

IRPO: Implicit Policy Regularized Preference Optimization

Youngsoo Jang^{1,*}, Yu Jin Kim², Geon-Hyeong Kim², Honglak Lee², Moontae Lee^{2,3}

¹UNIST, ²LG AI Research, ³University of Illinois Chicago

youngsoo.jang@unist.ac.kr

{yujin.kim, ghkim, honglak, moontae.lee}@lgresearch.ai

Abstract

Training complexity often scales with the size of hyperparameter space for Large Language Models (LLMs). While Direct Preference Optimization (DPO) offers learning stability through reparameterizing the reward function, its regularization against the reference policy can lead to suboptimal outcomes when the reference policy is not optimal. Recent DPO variants address this concern, but at a cost: they introduce additional hyperparameters, reducing feasibility for LLM fine-tuning. To overcome this challenge, we introduce Implicit policy Regularized Preference Optimization (IRPO), which tackles suboptimality while maintaining training simplicity. By treating the winning policy that generated the chosen responses in a pairwise dataset as an implicit policy, IRPO maximizes KL-regularized reward without extra hyperparameters. Then we propose a novel PO algorithm that directly optimizes the IRPO objective by estimating the likelihood ratio between implicit policies. As the winning policy generally outperforms the reference policy, IRPO can effectively address suboptimality. Our experiments show that IRPO significantly outperforms baseline algorithms with the same hyperparameter complexity. Moreover, IRPO demonstrates comparable performance to recent algorithms that rely on a larger number of hyperparameters, offering a practical solution for scalable LLM fine-tuning.

1 Introduction

Aligning Large Language Models (LLMs) with human preferences is crucial for improving their core capabilities to generate helpful, safe, and logical responses. Learning from human feedback has served a key approach for achieving this alignment (Pang et al., 2024a; Lai et al., 2024). Reinforcement learning from human feedback (RLHF) (Ouyang

et al., 2022) has been widely used for fine-tuning LLMs, which involves training a reward model and then optimizing the policy based on the trained reward model. However, optimizing LLMs against a trained reward model through RL poses significant challenges due to learning instability and non-trivial computational costs (Gao et al., 2022; Ramamurthy et al., 2023).

Direct Alignment Algorithms (DAAs) have become an effective approach for fine-tuning LLMs by optimizing policies directly from preference datasets without separately training a reward model (Rafailov et al., 2024; Gheshlaghi Azar et al., 2024; Zhao et al., 2023). Among these DAAs, Direct Preference Optimization (DPO) (Rafailov et al., 2024) is the most widely used and has shown strong performance on various NLP tasks, such as text summarization, question-answering, and mathematical reasoning (Pang et al., 2024b; Lai et al., 2024). Despite its advantages, DPO exhibits inherent limitations, including suboptimal performance due to regularization against the reference policy (Meng et al., 2024; Hong et al., 2024; Wu et al., 2024a) and susceptibility to length exploitation (Park et al., 2024; Meng et al., 2024).

Addressing suboptimality must stay within practical training complexity. Recently, several variations of DPO have been proposed to overcome the limitations of DPO (Meng et al., 2024; Hong et al., 2024; Wu et al., 2024a,b; Park et al., 2024; Yuan et al., 2023; Xu et al., 2024). Among them, SimPO (Meng et al., 2024) refines DPO’s learning objective with a reference-free length-normalized reward, then introducing a target reward margin to ensure a sufficient gap between winning and losing responses. While effective, SimPO adds an extra hyperparameter γ for the target reward margin, increasing training complexity and performance sensitivity over DPO. Similar to SimPO, most DPO variants enhance alignment performance by introducing additional hyperparameters, which

*Work done while the author was a research scientist at LG AI Research.

causes hyperparameter-sensitive behavior and significantly higher computational costs for tuning.

This paper introduces **Implicit policy Regularized Preference Optimization (IRPO)**, which addresses the suboptimality of DPO while maintaining the same training complexity without requiring any additional hyperparameters. Unlike DPO, which regularizes based on an explicitly given reference policy, IRPO formulates a different regularization by leveraging the winning policy — an implicit policy that generally outperforms the reference policy. Then we propose a novel preference optimization algorithm that directly optimizes this objective by estimating the likelihood ratio between implicit policies.

Note that IRPO bypasses the need for an explicit reference model and directly performs preference optimization using only a pairwise preference dataset. Avoiding the challenge of selecting a reference policy, IRPO contrasts recent DPO variants that improve performance at the cost of added hyperparameters. Our experiments on extensive chat-based evaluation benchmarks, including AlpacaEval 2 (Li et al., 2023; Dubois et al., 2024) and Arena-Hard (Li et al., 2024), show that IRPO significantly outperforms baseline algorithms with equivalent hyperparameter complexity by up to 15.4 points on AlpacaEval 2 and 4.6 points on Arena-Hard. Furthermore, IRPO achieves comparable performance to algorithms that require a greater number of hyperparameters, offering a more practical and optimal solution.

