

Intrinsic Mutual Information as a Modulator for Preference Optimization

Peng Liao¹ Peijia Zheng¹ Lingbo Li² Shangsong Liang^{1*} Lin Chen^{3*}

¹Sun Yat-sen University ²University of Warwick ³Macao Polytechnic University
liavonpenn@gmail.com liangshangsong@gmail.com lchen@mpu.edu.mo

Abstract

Offline preference optimization methods, such as Direct Preference Optimization (DPO), offer significant advantages in aligning Large Language Models (LLMs) with human values. However, achieving optimal performance with these methods typically involves additional hyperparameter tuning, resulting in substantial time overhead. Although prior work has proposed a range of improvements, these methods remain limited in effectiveness and have not fully eliminated reliance on hyperparameter tuning. In this work, we propose RMiPO, a lightweight and efficient framework for offline preference optimization. RMiPO leverages intrinsic **R**esponse-level **M**utual **I**nformation for **P**reference **O**ptimization with hyperparameter modulation, dynamically decoupling preference contributions at negligible additional computational cost. Extensive experimental results demonstrate that RMiPO achieves consistently superior performance over existing methods while reducing training overhead by more than 15%. Our code is available at <https://github.com/liavonpenn/rmipo>.

1 Introduction

Large Language Models (LLMs) demonstrate remarkable generation and comprehension capabilities across diverse NLP tasks (Llama, 2025; DeepSeek, 2025; OpenAI, 2025). However, when faced with complex tasks such as mathematical analysis and logical reasoning, LLMs may take incorrect reasoning paths or generate inconsistent conclusions. To address this issue, researchers have proposed Reinforcement Learning from Human Feedback (RLHF) (Ouyang et al., 2022). This approach fine-tunes pretrained LLMs using carefully curated human preference data, improving their reasoning fidelity and alignment with human values.

Although RLHF substantially enhances LLM performance, its intricate reward modeling pipeline and significant computational demands hinder scal-

*Corresponding authors.

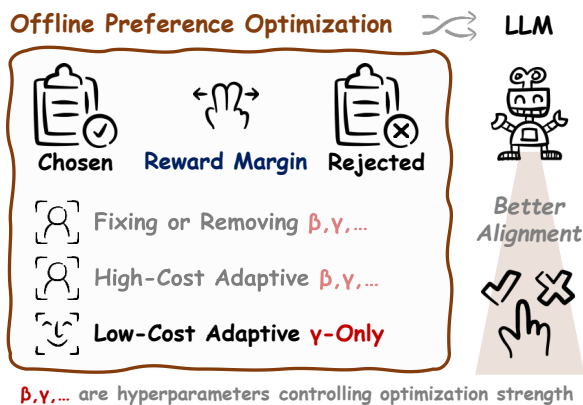


Figure 1: Unlike existing static or high-cost adaptive methods, RMiPO leverages intrinsic mutual information for modulation, enabling adaptive γ -only control to balance training efficiency and alignment performance.

ability. To mitigate these limitations while preserving alignment quality, researchers have introduced offline preference optimization methods such as DPO (Rafailov et al., 2023), IPO (Azar et al., 2024), and KTO (Ethayarajh et al., 2024). These approaches optimize preferences directly through the likelihood outputs of the policy model, thereby obviating the need for an explicit reward model. Building upon this paradigm, SimPO (Meng et al., 2024) further eliminates dependence on a reference model, yielding an approximately 20% reduction in training time for single-round preference optimization relative to DPO. Empirically, these methods perform comparably to, and in some cases surpass, RLHF. Nevertheless, as noted in Rasul et al. (2024), fully realizing their potential still necessitates careful tuning of several key hyperparameters. This process is nontrivial, as each alignment run is costly and configuration changes often require re-establishing optimal hyperparameters.

Recent solutions can be broadly divided into two types: one embeds dynamic mechanisms in preference optimization (Wu et al., 2024; Lee et al., 2025; Wu et al., 2025) to adaptively adjust key hyperparameters, while the other fundamentally alters the optimization paradigm, as exemplified by Xiao et al. (2025b), thereby removing hyper-

parameter dependence. Both approaches exhibit distinct theoretical and empirical advantages. However, the former often requires additional monitoring modules or complex control logic, adding extra training overhead, whereas the latter may degrade under preference drift or inconsistent data (Yang et al., 2024; Gorbатовski et al., 2024; Zeng et al., 2024). This raises a critical research question in LLM alignment: *Is it feasible to develop an efficient method for offline preference optimization?*

In this paper, we provide an affirmative answer to this question. As illustrated in Figure 1, we propose RMiPO, a lightweight and efficient framework for offline preference optimization. At its core is a dynamic modulation mechanism derived from intrinsic mutual information, enabling RMiPO to achieve instance-level adaptive tuning of hyperparameters. Through systematic theoretical analysis, we find that the hyperparameter γ , which controls the reward margin, plays a dominant role in preference optimization, while the intrinsic mutual information of responses provides a principled basis for its adaptive adjustment. Building on the above insight, RMiPO adaptively adjusts γ during optimization, enabling decoupled modeling of preference contributions at negligible computational overhead.

The primary contributions of this work are as follows: (1) **Lightweight Framework.** We propose RMiPO, a novel lightweight offline preference optimization framework that leverages intrinsic mutual information to enable instance-level adaptive hyperparameter control. (2) **Insightful Analysis.** We conduct a comprehensive examination of the roles of hyperparameters in offline preference optimization framework and offer meaningful insights that can guide future practice and research. (3) **Superior Performance.** Through extensive evaluations on benchmarks such as AlpacaEval 2 and MT-Bench, RMiPO reduces training costs by approximately 15–20% while achieving performance clearly superior to state-of-the-art baselines.

2 Preliminary

Problem Formulation. This paper focuses on the offline preference alignment task. We assume access to a dataset $\mathbb{D} = \{(x^{(i)}, y_w^{(i)}, y_l^{(i)})\}_{i=1}^N$ of N preference triplets, where each sample consists of a prompt $x^{(i)}$ and a pair of responses $y_w^{(i)}$ (preferred) and $y_l^{(i)}$ (less-preferred). Each preference triplet is labeled by a preference oracle, such as humans or LLM-based evaluators, which identifies

the preferred response. To learn from \mathbb{D} , Christiano et al. (2017) adopted the Bradley–Terry (BT) model (Bradley and Terry, 1952) as the basis for learning. In the BT formulation, the probability that y_w is preferred to y_l given x is:

$$\mathbb{P}(y_w \succ y_l | x) = \sigma(r(x, y_w) - r(x, y_l)), \quad (1)$$

where $\sigma(\cdot)$ is the logistic function and $r(x, y)$ is the implicit quality score induced by the current policy. Under this setup, we directly learn a parameterized policy model $\pi_\theta(y | x)$, using the preference signal implicitly encoded in $r(x, y)$ to guide updates. The policy is optimized solely on \mathbb{D} and progressively aligns its generation with the oracle’s preferences. **Direct Preference Optimization (DPO)** (Rafailov et al., 2023) is a widely used method for offline preference optimization. Its core idea is to leverage the log-likelihood of π_θ to construct an implicit, preference-aligned reward by relating the optimal policy in closed form to a reference policy π_{ref} . Specifically, DPO defines the reward as $r(x, y) = \beta \log \frac{\pi_\theta(y|x)}{\pi_{\text{ref}}(y|x)} + \beta \log Z(x)$, where $Z(x)$ is a partition function independent of y . The hyperparameter $\beta > 0$ scales the log-likelihood difference to control KL regularization and keep π_θ close to π_{ref} . Under this reward, the objective becomes

$$\begin{aligned} \mathcal{L}_{\text{DPO}} &= -\mathbb{E}_{(x, y_w, y_l) \sim \mathbb{D}} [\log \sigma(\Delta r)], \text{ where} \\ \Delta r &= \beta \log \frac{\pi_\theta(y_w|x)}{\pi_{\text{ref}}(y_w|x)} - \beta \log \frac{\pi_\theta(y_l|x)}{\pi_{\text{ref}}(y_l|x)}. \end{aligned} \quad (2)$$

This objective encourages increasing the preferred response’s log-likelihood ratio and decreasing the less-preferred one relative to the reference policy.

Simple Preference Optimization (SimPO) (Meng et al., 2024) removes DPO’s reliance on a reference model and mitigates length bias in autoregressive generation. It introduces a length-normalized reward $\tilde{r}(x, y) = \frac{\beta}{|y|} \log \pi_\theta(y|x)$, where $|y|$ is the response length. To further amplify preference signal, SimPO incorporates a margin hyperparameter $\gamma > 0$ within the BT formulation, requiring the preferred response to exceed the less-preferred one by at least γ . This yields the following objective:

$$\mathcal{L}_{\text{SimPO}} = -\mathbb{E}_{(x, y_w, y_l) \sim \mathbb{D}} [\log \sigma(\Delta \tilde{r} - \gamma)], \quad (3)$$

where $\Delta \tilde{r} = \tilde{r}(x, y_w) - \tilde{r}(x, y_l)$ by definition.

Alignment performance under the above and other offline preference objectives (Appendix A) is highly sensitive to hyperparameters, especially β and γ , whose optimal values demand costly tuning.

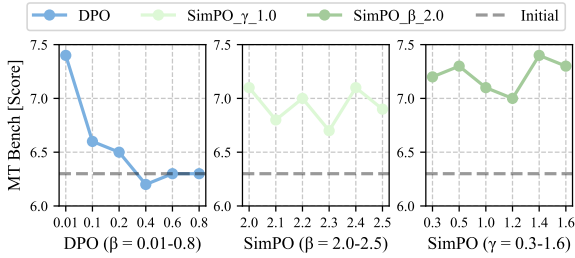


Figure 2: Hyperparameter sensitivity of Mistral-7B-Base on MT-Bench. DPO is evaluated across varying β , while SimPO is tested under two settings: varying β with fixed $\gamma = 1.0$, and varying γ with fixed $\beta = 2.0$. The dashed line denotes the SFT-initialized baseline.

To this end, we introduce RMiPO, with an instance-level adaptive mechanism that significantly mitigates hyperparameter tuning overhead.

3 Methodology

In this section, we address two core design questions: (1) Which hyperparameters necessitate adaptive adjustment, and (2) how to design effective mechanisms for such adaptation. §3.1 presents two key insights, and in §3.2 we introduce RMiPO, a novel offline preference optimization framework that effectively resolves both of these challenges.

3.1 Motivation and Key Insights

The performance of offline preference optimization depends on hyperparameter choice, particularly the trade-off between policy regularization and reward scaling. Through extensive experiments and theoretical analysis, we have distilled two key insights. **Insight I.** *In offline preference optimization, optimal performance is primarily determined by γ .* As shown in Appendix Table 4, the offline preference learning paradigm involves more than two hyperparameters, but its core remains β , which scales reward differences, and γ , which controls reward margins. We fine-tuned Mistral-7B-Base and conducted sensitivity analyses on MT-Bench (Zheng et al., 2023). Although training curves and in-domain validation are similar across configurations, MT-Bench scores vary sharply (Figure 2). In DPO, except for $\beta = 0.01$, which performs relatively well but is not a universal solution (Wu et al., 2025), all other β settings lead to substantial performance degradation, in some cases falling even below the SFT initialization baseline. In contrast, in SimPO, adjusting either β or γ improves performance, with γ exerting an especially pronounced positive effect.

