions via `vLLM` (Kwon et al., 2023) using the following sampling configuration:

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

We then score these completions with a strong off-the-shelf reward model `Skywork/Skywork-Reward-Gemma-2-27B-v0.2` (Liu et al., 2024a) 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. Specifically, we utilize the chat-template of `mistralai/Mistral-7B-Instruct-v0.3` for `Mistral7B-v0.3`, and the chat-template of `Qwen/Qwen3-4B-Instruct-2507` for `Qwen3-4B` with its thinking-tags removed.

[8] https://colab.google/

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

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

Reference policy setup. To prepare the reference policy, we fine-tune the base model on the chosen responses, following standard practice (Rafailov et al., 2023; 2024). We train for one epoch with an effective batch size of 256, using the Adam optimizer (Kingma & Ba, 2015) (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 chat-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 (Tieleman & Hinton, 2012) with 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. For `Qwen3-4B` under DPO-PG, we increase the learning rate to 1×10^{-5}, as this setting leads to more effective optimization, preventing LLD while optimizing the DPO loss (Appendix K). We fix the random seed to 1. Models checkpoints are saved after each epoch and trained in `bfloat16` precision. For DPO trained models, we test $\beta \in \{0.05, 0.1, 0.2, 1.0\}$.

Infrastructure and throughput. All experiments use PyTorch FSDP (Zhao et al., 2023a) on NVIDIA A100 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 `Qwen3-4B` on 4 A100 GPUs requires about the same time for the same data size.

Evaluation. We select the best checkpoint by absolute win-rate on Arena-Hard-v0.1 using `gpt-4.1-nano-2025-04-14` as the judge. Final performance on the Arena-Hard benchmark is reported using the judge `gpt-4.1-2025-04-14` to reduce evaluation costs, following Mao et al. (2024). For Wild-Bench-v2, we use `gpt-4o-2024-08-06` as recommended in the official repository.[11] During inference, we greedy-decode up to 4,096 tokens with `vLLM`. QA benchmarks are evaluated via the `lm-evaluation-harness` (Gao et al., 2024). The QA benchmarks consist of the following: PIQA (Bisk et al., 2020), SIQA (Sap et al., 2019), HellaSwag (Zellers et al., 2019), ARC-Easy/Challenge (Clark et al., 2018), MMLU (Hendrycks et al., 2021), GSM8k (Cobbe et al., 2021), and BoolQ (Clark et al., 2019).

The mean reciprocal rank (MRR) in Table 1 and Figure 16 provides a single aggregated metric for performance across the 8 QA benchmarks, each with its own primary metric. The procedure for measuring MRR is as follows.

1. For each of the 8 benchmarks, we evaluate all models using a pre-defined standard performance metric:
    - **ARC-Easy/Challenge, BoolQ, MMLU, PIQA, SIQA:** Accuracy.
    - **HellaSwag:** Normalized Accuracy.
    - **GSM8K:** Exact Match (flexible-extract).
2. Based on these scores, we rank the models for each benchmark.
3. We then calculate the reciprocal of each model's rank and average these reciprocal ranks across all 8 benchmarks to obtain the final MRR score.

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