2 Notations and Preliminaries

In this section, we formally introduce the foundations of preference optimization for Large Language Models (LLMs).

2.1 Problem Setup

We consider the offline preference optimization problem, where the policy should be optimized from a pre-collected pairwise dataset. In this problem, we assume a pairwise dataset $\mathcal{D} = \{(x, y_w, y_l)_i\}_{i=1}^N$ comprising prompts x and labeled response pairs (y_w, y_l) , where y_w and y_l denote winning and losing responses, respectively. We denote the dataset consisting of winning and losing responses as $\mathcal{D}_w = \{(x, y_w)_i\}_{i=1}^N$ and $\mathcal{D}_l = \{(x, y_l)_i\}_{i=1}^N$, respectively, and the winning and losing policies that generated these responses as π_w and π_l . Here, π_w and π_l are underlying poli-

cies that are not explicitly given. Given an offline dataset $\mathcal{D} = \mathcal{D}_w \cup \mathcal{D}_l$, our goal is to optimize the LLM policy to align human preference.

2.2 Reinforcement Learning from Human Feedback

RLHF (Ouyang et al., 2022) is a classic method for fine-tuning LLMs to achieve preference alignment, consisting of the following two processes: 1) reward modeling and 2) RL fine-tuning.

Reward Modeling. Human preferences are assumed to be labeled by an underlying latent reward model $r^*(x, y)$, to which we have no access. The Bradley-Terry (BT) model (Bradley and Terry, 1952) is most popular choice for modeling pairwise preferences:

$$p^*(y_w \succ y_l | x) = \frac{\exp(r^*(x, y_w))}{\exp(r^*(x, y_w)) + \exp(r^*(x, y_l))} = \sigma(r^*(x, y_w) - r^*(x, y_l)),$$

where $\sigma(x) = 1/(1 + \exp(-x))$ is the logistic function. Given an offline preference labels sampled from p^* , the parameterized reward model $r_\phi(x, y)$ can be estimated by maximizing the log-likelihood of the preference feedback:

$$-\mathbb{E}_{(x, y_w, y_l) \sim \mathcal{D}} [\log \sigma(r_\phi(x, y_w) - r_\phi(x, y_l))]. \quad (1)$$

RL Fine-Tuning. For the RL fine-tuning phase, RLHF methods optimize the policy π_θ to maximize the following KL-regularized reward objective:

$$\max_{\pi_\theta} \mathbb{E}_{\pi_\theta} [r_\phi(x, y)] - \beta \mathbb{D}_{\text{KL}}[\pi_\theta(y|x) || \pi_{\text{ref}}(y|x)], \quad (2)$$

where $r_\phi(x, y)$ is the reward function learned using the BT model on the preference dataset, π_{ref} is the reference policy, and $\beta > 0$ is a tunable hyperparameter that controls the deviation from the reference policy. For brevity, we will abuse the notation $\mathbb{E}_{\pi_\theta}[\cdot]$ to represent $\mathbb{E}_{x \sim \mathcal{D}, y \sim \pi_\theta(y|x)}[\cdot]$ unless ambiguous. This objective is typically optimized by RL algorithms, such as PPO (Schulman et al., 2017). Although RLHF approaches have achieved remarkable success, training with RLHF is still challenging due to learning instability and high computational requirements.

2.3 Direct Preference Optimization

DPO (Rafailov et al., 2024) is a standard preference optimization method that addresses the learning instability and complexity of RLHF. Instead of learning an explicit reward model (Ouyang et al., 2022),

DPO reparameterizes the reward function $r(x, y)$ using a closed-form expression with the optimal policy that satisfies Eq. (2):

$$r(x, y) = \beta \log \frac{\pi_\theta(y|x)}{\pi_{\text{ref}}(y|x)} + \beta \log Z(x),$$

where $Z(x) = \sum_y \pi_{\text{ref}}(y|x) \exp(\frac{1}{\beta} r(x, y))$ is the partition function. By incorporating this reward formulation into the BT model (i.e., Eq. (1)), it can be converted to the DPO loss for pairwise preference dataset \mathcal{D} :

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

where $\sigma(\cdot)$ denotes the logistic function. While DPO offers learning stability through reparameterizing the reward function, its regularization against the reference policy can lead to suboptimal outcomes when the reference policy is not optimal.