While incorporating γ into DPO yields negli-

ble improvements (Meng et al., 2024), it is indispensable for SimPO, as its removal degrades performance. This motivates us to reexamine the DPO objective (Eq. 2), we observe that although it lacks γ explicitly, the reference model π_{ref} induces an **instance-level implicit margin**. Specifically, the DPO loss can be rewritten analogously to SimPO:

$$\mathcal{L}_{\text{DPO}} = \log \sigma \left(\beta \log \pi_{\theta}(y_w|x) - \beta \log \pi_{\theta}(y_l|x) - \underbrace{(\beta \log \pi_{\text{ref}}(y_w|x) - \beta \log \pi_{\text{ref}}(y_l|x))}_{\text{Instance-level Implicit Margin } \gamma_{\text{ref}}} \right). \quad (4)$$

Based on this observation, we unify the optimization objectives of DPO and SimPO as $\mathcal{L} = -\mathbb{E} [\log \sigma(\Delta\mathbb{R} - \gamma)]$, where $\Delta\mathbb{R} = \beta\Delta \log$ denotes the reward difference induced by the log-likelihood gap of the policy model. This formulation encompasses Δr in DPO and $\Delta\tilde{r}$ in SimPO.

Under this unified framework, policy updates follow the gradient descent rule $\theta \leftarrow \theta - \eta \nabla_{\theta} \mathcal{L}(\theta; \mathbb{D})$, where $\eta > 0$ denotes the learning rate. The gradient of the objective can be written explicitly as

$$\nabla_{\theta} \mathcal{L} = -\beta \mathbb{E} [(1 - \sigma(\Delta\mathbb{R} - \gamma)) \nabla_{\theta} \Delta\mathbb{R}]. \quad (5)$$

It is noteworthy that β serves merely as a scalar multiplier rescaling gradients, whereas γ appears directly inside the sigmoid and directly determines each sample’s gradient-sensitivity region. Under the first-order optimality condition, stationary points satisfy $\nabla_{\theta} \mathcal{L} = 0$, which, after substituting the gradient and noting $\beta > 0$, is equivalent to $\mathbb{E} [(1 - \sigma(\beta\Delta \log - \gamma)) \nabla_{\theta} \Delta \log] = 0$. Thus, the stationary set is essentially governed by γ , as it modulates per-sample weighting via $\sigma(\beta\Delta \log - \gamma)$, whereas β only changes the sigmoid’s steepness without shifting stationary points. These observations clearly indicate that γ should serve as the **primary parameter for adaptive tuning** in offline preference optimization framework.

Insight II. *Modulating γ via mutual information differences offers a reliable, lightweight scheme.* An intuitive and previously explored strategy (Wu et al., 2024) is to adaptively adjust γ using log-likelihood differences computed by the policy model over preference pairs. However, our empirical analysis of the Anthropic-HH dataset (Bai et al., 2022), as shown in Figure 3, reveals several issues. For both Mistral-7B-Base and Llama-8B-Base, the log-likelihood differences in "Helpful" and "Harmless" categories cluster tightly near

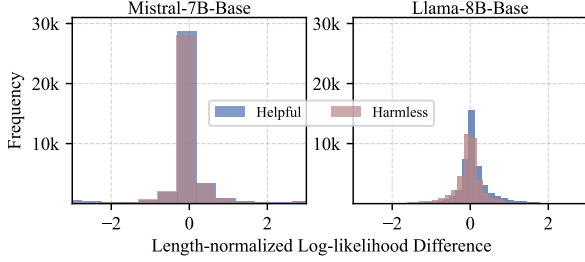


Figure 3: Near-zero, overlapping log-likelihood differences provide a weak signal for adaptive tuning.

zero, exhibit asymmetry, and substantially overlap between preferred and non-preferred samples. Moreover, their distributional shapes vary markedly across models and preference. These properties indicate that likelihood differences are a weak, sensitive, low-signal-to-noise signal with limited effectiveness for adaptive mechanisms built upon them.

This phenomenon motivates a deeper examination of intrinsic limitations in log-likelihood differences. Although $\log \pi_\theta(y | x)$ is intended to measure semantic relevance, it in practice reflects a composite signal entangled with multiple confounding factors. Applying Bayes’ rule and taking the logarithm yields: $\log \pi_\theta(y|x) = \log \frac{\pi_\theta(x|y)\pi_\theta(y)}{\pi_\theta(x)}$. Here, $\log \pi(x|y)$ captures the semantic coupling between the response and the prompt (the "reverse relevance" to be optimized), while $\log \pi(y)$ reflects the response’s prior popularity. The term $\log \pi(x)$ is a constant that does not affect preference ranking. If the adjustment mechanism is based on the policy model’s log-likelihood gap $\Delta \log = \log \frac{\pi_\theta(y_w|x)}{\pi_\theta(y_l|x)}$:

$$\gamma \propto \Delta \log = \underbrace{\log \frac{\pi_\theta(x|y_w)}{\pi_\theta(x|y_l)}}_{\text{Relevance Gap}} + \underbrace{\log \frac{\pi_\theta(y_w)}{\pi_\theta(y_l)}}_{\text{Popularity Bias}}, \quad (6)$$

then the resulting signal inevitably entangles true relevance differences with popularity bias. When y_w is a generic, fluent but low-informative “boilerplate” response such as “That’s a great question” (Li et al., 2016), $\log \pi(y_w)$ becomes large. Consequently, the policy model tends to favor such safe yet uninformative responses. Although sequence length replaces reference-model constraints (Wu et al., 2024; Xiao et al., 2025b), it remains insufficient to mitigate popularity bias.

In contrast, when incorporating a reference model, the adjustment signal is formulated as $\Delta \log = \log \frac{\pi_\theta(y_w|x)}{\pi_{\text{ref}}(y_w|x)} - \log \frac{\pi_\theta(y_l|x)}{\pi_{\text{ref}}(y_l|x)}$. When the policy and reference models have similar language modeling capabilities (reference is typically initialized policy), this formulation naturally cancels

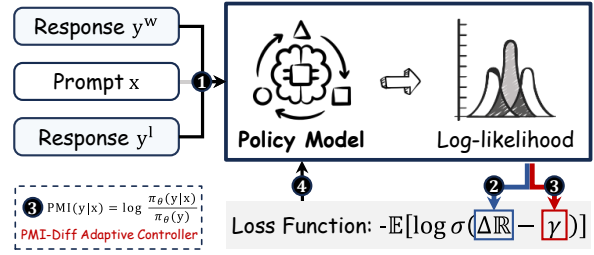


Figure 4: RMiPO training workflow: four-step reference-free policy optimization using mutual information.

prior popularity bias. Specifically, under the assumption that $\pi_\theta \approx \pi_{\text{ref}}$, we have $\log \pi_\theta(y) - \log \pi_{\text{ref}}(y) \approx 0$. This observation not only explains DPO’s robustness to generic responses but also reveals the reason behind the effectiveness of recent adaptive tuning methods (Lee et al., 2025; Wu et al., 2025). However, this comes at the cost of additional model storage and inference overhead.

Existing strategies face an **inherent trade-off**: either they fail to fully eliminate response-prior contamination in the adjustment signal, or rely on reference models, which impose additional computational and storage overhead. This raises a critical question: is there an adjustment signal capable of decoupling semantic relevance from response priors without relying on a reference model? We propose to leverage pointwise mutual information (PMI), a well-established information-theoretic measure (Chen et al., 2024; Liu et al., 2025b; Xiao et al., 2025a), as the adaptive signal for margin control. Formally, under the policy model π_θ , PMI is defined as $\text{PMI}(y|x) = \log \frac{\pi_\theta(y|x)}{\pi_\theta(y)}$. The resulting adjustment signal is $\Delta \text{pmi} = \text{PMI}(y_w|x) - \text{PMI}(y_l|x)$. Substituting the Bayes decomposition yields:

$$\gamma \propto \Delta \text{pmi} = \log \pi_\theta(x|y_w) - \log \pi_\theta(x|y_l). \quad (7)$$

This analysis provides an information-theoretic justification for using mutual-information differences as the adjustment signal for the hyperparameter γ ; the full derivation is provided in Appendix B.

3.2 Proposed Framework: RMiPO

These insights establish two principles: treating γ as the key adaptive variable and updating it via mutual information differences. To this end, we propose RMiPO, a novel optimization framework.

Overview. Figure 4 illustrates the RMiPO training workflow. Given a prompt x and a preference pair (y_w, y_l) , the policy model π_θ computes pointwise mutual information for each response, denoted

as $\text{PMI}(y|x)$. Unlike the raw log-likelihood difference, which entangles semantic relevance with response priors, the PMI difference Δ_{pmi} more directly disentangles conditional relevance from marginal popularity. This signal (Step ③) thus enables reliable and efficient adaptation of the reward margin γ and offers advantages over prior work. Details are provided in the following sections.

Objective. In offline preference optimization, the policy π_θ increases $\Delta_{\mathbb{R}}$ by maximizing the log-likelihood ratio between preferred and less-preferred responses, defining the gradient direction for preference learning. The reward margin γ gates updates based on semantic separability: samples with $\Delta_{\mathbb{R}} > \gamma$ are suppressed, others contribute. To this end, RMiPO employs the pointwise mutual information difference, $\Delta_{\text{pmi}}(x, y_w, y_l) = \text{PMI}(y_w|x) - \text{PMI}(y_l|x)$, as an intrinsic metric for semantic separability. This signal eliminates interference from non-semantic factors, such as prior frequencies. Building on this, RMiPO introduces a key design: the reward margin γ is negatively correlated with Δ_{pmi} . Under this mechanism, semantically distinct pairs (high Δ_{pmi}) receive smaller margins to promote saturation and reduce redundancy. Conversely, ambiguous samples (low Δ_{pmi}) retain larger margins to sustain gradients on semantic differences at the decision boundary.

As shown in Fig. 3, in early training the policy π_θ lacks reliable preference discrimination, resulting in an effectively ungated margin regime ($\Delta_{\mathbb{R}} < \gamma$) where most samples remain gradient-active. As learning progresses, the margin γ progressively re-allocates gradients, yielding a smooth transition from coarse discrimination to fine-grained alignment. To model this nonlinear dependency, we define the dynamic margin as a rectified exponential decay of the pointwise mutual information difference: $\gamma(x, y_w, y_l) = \gamma_{\min} + (\gamma_{\max} - \gamma_{\min}) \cdot \exp(-\max(0, \Delta_{\text{pmi}}))$. The rectification introduces asymmetry: when $\Delta_{\text{pmi}} \leq 0$, the margin is fixed at γ_{\max} , enforcing a strong constraint that continuously penalizes inverted semantic preferences; otherwise, the margin decays exponentially with increasing confidence. This design preserves optimization signals in low- and medium-confidence regions while preventing premature margin collapse.