2.4 Reference-free Preference Optimization

Recent studies have explored reference-free preference optimization methods (Meng et al., 2024; Hong et al., 2024; Yuan et al., 2023; Xu et al., 2024) that do not consider regularization with the reference policy to address the problem of suboptimality of DPO. SimPO (Meng et al., 2024) is the state-of-the-art algorithm, which eliminates the need for a reference policy. SimPO optimizes the length-normalized log-probability of response pairs with a target reward margin based on the BT model:

$$\begin{aligned} \mathcal{L}_{\text{SimPO}}(\pi_\theta) \\ = -\mathbb{E}_{\mathcal{D}} \left[\log \sigma \left(\frac{\beta}{|y_w|} \log \pi_\theta(y_w|x) - \frac{\beta}{|y_l|} \log \pi_\theta(y_l|x) - \gamma \right) \right], \end{aligned} \quad (3)$$

where $|y|$ denotes the length of the response y , normalizing the reward by the response lengths, β is a hyperparameter that controls the scaling of the reward difference, and γ is an additional hyperparameter for the target reward margin. Optimizing this objective requires extensive hyperparameter tuning for β and γ , which play a crucial role in alignment performance.

3 Implicit Policy Regularized Preference Optimization

In this section, we present **Implicit policy Regularized Preference Optimization (IRPO)**, which addresses the suboptimality issue with the equivalent hyperparameter complexity to DPO, without requiring any additional hyperparameters.

3.1 Preference Optimization with Implicit Policy Regularization

Unlike DPO, which considers KL-regularization against the reference policy, our derivation starts from a different KL-regularized reward maximization objective by leveraging the winning policy as follows:

$$\max_{\pi_\theta} \mathbb{E}_{\pi_\theta} [r'(x, y)] - \beta \mathbb{D}_{\text{KL}}[\pi_\theta(y|x) || \pi_w(y|x)], \quad (4)$$

where $r(x, y)$ is underlying reward function for human preferences and $r'(x, y) := r(x, y) - \beta \log \pi_l(y|x)$ is reshaped reward by penalizing log-probability of losing policy. Intuitively, this reward penalty encourages the learned policy to deviate from the losing policy by providing a lower reward if the log-likelihood of the losing policy is high. We also provide theoretical analyses of the KL-regularization with respect to the winning policy and the reward shaping in our objective Eq. (4) in Appendix B.

Following prior work (Rafailov et al., 2024; Peng et al., 2019; Peters and Schaal, 2007), the optimal solution to the KL-regularized reward maximization objective in Eq. (4) takes the form:

$$\pi_{r'}(y|x) = \frac{1}{Z_{r'}(x)} \pi_w(y|x) \exp \left(\frac{1}{\beta} r'(x, y) \right), \quad (5)$$

where $Z_{r'}(x) = \sum_y \pi_w(y|x) \exp(\frac{1}{\beta} r'(x, y))$ is the partition function. By rearranging Eq. (5), we obtain the formulation of shaped reward $r'(x, y)$ as follows:

$$r'(x, y) = \beta \log \frac{\pi_{r'}(y|x)}{\pi_w(y|x)} + \beta \log Z(x).$$

Since the given preference dataset \mathcal{D} is labeled with the underlying reward function $r(x, y)$, we need to rearrange it to $r(x, y)$ by plugging $r'(x, y) = r(x, y) - \beta \log \pi_l(y|x)$ into the equation:

$$r(x, y) = \beta \log \left(\frac{\pi_{r'}(y|x)}{\pi_w(y|x)} \cdot \pi_l(y|x) \right) + \beta \log Z(x).$$

Then, by incorporating this reward formulation into the BT model (i.e., Eq. (1)), it can be converted to the following objective for pairwise preference dataset \mathcal{D} :

$\gamma'_{\text{LN}}(x, y_w, y_l)$ in IRPO. Additionally, the following objective of SimPO without considering length normalization also exactly matches the objective of IRPO (Eq. (10)):

$$\begin{aligned} & \mathcal{L}_{\text{SimPO w/o LN}}(\pi_\theta) \\ &= -\mathbb{E}_{\mathcal{D}} \left[\log \sigma \left(\beta \log \pi_\theta(y_w|x) - \beta \log \pi_\theta(y_l|x) - \gamma \right) \right]. \end{aligned} \quad (12)$$

The assumption of a fixed target reward margin in SimPO may lead to suboptimal performance, and addressing this issue through the hyperparameter search is challenging due to significant training costs and hyperparameter-sensitive results. As shown in Figure 1, the results of SimPO are suboptimal and highly sensitive to the hyperparameter γ . In contrast, IRPO eliminates the cost of adjusting the hyperparameters for the reward margin by utilizing a discriminator-based dynamic reward margin and achieves performance comparable to or better than SimPO, even without additional hyperparameters. As a result, IRPO addresses the suboptimality issue with the same hyperparameter complexity as DPO without introducing extra hyperparameters.

4 Experiments

In this section, we present the main results of our experiments on widely used instruction-following benchmarks (AlpacaEval 2 (Dubois et al., 2024) and Arena-Hard (Li et al., 2024)), highlighting the comparable performance of IRPO without additional hyperparameters over existing methods (Section 4.2). We provide additional experimental results by comparing IRPO with SimPO with various hyperparameter values of γ to demonstrate the effectiveness and efficiency of IRPO (Section 4.3). Furthermore, we also evaluate our model on downstream tasks from the ZeroEval benchmark (Lin, 2024) to evaluate the degree of performance degradation (Section 4.4).

4.1 Experimental setup

Models. We performed preference alignment on the following two families of open-source models: Llama3-8B (Instruct) (Grattafiori et al., 2024) and Mistral-7B (Base and Instruct) (Jiang et al., 2023). For the initial model to train the discriminator, we used the same model as the initial model of the policy (Mistral-7B Base/Instruct for Mistral-7B Base/Instruct) or a smaller model from the same family (Llama3.2-1B Instruct for Llama3-8B Instruct/Instruct-v0.2). To demonstrate that IRPO

is not significantly affected by the training process of the discriminator, we employed two different training strategies. In the Llama-based model setting, we jointly optimize both the discriminator and policy, and we pre-train the discriminator and then optimize the policy in the Mistral-based model setting.

Datasets. For a fair comparison, we follow the same training settings as in SimPO (Meng et al., 2024). For Llama3-8B-Instruct setups, we optimize the policy with the following two different datasets: [princetonmlp/llama3-ultrafeedback](#) for Llama3-Instruct (8B) and [princetonmlp/llama3-ultrafeedback-armorm](#) for Llama3-Instruct-v0.2 (8B). For Mistral-7B-Base setups, we perform preference optimization on the UltraFeedback dataset (Cui et al., 2024) with an SFT model trained on the UltraChat-200k dataset (Ding et al., 2023). For Mistral-7B-Instruct setups, we perform preference optimization on the [princeton-mlp/mistral-instruct-ultrafeedback](#) dataset.

Evaluation Benchmarks. Following the prior work (Meng et al., 2024; Wu et al., 2024a; Xiao et al., 2025), we evaluate our algorithm and baselines on AlpacaEval 2 (Dubois et al., 2024) and Arena-Hard (Li et al., 2024), which are widely used chat-based evaluation benchmarks for LLM evaluation. For AlpacaEval 2, we report the win rate (WR) and length-controlled win rate (LC) against the baseline model (GPT-4 Turbo). For Arena-Hard, we provide the win rate (WR), length-controlled win rate (LC), and style-controlled win rate (SC) against the baseline model (GPT-4-0314). For both benchmarks, we use GPT-4 Turbo as the judge model for pairwise evaluation. To evaluate the degree of performance degradation, we evaluate our algorithm on various downstream tasks from the ZeroEval benchmark (Lin, 2024), including MMLU-Redux (Gema et al., 2025), GSM8K (Cobbe et al., 2021), MATH (Hendrycks et al., 2021), CRUXEval (Gu et al., 2024), and ZebraLogic (Lin et al., 2024).

Baselines. We compare our algorithm IRPO with the following recent preference optimization methods: DPO (Rafailov et al., 2024), SimPO (Meng et al., 2024), α -DPO (Wu et al., 2024a), SIMPER (Xiao et al., 2025), SLiC-HF (Zhao et al., 2023), RRHF (Yuan et al., 2023), IPO (Gheshlaghi Azar et al., 2024), CPO (Xu et al., 2024), KTO (Ethayarajh et al., 2024), ORPO (Hong et al.,

Method	Llama3-Instruct (8B)					Llama3-Instruct-v0.2 (8B)					# Hyper-parameters	w/o Reference Model
	AlpacaEval 2		Arena-Hard			AlpacaEval 2		Arena-Hard				
	LC (%)	WR (%)	SC (%)	LC (%)	WR (%)	LC (%)	WR (%)	SC (%)	LC (%)	WR (%)		
SFT	22.9	22.6	19.9	20.1	20.6	22.9	22.6	19.9	20.1	20.6	-	-
SimPER	37.2	35.5	29.1	30.8	30.8	41.8	37.8	30.0	29.1	32.9	0	✓
ORPO	27.0	26.4	26.2	27.0	27.3	35.7	32.3	27.6	30.1	29.6	1	✓
RRHF	30.3	28.0	26.9	27.8	27.2	32.6	28.7	28.0	31.1	29.5	1	✓
IPO	35.9	35.5	30.0	30.6	30.9	48.0	43.5	31.2	34.7	34.3	1	✗
DPO	39.0	37.4	31.7	32.4	32.8	46.6	46.0	<u>30.9</u>	35.5	36.0	1	✗
SLiC-HF	27.9	28.3	24.1	24.8	25.7	32.6	28.7	28.1	31.1	29.6	2	✓
CPO	28.1	31.4	25.8	27.2	29.0	32.7	35.1	27.4	29.3	30.6	2	✓
R-DPO	38.8	36.9	34.0	34.8	<u>34.9</u>	46.2	44.8	28.2	32.4	32.7	2	✗
SimPO	43.5	<u>38.6</u>	<u>32.7</u>	<u>34.4</u>	33.8	50.7	45.7	31.2	36.7	<u>37.0</u>	2	✓
KTO	31.0	30.4	25.1	26.3	26.8	32.0	30.4	26.4	28.1	28.1	3	✗
α -DPO	40.2	33.9	23.0	24.6	23.1	52.4	<u>48.5</u>	25.4	31.0	30.7	3	✗
IRPO	40.0	45.3	29.1	32.7	35.7	48.3	51.2	29.7	34.4	37.9	1	✓
IRPO w/ LN	<u>41.3</u>	35.2	27.6	26.1	28.3	<u>52.3</u>	44.0	<u>30.9</u>	<u>36.0</u>	33.9	1	✓