Moreover, inspired by SimPO (Meng et al., 2024), we incorporate length normalization into the reward signal to further stabilize training dynamics. Notably, as shown in the Appendix Table 5, even in settings without length normalization, our method

remains effective and consistently improves performance. Integrating the above design, the final loss function of RMiPO, $\mathcal{L}_{\text{RMiPO}}$, is formulated as:

$$-\mathbb{E}_{(x, y_w, y_l) \sim \mathbb{D}} [\log \sigma(\Delta_{\mathbb{R}} - \gamma(x, y_w, y_l))]. \quad (8)$$

Here, $\Delta_{\mathbb{R}} = \frac{\beta}{|y_w|} \log \pi_\theta(y_w|x) - \frac{\beta}{|y_l|} \log \pi_\theta(y_l|x)$ represents the preference score difference after length normalization. By explicitly coupling Δ_{pmi} with γ , RMiPO achieves adaptive perception and response to the preference structure. This framework not only has a solid theoretical foundation but also demonstrates computational feasibility, offering a new paradigm for offline preference optimization that combines efficiency and robustness. Further details and pseudocode are in the Appendix C.

4 Experiments

4.1 Experimental Setups

Following (Xiao et al., 2025b), we perform preference optimization in two settings (Base and Instruct) across two model families, namely Llama3-8B (Grattafiori et al., 2024) and Mistral-7B (Jiang et al., 2023). In the Base setting, the initial policy is constructed using the supervised fine-tuning (SFT) pipeline of Zephyr (Tunstall et al., 2023). In the Instruct setting, we directly adopt the officially released instruction-tuned models as the initial policy, skipping the SFT stage. Subsequently, these models are further optimized for preferences on the UltraFeedback dataset (Cui et al., 2024). All experiments are implemented using the alignment-handbook (Tunstall et al.) and conducted on four NVIDIA A800 (80GB) GPUs with batch size 128.

The resulting policy models are evaluated on multiple authoritative benchmarks, including the Open LLM Leaderboard v2 (Fourrier et al., 2024; Gao et al., 2024), AlpacaEval2 (Li et al., 2023), and MT-Bench (Zheng et al., 2023). To further assess safety alignment, we conduct targeted fine-tuning and evaluation on the Anthropic-HH (Helpful and Harmless) dataset (Bai et al., 2022). RMiPO is compared with **the standard offline approaches** (DPO (Rafailov et al., 2023), SLiC (Zhao et al., 2023), IPO (Azar et al., 2024), KTO (Ethayarajh et al., 2024), CPO (Xu et al., 2024), SimPO (Meng et al., 2024)), and **the solutions addressing hyperparameter sensitivity** (α -DPO (Wu et al., 2025), β -DPO (Wu et al., 2024), ϵ -DPO (Lee et al., 2025), SimPER (Xiao et al., 2025b)). For each baseline, we conduct extensive tuning within its recommended hyperparameter space and report the

Table 1: AlpacaEval 2 and MT-Bench results under four settings. LC and WR denote the widely adopted length-controlled and raw win rates, respectively. RMiPO consistently achieves measurable gains across various settings.

| Method | Mistral-7B-Base | | | Mistral-7B-Instruct | | | Llama3-8B-Base | | | Llama3-8B-Instruct | | |
|-----------------|-----------------|-------------|------------|---------------------|-------------|------------|----------------|-------------|------------|--------------------|-------------|------------|
| | AlpacaEval 2 | | MT-Bench | AlpacaEval 2 | | MT-Bench | AlpacaEval 2 | | MT-Bench | AlpacaEval 2 | | MT-Bench |
| | LC (%) | WR (%) | GPT-4 | LC (%) | WR (%) | GPT-4 | LC (%) | WR (%) | GPT-4 | LC (%) | WR (%) | GPT-4 |
| Initial | 8.4 | 6.2 | 6.3 | 17.1 | 14.7 | 7.5 | 6.2 | 4.6 | 6.6 | 26.0 | 25.3 | 8.1 |
| DPO | 15.1 | 12.5 | 7.3 | 26.8 | 24.9 | 7.6 | 18.2 | 15.5 | 7.7 | 40.3 | 37.9 | 8.0 |
| SLiC | 10.9 | 8.9 | 7.4 | 24.1 | 24.6 | 7.8 | 12.3 | 13.7 | 7.6 | 26.9 | 27.5 | 8.1 |
| IPO | 11.8 | 9.4 | 7.2 | 20.3 | 20.3 | 7.8 | 14.4 | 14.2 | 7.4 | 35.6 | 35.6 | 8.3 |
| KTO | 13.1 | 9.1 | 7.0 | 24.5 | 23.6 | 7.7 | 14.2 | 12.4 | 7.8 | 33.1 | 31.8 | 8.2 |
| CPO | 9.8 | 8.9 | 6.8 | 23.8 | 28.8 | 7.5 | 10.8 | 8.1 | 7.4 | 28.9 | 32.2 | 8.0 |
| SimPO | 21.5 | 20.8 | 7.3 | 32.1 | 34.8 | 7.6 | 22.0 | 20.3 | 7.7 | 44.7 | 40.5 | 8.0 |
| α -DPO | 21.0 | 20.5 | 7.5 | 32.3 | 32.6 | 8.0 | 24.6 | 22.7 | 7.7 | 46.6 | 38.1 | 8.1 |
| β -DPO | 15.7 | 16.8 | 7.3 | 23.6 | 20.4 | 7.7 | 23.5 | 20.7 | 7.7 | 43.4 | 38.2 | 8.0 |
| ϵ -DPO | 22.0 | 20.8 | 7.4 | 35.6 | 29.6 | 7.8 | 26.0 | 23.4 | 7.6 | 46.4 | 44.9 | 8.0 |
| SimPER | 22.4 | 21.3 | 7.5 | 37.8 | 39.5 | 7.8 | 25.2 | 22.9 | 7.7 | 48.5 | 45.7 | 8.2 |
| RMiPO | 22.3 | 21.4 | 7.5 | 38.1 | 37.0 | 7.8 | 25.2 | 23.6 | 7.8 | 49.1 | 46.3 | 8.3 |

best performance. All experimental details are provided in Appendices C and A.

4.2 Experimental Results

Results on instruction-following. As shown in Table 1, we report the performance of each model on commonly used instruction-following benchmarks. All preference optimization methods improve over the initial policy, with RMiPO achieving the best overall performance without requiring manual hyperparameter tuning. Under the Llama3-8B-Instruct setting, RMiPO attains 49.1% on the LC metric of AlpacaEval 2 and 8.3 on MT-Bench, both surpassing state-of-the-art approaches. In the Mistral-7B-Instruct setting, RMiPO achieves an LC win rate of 38.1%, corresponding to improvements of 11.3 and 6 percentage points over DPO and SimPO, respectively, demonstrating its effectiveness in capturing human preferences. Although α -DPO and ϵ -DPO achieve moderate gains through additional control variables or optimized loss functions, they remain sensitive to hyperparameter choices and incur higher training costs. By contrast, RMiPO leverages a fully adaptive hyperparameter mechanism to simplify training while maintaining high-quality alignment. Its performance may be slightly lower than SimPER in rare cases, but the gap is minimal. Furthermore, MT-Bench exhibits low discriminative power across methods, likely due to its limited evaluation set and single-instance scoring protocol. Examples of RMiPO generated responses provided in Appendix D further illustrate that its outputs are not only high in quality but also structurally coherent and well-organized.

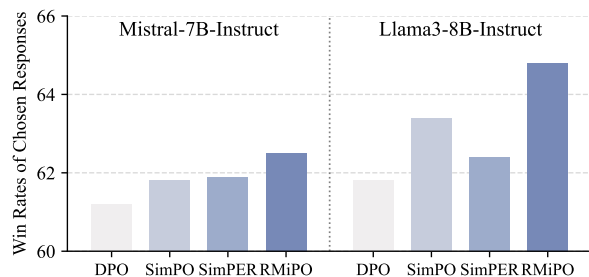


Figure 5: Performance (win rates) of chosen responses on the Anthropic-HH dataset across multiple methods.

Results on downstream tasks. To further evaluate the generalization capability of RMiPO, we conduct a systematic comparative analysis on multiple downstream tasks from the HuggingFace Leaderboard, including IFEval, BBH, MATH, GPQA, MUSR, and MMLU-PRO, with the results summarized in Table 2. Under the Mistral-7B-Instruct setting, RMiPO achieves the best average ranking of 2.50, substantially outperforming baseline methods such as DPO (6.33) and SimPO (3.67). This advantage is further amplified under the Llama3-8B-Instruct setting, where RMiPO attains an average ranking of 1.83. At the task level, RMiPO consistently achieves the highest scores on IFEval across both backbone models (52.76 and 69.07). In addition, RMiPO demonstrates strong performance on MATH, with scores of 3.50 and 8.96, highlighting its effectiveness in structured reasoning and mathematical problem solving. Although methods such as ϵ -DPO yield competitive results on specific tasks (e.g., MUSR or GPQA) by employing more complex optimization strategies, their performance varies considerably across tasks. In contrast, RMiPO achieves consistently strong results

Table 2: Evaluation results on selected leaderboard benchmarks. Average Rank (Avg. Rank) is the mean of per-task ranks for each method, with lower values indicating better performance. Competition ranking resolves ties.

| Model | Method | IFEval | BBH | MATH | GPQA | MUSR | MMLU-PRO | Avg. Score | Avg. Rank |
|----------------------------|-----------------|--------------|--------------|-------------|-------------|-------------|--------------|--------------|-------------|
| Mistral-7B Instruct | DPO | 51.76 | 16.88 | 3.10 | 2.46 | 5.75 | 19.43 | 16.56 | 6.33 |
| | SimPO | 46.87 | 22.38 | 2.87 | 3.80 | 9.76 | 19.96 | 17.61 | 3.67 |
| | α -DPO | 52.10 | 19.20 | 3.35 | 2.80 | 6.05 | 19.88 | 17.23 | 3.83 |
| | β -DPO | 51.95 | 17.05 | 3.27 | 2.74 | 8.91 | 19.84 | 17.29 | 4.67 |
| | ϵ -DPO | 52.05 | 17.25 | 3.50 | 2.85 | 6.88 | 19.75 | 17.05 | 3.83 |
| | SimPER | 47.10 | 22.50 | 2.95 | 3.91 | 9.47 | 20.05 | 17.66 | 3.00 |
| | RMiPO | 52.76 | 22.04 | 3.50 | 3.67 | 9.85 | 19.67 | 18.58 | 2.50 |
| LLama3-8B Instruct | DPO | 67.57 | 28.51 | 8.46 | 2.91 | 3.93 | 29.61 | 23.50 | 5.67 |
| | SimPO | 65.04 | 26.71 | 8.61 | 5.82 | 8.15 | 27.66 | 23.67 | 5.17 |
| | α -DPO | 68.12 | 29.13 | 8.75 | 4.15 | 5.67 | 29.80 | 24.27 | 3.00 |
| | β -DPO | 67.93 | 28.85 | 8.41 | 3.46 | 6.16 | 28.07 | 23.81 | 5.00 |
| | ϵ -DPO | 68.15 | 29.01 | 8.94 | 3.27 | 8.42 | 29.85 | 24.61 | 2.50 |
| | SimPER | 65.91 | 27.97 | 8.74 | 5.91 | 4.15 | 28.67 | 23.56 | 4.67 |
| | RMiPO | 69.07 | 29.01 | 8.96 | 3.50 | 8.41 | 30.46 | 24.90 | 1.83 |

Table 3: Ablation analysis on leaderboard benchmarks. GM: tuning γ ; MI: intrinsic mutual information-based fine-tuning; ED: exponential decay-based update.

| GM | MI | ED | IFEval | BBH | MATH | GPQA | MUSR | MMLU-PRO |
|----|----|----|--------------|--------------|-------------|-------------|-------------|--------------|
| ✓ | | | 66.14 | 27.32 | 8.52 | 3.01 | 4.25 | 28.04 |
| | ✓ | | 65.87 | 26.94 | 8.48 | 3.12 | 5.12 | 28.32 |
| ✓ | ✓ | | 68.21 | 28.47 | 8.74 | 3.28 | 7.31 | 29.56 |
| ✓ | | ✓ | 67.58 | 27.82 | 8.61 | 3.07 | 6.47 | 28.91 |
| | ✓ | ✓ | 68.82 | 29.73 | 8.85 | 4.18 | 8.28 | 29.87 |
| ✓ | ✓ | ✓ | 69.07 | 29.01 | 8.96 | 3.50 | 8.41 | 30.46 |

across diverse downstream tasks without incurring additional parameter search costs, underscoring its superior stability and generalization ability.