Method	Mistral-Base (7B)					Mistral-Instruct (7B)					# Hyper-parameters	w/o Reference Model
	AlpacaEval 2		Arena-Hard			AlpacaEval 2		Arena-Hard				
	LC (%)	WR (%)	SC (%)	LC (%)	WR (%)	LC (%)	WR (%)	SC (%)	LC (%)	WR (%)		
SFT	7.5	4.5	3.9	2.7	2.2	18.0	15.1	16.2	12.6	12.6	-	-
SimPER	13.2	10.3	11.6	11.3	8.5	24.5	26.8	20.1	<u>21.5</u>	17.6	0	✓
ORPO	15.1	12.8	11.3	8.9	8.6	24.2	23.9	<u>25.1</u>	20.6	21.1	1	✓
RRHF	9.3	8.4	8.6	6.6	6.7	26.0	24.1	25.8	21.2	21.8	1	✓
IPO	11.5	9.5	9.9	7.4	7.8	19.8	19.0	20.7	16.8	17.0	1	✗
DPO	15.1	13.6	11.5	8.7	9.3	25.4	22.7	23.1	19.0	19.1	1	✗
SLiC-HF	11.2	8.9	8.8	6.8	7.0	23.1	22.4	21.9	17.8	18.8	2	✓
CPO	11.3	10.1	7.2	5.4	6.1	25.1	30.5	19.6	16.5	20.3	2	✓
R-DPO	<u>17.8</u>	13.6	13.9	10.7	10.8	25.2	21.7	22.4	17.9	18.0	2	✗
SimPO	21.8	21.6	18.9	14.9	16.0	32.8	33.8	24.8	20.5	20.8	2	✓
KTO	12.3	8.5	8.4	6.4	5.6	24.4	22.7	24.6	19.9	19.8	3	✗
α -DPO	17.1	<u>16.5</u>	<u>18.3</u>	<u>14.4</u>	<u>15.0</u>	<u>32.4</u>	<u>34.2</u>	21.5	18.6	19.1	3	✗
IRPO	12.9	13.7	11.4	9.7	10.8	28.1	38.1	18.9	16.1	<u>21.7</u>	1	✓
IRPO w/ LN	<u>17.8</u>	15.6	17.2	13.9	13.9	28.0	29.4	22.1	22.9	19.1	1	✓

Table 1: **Best-tuned results** for AlpacaEval 2 and Arena-Hard in four settings with different models and training datasets. The results are reported as evaluation metrics for the win rate (**WR**), length-controlled win rate (**LC**), and style-controlled win rate (**SC**). **# Hyperparameters** denotes the number of hyperparameters that each algorithm requires additional tuning, except for the learning rate (i.e., denotes the complexity of the hyperparameter tuning). The best results are highlighted in bold and the second-best results are underlined. The details of the learning objectives and hyperparameters of each algorithm can be found in Appendix A. The results highlight that IRPO 1) **avoids the challenge of selecting a reference policy**, 2) **achieves significant performance improvement** with equivalent hyperparameter complexity to DPO, and 3) **demonstrates comparable performance to recent algorithms that rely on a larger number of hyperparameters without adding any hyperparameters**.

2024), and R-DPO (Park et al., 2024). The baseline algorithms we consider require varying levels of complexity of hyperparameter tuning, and all results are thoroughly tuned for all hyperparameters for each baseline *without considering fairness of cost*. Therefore, we explicitly specify the number of hyperparameters (**# Hyperparameters** in Table 1) required for each algorithm along with the results to distinguish the complexity of hyperparameter tuning. More details of baselines and the hyperparameter search space can be found in Appendix A.

4.2 Main Results

Table 1 summarizes the performance of our algorithm and the baseline methods on the AlpacaE-

val 2 and Arena-Hard benchmarks. The detailed results, including average generation length, standard error, and confidence interval, can be found in Appendix A.4. We evaluated both versions of IRPO with and without the length-normalized reward formulation and denoted them as IRPO w/ LN (Eq. (11)) and IRPO (Eq. (10)), respectively. As shown in Table 1, IRPO avoids the challenge of selecting a reference policy, which can lead to suboptimality. In terms of performance, IRPO consistently outperforms baselines with lower or equivalent hyperparameter tuning complexity (i.e., ORPO, RRHF, IPO, and DPO) for all experimental settings. Furthermore, IRPO achieves comparable performance even compared to baselines that require higher hyperparameter tuning complexity

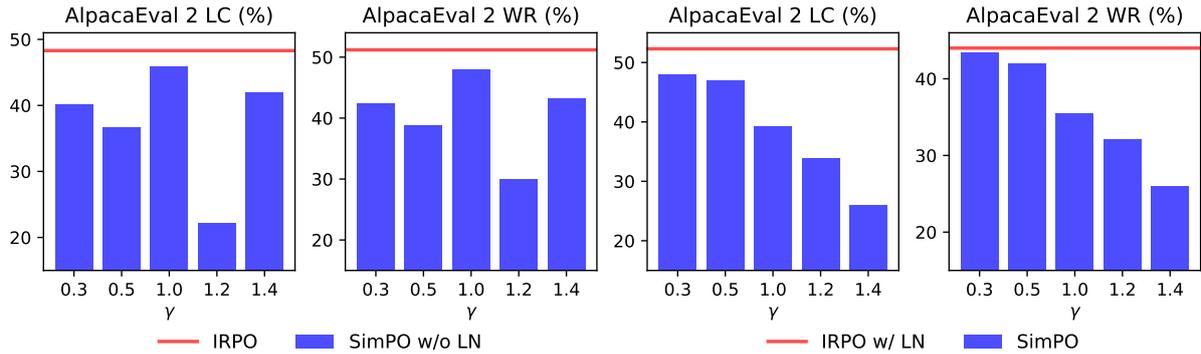


Figure 1: **Sensitivity analysis** on the AlpacaEval 2 of IRPO and SimPO with varying hyperparameter γ . All results are from training the Llama3-Instruct (8B) model using the [princetonnlpllama3-ultrafeedback](#) (i.e., same experimental settings of Llama3-Instruct-v0.2 (8B) in Table 1), with the fixed value of hyperparameter β for both IRPO and SimPO ($\beta = 2.5$). The two results on the left were obtained without considering length normalization (i.e., Eq. (10) for IRPO and Eq. (12) for SimPO w/o LN), while the two results on the right were obtained with objectives that incorporate length normalization (i.e., Eq. (11) for IRPO w/ LN and Eq. (3) for SimPO). The results are reported as evaluation metrics for the win rate (WR), and length-controlled win rate (LC). The results demonstrate that the performance of both SimPO w/o LN and SimPO are highly sensitive to additional hyperparameter γ .

(i.e., SLiC-HF, CPO, R-DPO, SimPO, KTO, and α -DPO). As a result, we experimentally verified that we can enhance the performance of preference optimization with equivalent hyperparameter complexity to DPO without using huge training costs by tuning additional hyperparameters. Furthermore, we provide detailed results, including average generation length, in Appendix A.4, and comparing IRPO and IRPO w/ LN results shows that the length-normalized reward formulation effectively prevents length exploitation. We also provide qualitative analysis with examples generated by IRPO, IRPO w/ LN, and SimPO in Appendix C. As can be seen in Table 7 and 8, both IRPO and IRPO w/ LN generate structured responses. However, due to the effect of length normalization, IRPO w/ LN generates relatively concise responses, whereas IRPO generates longer but more detailed responses.

4.3 Comparison IRPO with SimPO.

As described in Section 3.4, the objectives of IRPO and IRPO w/ LN align most closely with those of SimPO w/o LN and SimPO, respectively. Instead of extensive tuning of the fixed reward margin γ as a hyperparameter in SimPO, it can be seen that it is replaced by learnable dynamic reward margin $\gamma'(x, y_w, y_l)$, which is estimated directly based on the discriminator. Therefore, we provide additional experimental results to demonstrate how the learnable reward margin in IRPO, which is automatically determined by discriminator training, is more effective compared to using a fixed reward margin with various values of the hyperparameter

γ . Figure 1 demonstrates the effects of the trained dynamic reward margin (i.e., IRPO and IRPO w/ LN) and the fixed reward margin (i.e., SimPO and SimPO w/o LN) with various hyperparameter values of γ . As shown in Figure 1, SimPO shows highly sensitive performances depending on the hyperparameter γ , which can be critical for the fine-tuning of LLM, where training costs are extremely expensive. Moreover, while SimPO’s fixed reward margin leads to suboptimal performance, IRPO’s dynamic reward margin achieves better performance without additional hyperparameters.