Results on safety alignment. The above benchmarks primarily target open-domain tasks. To evaluate RMiPO’s performance in terms of safety and preference alignment, we compare it with baselines on the Anthropic-HH dataset. This dataset comprises the Helpful and Harmless subsets, with alignment effectiveness measured via the win rate of chosen responses. As shown in Figure 5, RMiPO achieves significantly higher average win rates across both subsets compared with DPO, SimPO, and its improved variant SimPER. Under the Llama3-8B-Instruct setting, RMiPO attains an average win rate of 64.8%, surpassing DPO’s 61.8%. Notably, on the Harmless subset, RMiPO demonstrates even greater gains, while maintaining a stable lead without requiring complex hyperparameter tuning on safety-related data, highlighting its robustness and practical utility.

Ablation analysis. We conduct an ablation study of the three key design components in RMiPO on the Llama3-8B-Instruct model, with the results summarized in Table 3. When only one of GM or MI is enabled, the model achieves relatively limited per-

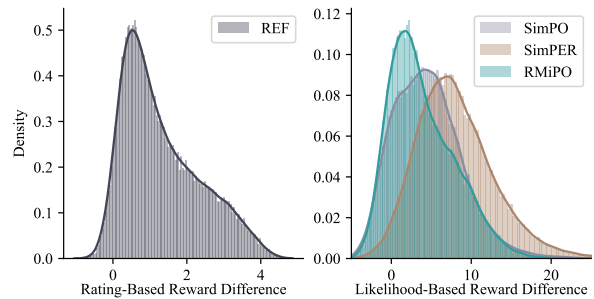


Figure 6: Density analysis of reward differences, with REF serving as the ground-truth reference.

formance across benchmarks (e.g., IFEval scores of 66.14 and 65.87, respectively), indicating that a single component cannot fully unlock the model’s potential. As components are progressively combined, overall performance improves substantially. For example, combining GM and MI increases MMLU-PRO from approximately 28% to 29.56, while the combination of MI and ED performs particularly well on BBH (29.73) and GPQA (4.18). When all three components (GM + MI + ED) are enabled simultaneously, the model achieves the best results on core benchmarks including IFEval (69.07), MATH (8.96), MUSR (8.41), and MMLU-PRO (30.46). These results indicate that the performance gains primarily stem from the synergistic interactions among RMiPO’s components, with the full system delivering the best overall performance.

Density analysis. We analyze the reward density of the Llama3-8B-Instruct model. Specifically, we construct reward differences from the rankings of responses in the UltraFeedback dataset, where each difference is defined as the ranking of the preferred response minus that of the less-preferred response. Since the ranking-based reward differences are discrete, we apply bounded uniform jittering to make

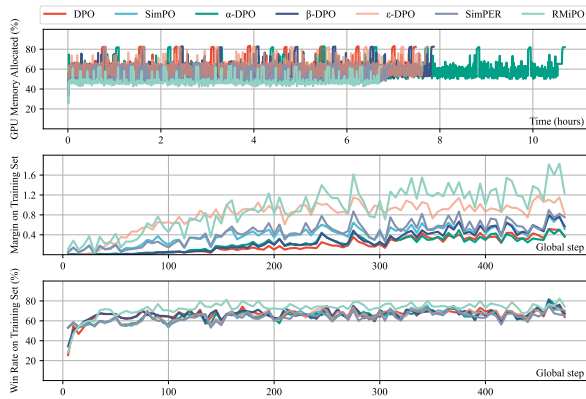


Figure 7: Training efficiency and dynamics, showing cost, margin, and win rate on the training set.

them continuous, thereby approximating the latent reward. This strategy is justified because the original rankings are generated by a higher-quality oracle model, which serves as a reliable proxy for human preferences. As shown in Figure 6, we compare the performance of RMiPO with that of SimPO and SimPER. The results indicate that the reward distribution generated by RMiPO aligns most closely with the reference benchmark distribution. This demonstrates its ability to capture fine-grained differences in human preference intensity and provides strong evidence for the superiority of RMiPO.

Efficiency analysis. To further evaluate the practicality of each algorithm in real-world production environments, we analyzed the computational efficiency and training stability of RMiPO and its baseline methods, focusing on GPU memory usage, training duration, and performance convergence. As shown in Figure 7, RMiPO demonstrates superior efficiency, requiring only approximately 6.75 hours of training with a peak GPU memory usage of about 62%. In comparison, SimPO and SimPER require 6.17 hours and 6.92 hours, respectively, with similar peak memory usage, while DPO and its variants β -DPO and ϵ -DPO exhibit peak memory usage as high as 82% and training durations approaching 8 hours. Notably, α -DPO requires approximately 10 hours of training due to complex dynamic reference model computations and additional preprocessing steps. During training, RMiPO consistently outperforms other methods in Margin and Win Rate metrics on the training set, maintaining a leading position throughout the entire training cycle. Overall, RMiPO combines high efficiency, low resource consumption, and superior performance, making it a highly practical solution for large-scale language model alignment tasks.

5 Related Work

Preference Optimization for LLM. Preference optimization is a key technique for aligning LLMs with human values. This paradigm was originally proposed by (Ouyang et al., 2022) and employs proximal policy optimization for online training. However, the high complexity of online algorithms has driven researchers to shift toward more efficient offline methods. Offline preference optimization (Park et al., 2024; Rafailov et al., 2023; Zhao et al., 2023; Azar et al., 2024; Ethayarajh et al., 2024; Xu et al., 2024; Zeng et al., 2024; Amini et al., 2024; Tang et al., 2024; Liu et al., 2025a,c) directly leverages paired preference data to guide policies toward preferred behaviors. Building on this line of work, (Meng et al., 2024; Hong et al., 2024) proposed an approach that does not require a reference model. Despite their effectiveness, these methods require extensive hyperparameter tuning. **Efficiency Improvement in DPO.** Unlike prior work on data filtering (Muldrew et al., 2024; Yang et al., 2024; Huang et al., 2025; Corrado et al., 2025), this study focuses on minimizing hyperparameter tuning overhead in DPO and its variants. Existing research has explored several strategies: (Wu et al., 2024) and (Lee et al., 2025) dynamically adjust hyperparameters via KL regularization to account for instance-level preference intensity; (Wu et al., 2025; Sun et al., 2025) introduces an implicit reference model, reducing hyperparameters that require tuning; and (Xiao et al., 2025b) proposes the first parameter-free preference optimization objective based on perplexity. While these methods demonstrate success, they remain limited in effectiveness. To address this, we propose RMiPO, an adaptive hyperparameter tuning mechanism based on intrinsic mutual information that improves alignment while reducing optimization overhead.

6 Conclusion

In this paper, we propose RMiPO, a lightweight and efficient offline preference optimization framework. Its core innovation is an adaptive modulation mechanism based on intrinsic response-level mutual information. This mechanism dynamically decouples the contribution of different samples to the overall preference signal, enabling finer-grained alignment control. Theoretical analysis shows that this mutual information-driven strategy offers strong interpretability, establishing a new principled paradigm for sample weighting in preference

optimization. Extensive experiments demonstrate that RMiPO achieves superior alignment gains with lower training costs and favorable convergence.

Limitations

The core advantage of RMiPO lies in its ability to obviate the manual tuning of task-sensitive hyperparameters, rather than reducing the number of tunable parameters. This property substantially enhances the practicality of preference optimization algorithms. Nevertheless, RMiPO still exhibits several limitations that warrant further investigation.

Sensitivity. While γ is the primary driver of adaptivity, its efficacy presupposes that β lies within a reasonable range, as typically ensured by standard model-scaling practices. Although extreme β values destabilize all preference-optimization baselines, RMiPO exhibits greater robustness, with performance degrading more gracefully than static-margin methods. By dynamically adjusting the reward margin γ (and its bounds) as a function of β , the proposed exponential decay mechanism partially mitigates mis-scaled preference signals and reduces performance degradation under suboptimal settings. Although RMiPO cannot compensate for arbitrarily ill-chosen β , its adaptive margin consistently offers a safety buffer over fixed-margin baselines. The coupled dynamics between β and γ therefore warrant further study in the context of optimization stability and generalization.

Scalability. Recent studies suggest increasing policy optimization iterations can further improve instruction-following performance. However, due to computational constraints, this work, like most existing studies, does not conduct extensive validation under large-scale regimes. Our experiments focus on billion-parameter models, which suffice to elucidate RMiPO’s key mechanisms and provide empirical support for its effectiveness. The generalization of RMiPO to ten-billion-parameter models remains to be systematically evaluated.

Adaptability. This work focuses on offline preference optimization with fixed datasets, which limits adaptation to dynamic environments. Extending RMiPO to an online learning framework with real-time interaction feedback could improve its applicability to scenarios such as human–computer interaction and autonomous agents, for example by using an online oracle to provide preferences and intrinsic mutual information to guide exploration.

Ethical Considerations

This work proposes a lightweight framework for LLM alignment. The study does not involve human participants or personal or sensitive data, nor does it enable malicious use; therefore, it raises no ethical concerns. The proposed method aims to improve the robustness and efficiency of value alignment, supporting the reliable deployment of large language models. Furthermore, LLM-based assistants were employed solely for linguistic refinement, with the authors explicitly retaining full responsibility for the research design, technical content, and final conclusions.

Acknowledgments

This work was supported by the Science and Technology Development Fund of the Macau SAR (Grant No. 0119/2025/RIA2) and the National Natural Science Foundation of China (Grant Nos. W2512008, 62572500, and 62272498).