4.4 Downstream Task Evaluation

In this section, we also evaluate IRPO’s performance on downstream tasks to measure the degree of performance degradation for other benchmarks. We compare the degradation in the performance of IRPO and baselines (DPO, SimPO, α -DPO) compared to the initial performance of the base model. Table 2 summarizes the overall results on various downstream tasks. Although both IRPO and the baseline algorithms show performance degradation on other tasks due to training, IRPO shows similar or less degradation than baselines. However, this performance degradation issue is a limitation of IRPO and needs to be addressed in future work.

4.5 Training Efficiency Evaluation

Additionally, we measured the total time required for training in the same computing environment to show how significantly additional hyperparameters affect training efficiency. Table 3 demonstrates the training efficiency of IRPO compared to baseline

Model	MMLU-Redux	GSM8K	MATH	CRUXEval	ZebraLogic
Base Model	62.4	78.1	6.9	38.8	10.9
DPO	58.1	51.2	5.1	32.3	7.2
SimPO	54.6	56.6	6.4	32.4	3.3
α -DPO	37.0	53.2	5.8	31.3	4.1
IRPO	60.1	54.6	5.1	33.3	5.4
IRPO w/ LN	58.8	56.4	5.8	32.3	4.1

Table 2: Results on downstream tasks from ZeroEval benchmark including MMLU-Redux, GSM8K, MATH, CRUXEval, ZebraLogic when continuing training with SimPO and IRPO from Llama-3-8B-Instruct (i.e., Base Model).

algorithms (DPO, SimPO, and α -DPO) with various hyperparameter complexities. IRPO requires a little more run time for discriminator training than other algorithms, but by eliminating the additional hyperparameters (γ in SimPO, and $\{\gamma, \alpha\}$ in α -DPO), the cost of hyperparameter tuning can be dramatically reduced. As can be seen in Figure 1, SimPO shows highly sensitive results to the hyperparameter γ , which dramatically increases the total cost of learning. In particular, optimal hyperparameter values vary depending on the training dataset and model, necessitating a broader range of hyperparameter searches. Therefore, the complexity of hyperparameters significantly affects training costs and may lead to LLM fine-tuning not being scalable.

Model	Hyperparameters	Runtime (h)	Total time (h)
DPO	$\beta \in [0.01, 0.05, 0.1]$	2.0	6.0
SimPO	$\beta \in [2.0, 2.5, 10.0]$ $\gamma \in [0.3, 0.5, 1.0, 1.2, 1.4, 1.6]$	1.5	27.0
α -DPO	$\beta \in [2.5, 10.0]$ $\gamma \in [0.1, 0.3, 0.5]$ $\alpha \in [0.01, 0.05, 0.1, 0.2]$	2.0	48.0
IRPO	$\beta \in [0.01, 0.05, 0.1]$	3.7	11.0
IRPO w/ LN	$\beta \in [2.0, 2.5, 10.0]$	3.7	11.0

Table 3: Runtime and required hyperparameters for DPO, SimPO, α -DPO, and IRPO. Runtime is measured with $16 \times$ NVIDIA A100 (40GB) GPUs for the experimental setting of Llama3-Instruct-v0.2 (8B) in Table 1. Total time is calculated by multiplying the runtime by the number of hyperparameter tuning.

5 Related Work

Reinforcement Learning from Human Feedback. RLHF is a standard technique that aligns large language models with human preferences (Christiano et al., 2023; Ouyang et al., 2022). The training process of RLHF typically comprises three phases: supervised fine-tuning (Zhou et al., 2023; Köpf et al., 2023), reward modeling (Gao et al., 2022; Luo et al., 2025), and RL fine-tuning (Schulman et al., 2017; DeepSeek-AI et al., 2025). For the last phase, Proximal Policy Optimization

(PPO) (Schulman et al., 2017) is a widely used algorithm that successfully improves large language models in various domains (Ouyang et al., 2022; Bai et al., 2022; Havrilla et al., 2024). Recently, Shao et al. (2024) introduced Group Relative Policy Optimization (GRPO), a variant of PPO, that foregoes the critic model, instead estimating the baseline from group scores, significantly reducing training resources compared to PPO.

Direct Preference Optimization. Direct Preference Optimization (DPO) (Rafailov et al., 2024) is a promising method that performs preference optimization directly from an offline preference dataset instead of the complex training process of RLHF. Recently, many alternative preference optimization methods have been explored to address the limitations of DPO. RRHF (Yuan et al., 2023), SLiC-HF (Zhao et al., 2023), CPO (Xu et al., 2024), SimPO (Meng et al., 2024), and ORPO (Hong et al., 2024) eliminate the dependence on a reference model. R-DPO (Park et al., 2024) and SimPO (Meng et al., 2024) address the length exploitation problem, and IPO (Gheshlaghi Azar et al., 2024) addresses overfitting issues. KTO (Ethayarajh et al., 2024) deals with the non-paired preference datasets. β -DPO (Wu et al., 2024b) focuses on the dynamic calibration of β and data filtering mechanism. α -DPO (Wu et al., 2024a) presents an adaptive preference optimization method by introducing a dynamic reward margin to improve the performance of preference alignment. Concurrent with our work, SIMPER (Xiao et al., 2025) aims to eliminate the need for expensive hyperparameter tuning, making it computationally efficient. Recent studies have extended offline preference optimization to an iterative training setup (Pang et al., 2024b; Kim et al., 2025; Rosset et al., 2024) by continuously updating the pairwise preference dataset and reference model during the preference optimization process.