References

- Afra Amini, Tim Vieira, and Ryan Cotterell. 2024. Direct preference optimization with an offset. *Findings of the Association for Computational Linguistics: ACL 2024*, pages 9954–9972.
- Mohammad Gheshlaghi Azar, Zhaohan Daniel Guo, Bilal Piot, Remi Munos, Mark Rowland, Michal Valko, and Daniele Calandriello. 2024. A general theoretical paradigm to understand learning from human preferences. In *International Conference on Artificial Intelligence and Statistics*, pages 4447–4455. PMLR.
- Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, Nicholas Joseph, Saurav Kadavath, Jackson Kernion, Tom Conerly, Sheer El-Showk, Nelson Elhage, Zac Hatfield-Dodds, Danny Hernandez, Tristan Hume, Scott Johnston, Shauna Kravec, Liane Lovitt, Neel Nanda, Catherine Olsson, Dario Amodei, Tom Brown, Jack Clark, Sam McCandlish, Chris Olah, Ben Mann, and Jared Kaplan. 2022. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*.
- Ralph Allan Bradley and Milton E Terry. 1952. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39(3/4):324–345.
- Junjie Chen, Xiangheng He, and Yusuke Miyao. 2024. Language model based unsupervised dependency parsing with conditional mutual information and

- grammatical constraints. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 6355–6366.
- Paul F Christiano, Jan Leike, Tom Brown, Miljan Martić, Shane Legg, and Dario Amodei. 2017. Deep reinforcement learning from human preferences. *Advances in neural information processing systems*, 30.
- Nicholas E Corrado, Julian Katz-Samuels, Adithya M Devraj, Hyokun Yun, Chao Zhang, Yi Xu, Yi Pan, Bing Yin, and Trishul Chilimbi. 2025. Automixalign: Adaptive data mixing for multi-task preference optimization in llms. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 20234–20258.
- Ganqu Cui, Lifan Yuan, Ning Ding, Guanming Yao, Bingxiang He, Wei Zhu, Yuan Ni, Guotong Xie, Ruobing Xie, Yankai Lin, Zhiyuan Liu, and Maosong Sun. 2024. [Ultrafeedback: Boosting language models with scaled ai feedback](#). *Preprint*, arXiv:2310.01377.
- DeepSeek. 2025. Introducing deepseek-v3.2-exp. <https://api-docs.deepseek.com/news/news250929>. Accessed: 2025-10-8.
- Kawin Ethayarajh, Winnie Xu, Niklas Muennighoff, Dan Jurafsky, and Douwe Kiela. 2024. Kto: Model alignment as prospect theoretic optimization. *arXiv preprint arXiv:2402.01306*.
- Clémentine Fourier, Nathan Habib, Alina Lozovskaya, Konrad Szafer, and Thomas Wolf. 2024. Open llm leaderboard v2. https://huggingface.co/spaces/open-llm-leaderboard/open_llm_leaderboard.
- 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. 2024. [The language model evaluation harness](#).
- Alexey Gorbатовski, Boris Shaposhnikov, Alexey Malakhov, Nikita Surnachev, Yaroslav Aksenov, Ian Maksimov, Nikita Balagansky, and Daniil Gavrilov. 2024. Learn your reference model for real good alignment. *arXiv preprint arXiv:2404.09656*.
- Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurelien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Roziere, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong, Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Al-lonsius, Daniel Song, Danielle Pintz, Danny Livshits, Danny Wyatt, David Esiobu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino, and 508 others. 2024. The llama 3 herd of models. *arXiv e-prints*, pages arXiv–2407.
- Jiwoo Hong, Noah Lee, and James Thorne. 2024. Orpo: Monolithic preference optimization without reference model. In *2024 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.
- Kexin Huang, Junkang Wu, Ziqian Chen, Xue Wang, Jinyang Gao, Bolin Ding, Jiancan Wu, Xiangnan He, and Xiang Wang. 2025. Larger or smaller reward margins to select preferences for llm alignment? In *Forty-second International Conference on Machine Learning*.
- Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Léo Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. [Mistral 7b](#). *Preprint*, arXiv:2310.06825.
- Sangkyu Lee, Janghoon Han, Hosung Song, Stanley Jungkyu Choi, Honglak Lee, and Youngjae Yu. 2025. K1 penalty control via perturbation for direct preference optimization. *arXiv preprint arXiv:2502.13177*.
- Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and William B Dolan. 2016. A diversity-promoting objective function for neural conversation models. In *Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: human language technologies*, pages 110–119.
- Jiyong Li, Dilshod Azizov, Shangsong Liang, and 1 others. 2024. Contrastive continual learning with importance sampling and prototype-instance relation distillation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 13554–13562.
- Xuechen Li, Tianyi Zhang, Yann Dubois, Rohan Taori, Ishaan Gulrajani, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2023. AlpacaEval: An automatic evaluator of instruction-following models. https://github.com/tatsu-lab/alpaca_eval.
- Tianqi Liu, Zhen Qin, Junru Wu, Jiaming Shen, Misha Khalman, Rishabh Joshi, Yao Zhao, Mohammad Saleh, Simon Baumgartner, Jialu Liu, Peter J. Liu, and Xuanhui Wang. 2025a. Lipo: Listwise preference optimization through learning-to-rank. In *Proceedings of the 2025 Conference of the Nations of*

- the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 2404–2420.
- Tianyu Liu, Jirui Qi, Paul He, Arianna Bisazza, Mrinmaya Sachan, and Ryan Cotterell. 2025b. Pointwise mutual information as a performance gauge for retrieval-augmented generation. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 1628–1647.
- Yixin Liu, Pengfei Liu, and Arman Cohan. 2025c. Understanding reference policies in direct preference optimization. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 8022–8037.
- Llama. 2025. The llama 4 herd: The beginning of a new era of natively multimodal ai innovation. <https://ai.meta.com/blog/llama-4-multimodal-intelligence/>. Accessed: 2025-10-8.
- Yu Meng, Mengzhou Xia, and Danqi Chen. 2024. Simpo: Simple preference optimization with a reference-free reward. *Advances in Neural Information Processing Systems*, 37:124198–124235.
- William Muldrew, Peter Hayes, Mingtian Zhang, and David Barber. 2024. Active preference learning for large language models. In *International Conference on Machine Learning*, pages 36577–36590. PMLR.
- OpenAI. 2025. Introducing gpt-5. <https://openai.com/index/introducing-gpt-5/>. Accessed: 2025-10-8.
- Long Ouyang, Jeff 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 Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744.
- Ryan Park, Rafael Rafailov, Stefano Ermon, and Chelsea Finn. 2024. Disentangling length from quality in direct preference optimization. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 4998–5017.
- Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. *Advances in neural information processing systems*, 36:53728–53741.
- Kashif Rasul, Edward Beeching, Lewis Tunstall, Leandro von Werra, and Omar Sanseviero. 2024. Preference tuning llms with direct preference optimization methods. <https://huggingface.co/blog/pref-tuning>. Accessed: 2024-11-27.
- Jie Sun, Junkang Wu, Jiancan Wu, Zhibo Zhu, Xingyu Lu, Jun Zhou, Lintao Ma, and Xiang Wang. 2025. Robust preference optimization via dynamic target margins. *arXiv preprint arXiv:2506.03690*.
- Yunhao Tang, Zhaohan Daniel Guo, Zeyu Zheng, Daniele Calandriello, Remi Munos, Mark Rowland, Pierre Harvey Richemond, Michal Valko, Bernardo Avila Pires, and Bilal Piot. 2024. Generalized preference optimization: A unified approach to offline alignment. In *Forty-first International Conference on Machine Learning*.
- Lewis Tunstall, Edward Beeching, Nathan Lambert, Nazneen Rajani, Shengyi Huang, Kashif Rasul, Alvaro Bartolome, Carlos M. Patiño, Alexander M. Rush, and Thomas Wolf. The Alignment Handbook. <https://huggingface.co/alignment-handbook>.
- Lewis Tunstall, Edward Beeching, Nathan Lambert, Nazneen Rajani, Kashif Rasul, Younes Belkada, Shengyi Huang, Leandro von Werra, Clémentine Fourrier, Nathan Habib, Nathan Sarrazin, Omar Sanseviero, Alexander M. Rush, and Thomas Wolf. 2023. Zephyr: Direct distillation of lm alignment. *arXiv preprint arXiv:2310.16944*.
- Junkang Wu, Xue Wang, Zhengyi Yang, Jiancan Wu, Jinyang Gao, Bolin Ding, Xiang Wang, and Xiangnan He. 2025. *Alphadpo: Adaptive reward margin for direct preference optimization*. *Preprint*, arXiv:2410.10148.
- Junkang Wu, Yuexiang Xie, Zhengyi Yang, Jiancan Wu, Jinyang Gao, Bolin Ding, Xiang Wang, and Xiangnan He. 2024. β -dpo: direct preference optimization with dynamic β . In *Proceedings of the 38th International Conference on Neural Information Processing Systems*, pages 129944–129966.
- Teng Xiao, Zhen Ge, Sujay Sanghavi, Tian Wang, Julian Katz-Samuels, Marc Versage, Qingjun Cui, and Trishul Chilimbi. 2025a. Infopo: On mutual information maximization for large language model alignment. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 11699–11711.
- Teng Xiao, Yige Yuan, Zhengyu Chen, Mingxiao Li, Shangsong Liang, Zhaochun Ren, and Vasant G Honavar. 2025b. Simper: A minimalist approach to preference alignment without hyperparameters. *arXiv preprint arXiv:2502.00883*.
- Haoran Xu, Amr Sharaf, Yunmo Chen, Weiting Tan, Lingfeng Shen, Benjamin Van Durme, Kenton Murray, and Young Jin Kim. 2024. Contrastive preference optimization: Pushing the boundaries of llm performance in machine translation. *arXiv preprint arXiv:2401.08417*.

Sen Yang, Leyang Cui, Deng Cai, Xinting Huang, Shuming Shi, and Wai Lam. 2024. Not all preference pairs are created equal: A recipe for annotation-efficient iterative preference learning. *arXiv preprint arXiv:2406.17312*.

Yongcheng Zeng, Guoqing Liu, Weiyu Ma, Ning Yang, Haifeng Zhang, and Jun Wang. 2024. Token-level direct preference optimization. *arXiv preprint arXiv:2404.11999*.

Yao Zhao, Rishabh Joshi, Tianqi Liu, Misha Khalman, Mohammad Saleh, and Peter J Liu. 2023. Slic-hf: Sequence likelihood calibration with human feedback. *arXiv preprint arXiv:2305.10425*.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in neural information processing systems*, 36:46595–46623.

A Revisiting Baseline Methods

This section revisits SOTA baseline methods for offline preference optimization by examining the underlying design motivations and the functional roles of hyperparameters in their loss formulations. Based on this analysis, we summarize and contrast the resulting optimization objectives in Table 4.

- **DPO** (Rafailov et al., 2023): By introducing a difference term of log-likelihood ratios, DPO implicitly models the reward function, thereby avoiding reliance on an explicit reward model in RLHF. This method introduces a tunable hyperparameter β to control the deviation of the policy model from the reference model, enabling a controllable trade-off between preference alignment and policy updates.
- **SLiC** (Zhao et al., 2023): The optimization objective of SLiC incorporates a supervised fine-tuning (SFT) term and is governed by two hyperparameters: the margin parameter δ , which sets the boundary of the ranking loss, and the weighting coefficient λ , which balances the contribution of the SFT term within the overall objective.
- **IPO** (Azar et al., 2024): IPO transforms preference learning into a regression problem with a squared loss by defining a surrogate reward function, thereby avoiding the instability associated with reinforcement learning training. This approach involves a hyperparameter β

that controls the magnitude of the reward signal.

- **CPO** (Xu et al., 2024): CPO directly uses the log-likelihood as the reward signal and jointly optimizes it with a SFT term. This method introduces two hyperparameters: β , which scales the log-probability term, and λ , which adjusts the weight of the SFT term in the joint objective.
- **KTO** (Ethayarajh et al., 2024): KTO learns from unpaired preference data, with its optimization objective governed by three hyperparameters: β , which controls the degree of deviation of the policy model from the reference model, and λ_w and λ_l , which weight the preference terms corresponding to winning and losing responses, respectively.
- **SimPO** (Meng et al., 2024): SimPO does not rely on a reference model and is trained by optimizing a length-regularized objective over response pairs, with its objective function governed by two hyperparameters: β , which scales the log-probability term, and γ , which adjusts the reward margin.

A review of the above paradigms reveals that hyperparameters are ubiquitous across different methods; as demonstrated in prior studies, their tuning is not only unavoidable but also often incurs substantial computational overhead. To address the issue of hyperparameter dependence, researchers have recently proposed several approaches.

- **α -DPO** (Wu et al., 2025): α -DPO aims to combine the advantages of DPO and SimPO to reduce sensitivity to hyperparameters β and γ . This approach introduces a smoothing parameter $\alpha \in [0, 1]$ to interpolate between the two objectives: when $\alpha = 0$, the optimization objective reduces to SimPO; when $\alpha = 1$, it is equivalent to DPO. Central to SimPO is the length-normalized utility $u(x, y_w, y_l) = \frac{\beta}{|y_w|} \log \pi_\theta(y_w|x) - \frac{\beta}{|y_l|} \log \pi_\theta(y_l|x)$, which mitigates the bias introduced by response length. α -DPO incorporates this utility into its unified objective. The method is based on the preference scoring function $M(x, y_w, y_l) = \beta \log \frac{\pi_\theta(y_w|x) \pi_{\text{ref}}(y_l|x)}{\pi_{\text{ref}}(y_w|x) \pi_\theta(y_l|x)}$, which measures the relative difference between the policy π_θ

Table 4: Various preference optimization objectives and hyperparameter search range.

| Method | Objective | Hyperparameter |
|---|--|---|
| <i>Common Offline Preference Optimization Methods</i> | | |
| DPO (Rafailov et al., 2023) | $-\log \sigma \left(\beta \log \frac{\pi_\theta(y_w x)}{\pi_{\text{ref}}(y_w x)} - \beta \log \frac{\pi_\theta(y_l x)}{\pi_{\text{ref}}(y_l x)} \right)$ | $\beta \in [0.01, 0.05, 0.1]$ |
| SLiC (Zhao et al., 2023) | $\max(0, \delta - \log \pi_\theta(y_w x) + \log \pi_\theta(y_l x)) - \lambda \log \pi_\theta(y_w x)$ | $\lambda \in [0.1, 0.5, 1.0, 10.]$ $\beta \in [0.1, 0.5, 1.0, 2.0]$ |
| IPO (Azar et al., 2024) | $\left(\log \frac{\pi_\theta(y_w x)}{\pi_{\text{ref}}(y_w x)} - \log \frac{\pi_\theta(y_l x)}{\pi_{\text{ref}}(y_l x)} - \frac{1}{2\tau} \right)^2$ | $\tau \in [0.01, 0.1, 0.5, 1.0]$ |
| CPO (Xu et al., 2024) | $-\log \sigma \left(\beta \log \pi_\theta(y_w x) - \beta \log \pi_\theta(y_l x) \right) - \lambda \log \pi_\theta(y_w x)$ | $\alpha = 1.0, \beta \in [0.01, 0.05, 0.1]$ |
| KTO (Ethayarajh et al., 2024) | $-\lambda_w \sigma \left(\beta \log \frac{\pi_\theta(y_w x)}{\pi_{\text{ref}}(y_w x)} - z_{\text{ref}} \right) + \lambda_l \sigma \left(z_{\text{ref}} - \beta \log \frac{\pi_\theta(y_l x)}{\pi_{\text{ref}}(y_l x)} \right)$, where $z_{\text{ref}} = \mathbb{E}_{(x,y) \sim \mathcal{D}} [\beta \text{KL}(\pi_\theta(y x) \pi_{\text{ref}}(y x))]$ | $\lambda_l = \lambda_w = 1.0$ $\beta \in [0.01, 0.05, 0.1]$ |
| SimPO (Meng et al., 2024) | $-\log \sigma \left(\frac{\beta}{ y_w } \log \pi_\theta(y_w x) - \frac{\beta}{ y_l } \log \pi_\theta(y_l x) - \gamma \right)$ | $\beta \in [2.0, 4.0, 6.0, 8.0]$ $\gamma \in [0.3, 0.5, 1.0, 1.2, 1.4, 1.6]$ |
| <i>Solutions for Mitigating Hyperparameter Sensitivity</i> | | |
| α -DPO (Wu et al., 2025) | $-\log \sigma \left(u(x, y_w, y_l) - \text{sg}[\gamma + \alpha M^*(x, y_w, y_l)] \right)$ where $u(x, y_w, y_l) = \frac{\beta}{ y_w } \log \pi_\theta(y_w x) - \frac{\beta}{ y_l } \log \pi_\theta(y_l x)$ | $\beta \in [2.5, 10.0], \gamma \in [0.1, 0.3, 0.5]$ $\alpha \in [1e-2, 5e-2, 0.1, 0.2]$ |
| β -DPO (Wu et al., 2024) | $-\log \sigma \left(\beta_i \log \frac{\pi_\theta(y_w x)}{\pi_{\text{ref}}(y_w x)} - \beta_i \log \frac{\pi_\theta(y_l x)}{\pi_{\text{ref}}(y_l x)} \right)$ where $\beta_i = [1 + \alpha(M_i - M_0)]\beta_0$ | $\beta = 0.1, \alpha \in [0.4, 0.6, 0.8, 1.0]$ $\rho \in [0.0, 0.1, 0.2, 0.3, 0.4, 0.5]$ |
| ϵ -DPO (Lee et al., 2025) | $-\log \sigma \left(\beta_i \log \frac{\pi_\theta(y_w x)}{\pi_{\text{ref}}(y_w x)} - \beta_i \log \frac{\pi_\theta(y_l x)}{\pi_{\text{ref}}(y_l x)} \right)$ where $\beta_i \leftarrow \{\beta_\epsilon^-, \beta_\epsilon^+, \beta\}$ | $\beta = 0.01$ $\epsilon \in [0.005, 0.01, 0.02]$ |
| <i>Solutions for Eliminating Hyperparameter Sensitivity</i> | | |
| SimPER (Xiao et al., 2025b) | $-\exp \left(\frac{1}{ y_w } \log \pi_\theta(y_w x) \right) + \exp \left(\frac{1}{ y_l } \log \pi_\theta(y_l x) \right)$ | - |
| RMiPO | $-\log \sigma \left(\frac{\beta}{ y_w } \log \pi_\theta(y_w x) - \frac{\beta}{ y_l } \log \pi_\theta(y_l x) - \gamma(x, y_w, y_l) \right)$ where $\gamma(x, y_w, y_l) = \gamma_{\min} + (\gamma_{\max} - \gamma_{\min}) \cdot \exp(-\max(0, \Delta_{\text{pmi}}))$ | $\beta = 2.0 \gamma_{\min} = 0.3 \gamma_{\max} = 1.6$ |

and the reference π_{ref} for a preference pair (y_w, y_l) . To enhance stability, $M(x, y_w, y_l)$ is Z-score normalized to obtain $M^*(x, y_w, y_l)$.

- **β -DPO (Wu et al., 2024)**: β -DPO proposes dynamically adjusting the hyperparameter β based on the quality of preference pairs to mitigate sensitivity and reduce tuning burden. Specifically, the quality of a preference pair (y_w, y_l) is measured by $M_i = \beta_0 \log \frac{\pi_\theta(y_w|x)}{\pi_{\text{ref}}(y_w|x)} - \beta_0 \log \frac{\pi_\theta(y_l|x)}{\pi_{\text{ref}}(y_l|x)}$. A scaling factor $\alpha \in [0, 1]$ is then introduced to modulate the influence of M_i on the update of β_i . To further enhance robustness, the method incorporates a data filtering mechanism controlled by the hyperparameter ρ , which suppresses the adverse effects of outlier preference pairs during β selection.
- **ϵ -DPO (Lee et al., 2025)**: ϵ -DPO examines the behavior of preference pairs under hyperparameter perturbations and proposes an adaptive mechanism for regulating β . Specifically, a perturbation scale $\epsilon > 0$ is introduced, defining the bounds $\beta_\epsilon^- := \frac{\beta}{1+\epsilon}, \beta_\epsilon^+ := \beta(1+\epsilon)$, which correspond to

values slightly below and above β , respectively. It has been observed that the preference likelihood of the policy model exhibits local monotonicity around β . Based on this property, ϵ -DPO dynamically adjusts β for each sample within the neighborhood determined by ϵ , thereby enhancing the robustness and adaptability of the optimization.

- **SimPER (Xiao et al., 2025b)**: SimPER demonstrates that optimizing only the inverse perplexity is sufficient for strong performance, where inverse perplexity is defined as the reciprocal of the exponentiated average log-likelihood of the chosen versus rejected responses in the preference dataset. Based on this, the method constructs a hyperparameter-free objective, offering advantages in computational and memory efficiency.

The above solutions for hyperparameter tuning have all demonstrated significant effectiveness. However, an analysis of their optimization objectives indicates that, except for SimPER, existing methods primarily alleviate the tuning burden by

reducing the number of tunable hyperparameters or narrowing the search space. In practical deployment, we observe that although SimPER has a simple structure and strong performance, its training stability is relatively limited. *Therefore, an ideal method should completely eliminate hyperparameter dependence while maintaining good stability.*

B Full Derivations

Derivation of Insight I. To analyze the impact of hyperparameters on offline preference optimization methods such as DPO and SimPO, we unify them under the following objective function framework: $\mathcal{L}(\theta) = -\mathbb{E}_{(x, y_w, y_l) \sim \mathcal{D}} [\log \sigma(\beta \Delta \log - \gamma)]$, where $\Delta \log := \log \pi_\theta(y_w | x) - \log \pi_\theta(y_l | x)$ denotes the log-likelihood difference of the policy model π_θ on a preference triplet (x, y_w, y_l) . Taking the gradient with respect to θ and applying the chain rule yields:

$$\nabla_\theta \mathcal{L} = -\mathbb{E} \left[\frac{1}{\sigma(\beta \Delta \log - \gamma)} \cdot \sigma(\beta \Delta \log - \gamma) \cdot (1 - \sigma(\beta \Delta \log - \gamma)) \cdot \nabla_\theta (\beta \Delta \log - \gamma) \right]. \quad (9)$$

Since γ is a constant independent of θ , we have $\nabla_\theta \gamma = 0$. After simplification, the gradient can be rewritten as:

$$\nabla_\theta \mathcal{L} = -\beta \cdot \mathbb{E} \left[\underbrace{(1 - \sigma(\beta \Delta \log - \gamma))}_{\text{Weight } w_i} \nabla_\theta \Delta \log \right] \quad (10)$$

Here, $w_i = 1 - \sigma(\beta \Delta \log - \gamma)$ can be interpreted as the effective weight of the i -th sample in the gradient update. Stationary points satisfy $\nabla_\theta \mathcal{L} = 0$. Because $\beta > 0$, this factor can be canceled in the first-order necessary condition, yielding a characterization of the stationary set \mathcal{S} :

$$\begin{aligned} & \mathbb{E} [w_i \nabla_\theta \Delta \log] \\ & = \mathbb{E} [(1 - \sigma(\beta \Delta \log - \gamma)) \nabla_\theta \Delta \log] = 0. \end{aligned} \quad (11)$$

To clarify the distinct roles of β and γ in convergence behavior, we examine the structure of the weight term w_i . Specifically, the symmetry center of the sigmoid function is determined by $\beta \Delta \log = \gamma$, at which point $w_i = 0.5$. This indicates that γ directly governs the relative importance of different samples in the expected gradient, thereby systematically shifting the stationary set \mathcal{S} . In contrast, β acts as a global scaling factor

on the gradient magnitude, controlling the steepness of the sigmoid function, i.e., the sharpness of the decision boundary. Although β substantially influences optimization dynamics, such as convergence speed and robustness to noise, it does not alter the sample-importance structure dominated by γ in the first-order stationary condition. *Thus, γ plays a more fundamental role than β in shaping the stationary points of the optimization.*

Derivation of Insight II. As discussed previously, the hyperparameter γ plays a central role in preference optimization objectives by regulating the importance of samples. However, if this modulation signal itself is influenced by bias in the prior distribution of responses, it may cause the model to overfit high-frequency yet weakly semantically related responses, thereby degrading alignment quality and generalization ability. Therefore, in the ideal case, there is an urgent need for a modulation signal that reflects purely semantic relevance and is insensitive to response popularity.

Current mainstream methods directly use the log-likelihood difference $\Delta \log$ of the policy model as the optimization signal. However, according to Bayes' theorem, the conditional probability $\pi_\theta(y|x)$ can be decomposed as $\pi_\theta(y|x) = \frac{\pi_\theta(x|y) \cdot \pi_\theta(y)}{\pi_\theta(x)}$. Taking the logarithm yields:

$$\log \pi_\theta(y|x) = \log \pi_\theta(x|y) + \log \pi_\theta(y) - \log \pi_\theta(x). \quad (12)$$

Correspondingly, the log-likelihood difference $\Delta \log$ for a preference pair (x, y_w, y_l) can be decomposed as:

$$\begin{aligned} \Delta \log &= \underbrace{(\log \pi_\theta(x|y_w) - \log \pi_\theta(x|y_l))}_{\text{Semantic Relevance Gap}} \\ &+ \underbrace{(\log \pi_\theta(y_w) - \log \pi_\theta(y_l))}_{\text{Prior Popularity Bias}}. \end{aligned} \quad (13)$$

Here, the term $\log \pi_\theta(y_l)$ characterizes the model's prior preference for response y , which is typically highly correlated with its frequency in the training data. The associated risk is that high-frequency responses with generic content are often excessively labeled as "preferred," even when their semantic association with the input is weak. If the optimization objective directly relies on $\Delta \log$, the model will tend to further amplify the likelihood of such popular responses, leading to degradation in semantic-input alignment and triggering downstream issues such as reduced generation diversity and weakened factual consistency (Li et al., 2024).

Algorithm 1 RMiPO: Response-level mutual information for preference optimization

Require: Preference dataset \mathbb{D} , Batch size b , Hyperparameters $\beta, \gamma_{\min}, \gamma_{\max}$, Learning rate η

```
1: Initialize policy parameters  $\pi_\theta$ 
2: while not converged do
3:   Sample a batch of preference pairs  $\{(x^{(i)}, y_w^{(i)}, y_l^{(i)})\}_{i=1}^b$  from  $\mathbb{D}$ 
4:   // Step 1: Compute the relevant probabilities based on the policy model.
5:   Compute the policy probabilities for the preferred response  $y_w$  and the less-preferred response  $y_l$ :
      $\pi_\theta(y_w|x), \pi_\theta(y_w), \pi_\theta(y_l|x), \pi_\theta(y_l)$ .
6:   Compute the modulation signal  $\Delta\text{pmi}$  using the formulation in Equation 7.
7:   // Step 2: Compute the length-normalized reward difference
8:    $\Delta\mathbb{R} = \frac{\beta}{|y_w|} \log \pi_\theta(y_w|x) - \frac{\beta}{|y_l|} \log \pi_\theta(y_l|x)$ 
9:   // Step 3: Compute the reward margin  $\gamma$  corresponding to each triplet.
10:   $\gamma(x, y_w, y_l) = \gamma_{\min} + (\gamma_{\max} - \gamma_{\min}) \cdot \exp(-\max(0, \Delta\text{pmi}))$ 
11:  // Step 4: Optimization Update
12:  Compute the loss  $\mathcal{L}_{\text{RMiPO}}$  using the formulation in Equation 8
13:  Compute the gradient and update the policy  $\theta_t \leftarrow \theta_{t-1} - \eta \nabla_\theta \mathcal{L}_{\text{RMiPO}}(\theta_{t-1})$ 
14: end while
15: return Optimized policy  $\pi_\theta$ 
```

To eliminate this bias, this paper introduces pointwise mutual information (PMI) as a decoupling tool. PMI is defined as $\text{PMI}(y|x) = \log \frac{\pi_\theta(y|x)}{\pi_\theta(y)}$. Clearly, PMI measures the additional information gain brought by response y given input x , and it naturally removes the prior probability $\pi_\theta(y)$ of the response itself. Based on this, we define the adaptive modulation signal for a preference pair as the PMI difference: $\Delta\text{pmi} = \text{PMI}(y_w|x) - \text{PMI}(y_l|x)$. Substituting the expression of PMI and simplifying yields:

$$\begin{aligned} \Delta\text{pmi} &= (\log \pi_\theta(x|y_w) - \log \pi_\theta(x)) \\ &\quad - (\log \pi_\theta(x|y_l) - \log \pi_\theta(x)) \quad (14) \\ &= \log \pi_\theta(x|y_w) - \log \pi_\theta(x|y_l). \end{aligned}$$

It follows that Δpmi depends solely on the semantic relevance gap, effectively eliminating the prior bias from $\pi_\theta(y)$. *An adaptive γ modulation based on Δpmi can improve generalization performance and generation quality in real-world scenarios.*

C Implementation Details

Dataset: In this work, the training of the policy model primarily relies on two preference datasets: UltraFeedback-Binarized (Cui et al., 2024) and Anthropic-HH (Bai et al., 2022).

- **UltraFeedback-Binarized:** The original UltraFeedback dataset contains 64,000 prompts,

each with four responses from various open- and closed-source language models. All responses are rated by GPT-4 based on overall quality (overall_score). In the UltraFeedback-Binarized version, a preference pair is formed by the highest-scoring response (the “preferred” response) and one randomly selected response from the remaining three (the “less-preferred” response). Policy models trained on this dataset are evaluated on instruction-following benchmarks and downstream tasks.

- **Anthropic-HH:** Anthropic-HH is a human-labeled dialogue preference dataset comprising two subsets, focusing on “helpfulness” and “harmlessness.” The dataset originates from manually constructed and red-teamed dialogue samples, designed to study how crowd-sourced annotators perform red-team attacks and which strategies are effective. Policy models trained on Anthropic-HH are primarily used for safety alignment evaluation.

Training: RMiPO is designed for simplicity and strong performance, with the full procedure provided in Algorithm 1. Its core lies in leveraging response-level PMI for hyperparameter adaptation. Specifically, $\pi_\theta(y)$ denotes the prior popularity of response y , defined as the joint probability of the sequence: $\pi_\theta(y) = \prod_{t=1}^{|y|} \pi_\theta(y_t | y_{<t})$. Unlike $\pi_\theta(y | x)$, this formulation removes dependence on

the prompt x , enabling explicit decoupling between response priors and semantic relevance.

For probability estimation, $\pi_\theta(y)$ is computed exactly over the full sequence without truncation or approximation, ensuring consistency with information-theoretic definitions. For datasets where prompts and responses are concatenated, we apply a lightweight string-matching heuristic to extract y . Although this may introduce minor boundary inaccuracies, it preserves intended decoupling. Consequently, Δ_{pmi} serves as an intrinsic measure of semantic separability, mitigating bias from non-semantic factors such as high-frequency tokens.

All models are trained under a unified setup with $\beta = 2.0$, $\gamma_{\min} = 0.3$, and $\gamma_{\max} = 1.6$, following the search space of SimPO. During SFT, we use a learning rate of 2×10^{-5} ; across both SFT and preference optimization, we adopt a batch size of 128, a maximum sequence length of 2048, and train for one epoch using Adam with cosine annealing. All experiments are conducted on four 80GB A800 GPUs. Baselines follow the same search protocol as SimPO: method-specific hyperparameters are detailed in Table 4, and learning rates are tuned within $[3 \times 10^{-7}, 1 \times 10^{-6}]$. In contrast, RMiPO (and SimPER) require tuning only the learning rate, without introducing additional hyperparameters.

Evaluation: Evaluation on AlpacaEval 2 adopts the same hyperparameter settings as SimPO: decoding temperatures are set to 0.7 and 0.5 for Mistral-Base and Mistral-Instruct, respectively, while both Llama-3 variants use a temperature of 0.9. Evaluation on MT-Bench follows the official decoding configuration, which specifies task-specific sampling temperatures. To systematically examine the impact of preference optimization methods on downstream task performance, we evaluate models trained with each method on the standard tasks included in the Hugging Face Open LLM Leaderboard, strictly following established evaluation protocols. For safety evaluation, we assess models trained on the Anthropic HH dataset using its test set as the benchmark, measuring the probability that the model generates preferred responses over less-preferred ones.

D Extended Experiments

Length-normalized Performance Analysis. In this work, length-normalized training is treated as an auxiliary technique for improving training stability rather than a core contribution. Although

Table 5: Without length-normalization on AlpacaEval2.

| Method | Mistral-7B-Instruct | | Llama3-8B-Instruct | | |
|---------------|---------------------|-------------|--------------------|-------------|-------------|
| | w/o LN | LC (%) | WR (%) | LC (%) | WR (%) |
| DPO | | 20.3 | 18.0 | 40.2 | 38.1 |
| SimPO | | 30.5 | 38.2 | 42.4 | 40.4 |
| α -DPO | | 32.0 | 38.4 | 44.4 | 42.6 |
| SimPER | | 28.5 | 34.6 | 41.7 | 40.2 |
| RMiPO | | 33.7 | 38.8 | 44.4 | 43.1 |

SimPO shows that length normalization improves model performance, particularly in terms of length-controlled win rate, we further examine the inherent stability and effectiveness of RMiPO without relying on this technique. To this end, we conduct experiments on RMiPO and several baselines with Llama-3-8B-Instruct and Mistral-7B-Instruct. As shown in Table 5, the results indicate that even without length normalization, RMiPO consistently yields improvements that are substantially stronger than those of the baselines.

Generated Response Case Analysis. To more intuitively demonstrate the performance advantages of RMiPO in practical generation tasks, we select a complex instruction involving “multicultural background and political viewpoint balance” from the UltraFeedback test set as an illustrative case. Table 6 presents a comparative illustration of outputs from the Mistral-7B-Base model optimized with RMiPO, SimPER, and SimPO, with results from GPT-4O-MINI provided for qualitative reference.

The analysis indicates that RMiPO exhibits significant advantages in terms of instruction-following and operational depth. While the prompt asks for a “comprehensive approach” to include diverse backgrounds, RMiPO is the only model that successfully translates this abstract meta-instruction into a concrete, executable prompt. It proposes a structured dialogue involving at least three distinct cultures and specifically identifies intersectional themes such as globalization, economic inequality, and environmental sustainability. This demonstrates a shift from simple “passive inclusion” to “active, high-fidelity cultural synthesis.” In contrast, SimPO and SimPER remain at a superficial or meta-discursive level: SimPO provides a high-level summary of the desired outcome but fails to generate a functional revised prompt, offering only a general goal of being “coherent and engaging.” SimPER focuses on the attitude and mindset of the writer, providing ethical guidelines rather than a structured response to the task itself. This

| | |
|--------------------|---|
| Prompt | To ensure a more comprehensive approach, the #Given Prompt# should now include a diverse range of cultural backgrounds and political perspectives. This will require thorough analysis and understanding of various societal norms and beliefs, emphasizing the criticality of cross-cultural communication and sensitivity towards different ideologies. |
| RMiPO | The revised prompt may read: "Create a dialogue within a multicultural community consisting of individuals from at least three distinct cultural backgrounds and varying political affiliations, discussing the challenges and opportunities presented by globalization in today's interconnected world, while addressing concerns related to economic inequality, cultural assimilation, communal identity, and environmental sustainability." |
| SimPER | It is crucial to approach this prompt with an open mind and a willingness to learn about perspectives that may differ from one's own. The end goal is to create a comprehensive and well-rounded essay that addresses the given prompt from multiple cultural and political perspectives. |
| SimPO | The resulting written piece should strive to accurately represent and explore multiple perspectives while remaining coherent and engaging to a broad audience. |
| GPT-4O-MINI | To ensure a more comprehensive approach, the #Given Prompt# should actively incorporate a diverse range of cultural backgrounds and political perspectives. This will involve: <ol style="list-style-type: none"> 1. Researching Cultural Norms: Understanding the foundational beliefs, values, and customs that shape various societies. This includes exploring different family structures, social hierarchies, and community practices that influence individual and collective behaviors. 2. Exploring Political Perspectives: Analyzing different political ideologies such as liberalism, conservatism, socialism, and others prevalent in specific regions. Acknowledging how these ideologies impact governance, social policies, and public opinion in various countries. 3. Cross-Cultural Communication: Emphasizing the importance of understanding and respecting cultural differences in communication styles. This includes recognizing the significance of non-verbal cues, context, and the role of language in shaping interactions across cultures. 4. Sensitivity Towards Ideologies: Being aware of and respecting differing beliefs and values, particularly when they conflict with one's own. This requires an open-minded approach, active listening, and a willingness to understand perspectives that may be contrary to personal views. 5. Diverse Representation: Ensuring that voices from various backgrounds are included in discussions, thereby enhancing the richness of the conversation. This can involve engaging with community leaders, cultural representatives, and activists from different political spectra. 6. Empathy and Understanding: Cultivating empathy by placing oneself in the shoes of individuals from various backgrounds. This helps in appreciating the complexities of their experiences and the factors that shape their viewpoints. 7. Critical Reflection: Encouraging self-reflection on one's biases and preconceived notions. This is crucial for fostering an inclusive dialogue that values all perspectives and contributions. <p>By integrating these elements, the #Given Prompt# will promote a fuller understanding of the multifaceted nature of human experience and the importance of fostering inclusive environments where diverse perspectives can coexist and enrich dialogue.</p> |

Table 6: Comparative analysis of response quality on the UltraFeedback test set. For identical prompts, the Mistral-7B-Base optimized with RMiPO produces significantly higher-quality and more helpful outputs than SOTA methods.

case clearly substantiates the quantitative results reported in Table 1: RMiPO not only outperforms state-of-the-art baselines in instruction-following win rates, but also enhances the model's reasoning

capability and structured expression ability when addressing high-difficulty, multi-dimensional tasks.

Adjusted γ Distribution Analysis. To investigate the adaptive alignment mechanism of RMiPO, this

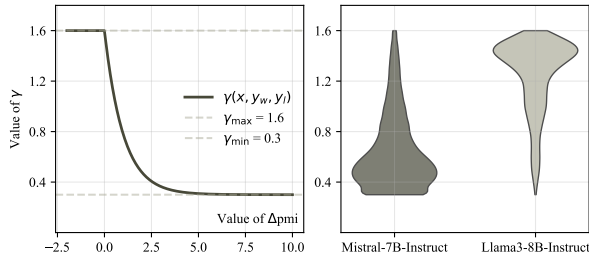


Figure 8: Adaptive γ Analysis. **Left:** The adaptive mechanism $\gamma(x, y_w, y_l)$ follows an exponential decay based on Δpmi , dynamically scaling the learning signal for each instance. **Right:** Empirical distribution of γ values on Mistral-7B and Llama3-8B. The density peaks (approximately 0.4 and 1.4, respectively) are approximately consistent with the optimal fixed values reported by SimPO, supporting the effectiveness of RMiPO’s automated, instance-level parameter optimization.

paper systematically analyzes the modulation coefficient γ from two perspectives—mechanism design and empirical distribution (see Figure 8).

First, at the level of mechanism design, RMiPO introduces a dynamic decay strategy based on pointwise mutual information differences between samples. As shown in the left subfigure of Figure 8, γ is defined as follows: $\gamma(x, y_w, y_l) = \gamma_{\min} + (\gamma_{\max} - \gamma_{\min}) \cdot \exp(-\max(0, \Delta\text{pmi}))$. This mechanism dynamically adjusts the value of γ according to the pointwise mutual information difference (Δpmi) between positive and negative samples. When the discriminability of a sample pair is low, γ remains at a high level ($\gamma_{\max} = 1.6$) to strengthen the learning signal; as the difference increases, γ smoothly decays toward the lower-bound value ($\gamma_{\min} = 0.3$) following an exponential decay pattern. This design enables fine-grained weight allocation across examples of varying difficulty. *We emphasize that this functional form serves as a practical instantiation of our principle, rather than the only valid implementation. While other monotonic decay functions could be used, we adopt the exponential form for superior smoothness and empirical stability.*

Second, the empirical distribution of γ observed during training validates the effectiveness and robustness of this adaptive mechanism. As shown in the right subfigure of Figure 8, upon completion of training, we compute the corresponding γ value for each sample in the dataset based on the final policy model. The results show that for Mistral-7B-Instruct, the density of γ is mainly concentrated around 0.4, whereas for Llama3-8B-Instruct, the primary distribution range is close to 1.4. Notably, these automatically emerging distribution centers

are closely aligned with the optimal fixed parameter values obtained by SimPO on the same models through extensive grid search.

These results demonstrate that RMiPO effectively replaces manual hyperparameter tuning with a principled adaptive parameterization mechanism. By transforming hyperparameters that traditionally require manual tuning into internal adaptive variables, RMiPO not only eliminates costly hyperparameter search, but also achieves more precise alignment through instance-level dynamic weight adjustment. This instance-level adaptation is a key reason why RMiPO consistently maintains strong performance across diverse model architectures.

Space and Time Complexity Analysis. To evaluate the practicality of RMiPO and baseline methods, we analyze space consumption (GPU memory) and time efficiency (computational throughput) under identical hardware settings, parallelization strategies, and model architectures.

Space complexity. GPU memory during training is dominated by model parameters, optimizer states, gradients, and activations. DPO-based methods (e.g., DPO, β -DPO, ϵ -DPO) require an additional frozen reference model π_{ref} to compute the KL constraint. Although it does not require optimizer states, its parameters still occupy substantial memory, leading to a peak utilization of 82%. In contrast, SimPO, SimPER, and RMiPO eliminate the reference model entirely. As a result, RMiPO reduces peak memory usage to 62% (an approximately 20% absolute reduction), enabling larger batch sizes or longer sequences and mitigating memory and communication bottlenecks in distributed training.

Time complexity. Per-step runtime is determined by the number of forward and backward passes. DPO-style methods incur additional latency due to an extra forward pass through π_{ref} , although this overhead is limited because no gradients are computed. SimPO and SimPER require only a single forward and backward pass, achieving the highest efficiency. In contrast, α -DPO introduces complex dynamic offsets and nonlinear gradient operations, resulting in the highest per-step latency. RMiPO has a runtime of $O(T) + \Delta$, where Δ arises from computing the PMI-based margin term $\gamma(x, y_w, y_l)$ with lightweight logit-level operations. Despite this small overhead, RMiPO converges faster and achieves better alignment.

Compared with DPO, which incurs an additional 30%–50% memory overhead, RMiPO maintains a lower memory footprint via its reference-free de-

sign while introducing only minimal computational overhead, yielding a favorable trade-off between efficiency, scalability, and alignment performance.

Mechanism analysis. This section provides additional theoretical clarification and empirical validation for the exponential scheduling mechanism and constraint bounds adopted in RMiPO. We aim to further substantiate that these design choices are grounded in principled considerations rather than heuristic tuning. Preference optimization is inherently conducted in the log-probability space. Therefore, an exponential mapping naturally aligns with the log-ratio structure of the objective function. This alignment ensures that the adaptive scaling remains geometrically consistent with the additive formulation in the log domain. More specifically, the exponential transformation preserves the relative ordering of preference signals while enabling controlled amplification of marginal differences. Compared with alternative mappings (e.g., linear or cosine), it provides a more balanced modulation behavior: in high-confidence regions, it avoids undermodulation and preserves discriminative gradients; in low-confidence regions, it suppresses excessive scaling and prevents noisy updates. As a result, the exponential schedule facilitates a smoother and more stable gradient flow throughout training. The parameters γ_{\min} and γ_{\max} are introduced as stability constraints rather than task-specific hyperparameters. Their primary role is to restrict the adaptive scaling within a safe region. These bounds serve two key purposes: preventing gradient explosion during early training stages or under noisy PMI estimation, and avoiding the dominance of marginal scaling over the main optimization objective. From a regularization perspective, the constraint interval enforces robustness across different random seeds and model architectures, ensuring consistent convergence behavior.

Ablation on Scheduling Strategies. We compare four scheduling strategies: *Exponential*, *Linear*, *Cosine*, and a baseline without explicit scheduling (*w/o*). Experiments are conducted on the AlpacaEval 2 benchmark under the length-control (LC) setting. The results are summarized in Table 7. The exponential mapping consistently outperforms alternative strategies across both models. This suggests that the nonlinear curvature of the exponential mapping is more effective at capturing subtle preference distinctions in complex semantic comparisons, thereby supporting our geometric-consistency hypothesis.

| Model | Exponential | Linear | Cosine | w/o |
|---------------------|-------------|--------|--------|------|
| Mistral-7B-Instruct | 38.1 | 36.9 | 35.5 | 37.2 |
| Llama-8B-Instruct | 49.1 | 47.9 | 46.5 | 47.2 |

Table 7: Win rates (%) under different scheduling strategies on AlpacaEval 2 (LC setting).

Sensitivity Analysis on Constraint Bounds. We further evaluate the sensitivity of RMiPO to different choices of $(\gamma_{\min}, \gamma_{\max})$. The results are summarized in Table 8. The performance remains stable across a wide range of boundary settings, with only minor fluctuations. This robustness suggests that the bounds primarily function as safety constraints rather than sensitive tuning parameters.

| Model | [0, 2] | [0.3, 1.6] | [0.5, 1.4] | [1.0, 1.2] |
|---------------------|--------|------------|------------|------------|
| Mistral-7B-Instruct | 38.5 | 38.1 | 37.4 | 36.9 |
| Llama-8B-Instruct | 50.4 | 49.1 | 48.3 | 47.5 |

Table 8: Sensitivity analysis of constraint bounds.

Overall, both theoretical analysis and empirical results demonstrate that: (i) the exponential scheduling mechanism is well-aligned with the log-space structure of preference optimization; and (ii) constraint bounds act as regularization safeguards, ensuring stable and robust training dynamics. These findings confirm that the proposed design is principled and generalizable, rather than relying on heuristic parameter choices.