6 Conclusion

In this paper, we introduce the Implicit policy Regularized Preference Optimization (IRPO), which addresses the suboptimality issue of preference optimization methods with equivalent hyperparameter complexity to DPO, without requiring additional hyperparameters. Unlike DPO, which regularizes based on an explicitly given reference policy, we formulate the preference optimization based on a different regularization with the winning policy that

generally outperforms the reference policy. Then we present a novel preference optimization algorithm that can be directly optimized by discriminator training instead of introducing additional hyperparameters. Our experimental results show that IRPO consistently outperforms baseline algorithms with the same level of training complexity on benchmarks, including AlpacaEval 2 and Arena-Hard. Additionally, IRPO achieves comparable performance compared to algorithms that require a greater number of hyperparameters while dramatically reducing training costs.

Limitations

Our study acknowledges several limitations and suggests future directions for further improvements. First, our IRPO algorithm addresses the DPO’s sub-optimality problem, but the performance degradation problem in downstream tasks remains a limitation. This issue is a common challenge among fine-tuning methods, including preference alignment approaches, and is also one of our future research focuses. Second, our algorithm requires less cost than recent methods introducing additional hyperparameters but incurs more computational cost due to discriminator training than DPO. In future work, we could explore additional research on a more computationally efficient method compared to IRPO and DPO.

Ethics Statement

This paper introduces IRPO, a new preference optimization algorithm that offers a practical solution for scalable LLM fine-tuning. IRPO improves LLM performance on chat-based evaluation benchmarks, including AlpacaEval 2 and Arena-Hard. We comply with the Ethics Policy while conducting this research. We also hope that research on LLM fine-tuning, such as IRPO, can enhance our workplaces and enrich our daily lives by increasing productivity and creativity.

References

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](#). *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*, 39:324.

Paul Christiano, Jan Leike, Tom B. Brown, Miljan Martic, Shane Legg, and Dario Amodei. 2023. [Deep reinforcement learning from human preferences](#). *Preprint*, arXiv:1706.03741.

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. 2021. [Training verifiers to solve math word problems](#). *Preprint*, arXiv:2110.14168.

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-AI, Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, Aixin Liu, Bing Xue, Bingxuan Wang, Bochao Wu, Bei Feng, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, Damai Dai, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Han Bao, Hanwei Xu, Haocheng Wang, Honghui Ding, Huajian Xin, Huazuo Gao, Hui Qu, Hui Li, Jianzhong Guo, Jiashi Li, Jiawei Wang, Jingchang Chen, Jingyang Yuan, Junjie Qiu, Junlong Li, J. L. Cai, Jiaqi Ni, Jian Liang, Jin Chen, Kai Dong, Kai Hu, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong Zhang, Liang Zhao, Litong Wang, Liyue Zhang, Lei Xu, Leyi Xia, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Meng Li, Miaojun Wang, Mingming Li, Ning Tian, Panpan Huang, Peng Zhang, Qiancheng Wang, Qinyu Chen, Qiushi Du, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, R. J. Chen, R. L. Jin, Ruyi Chen, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shengfeng Ye, Shiyu Wang, Shuiping Yu, Shunfeng Zhou, Shuting Pan, S. S. Li, Shuang Zhou, Shaoqing Wu, Shengfeng Ye, Tao Yun, Tian Pei, Tianyu Sun, T. Wang, Wangding Zeng, Wanbiao Zhao, Wen Liu, Wenfeng Liang, Wenjun Gao, Wenqin Yu, Wentao Zhang, W. L. Xiao, Wei An, Xiaodong Liu, Xiaohan Wang, Xiaokang Chen, Xiaotao Nie, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xinyu Yang, Xinyuan Li, Xuecheng Su, Xuheng Lin, X. Q. Li, Xiangyue Jin, Xiaojin Shen, Xiaosha Chen, Xiaowen Sun, Xiaoxiang Wang, Xinnan Song, Xinyi Zhou, Xianzu Wang, Xinxia Shan, Y. K. Li, Y. Q. Wang, Y. X. Wei, Yang Zhang, Yanhong Xu, Yao Li, Yao Zhao, Yaofeng

- Sun, Yaohui Wang, Yi Yu, Yichao Zhang, Yifan Shi, Yiliang Xiong, Ying He, Yishi Piao, Yisong Wang, Yixuan Tan, Yiyang Ma, Yiyuan Liu, Yongqiang Guo, Yuan Ou, Yudian Wang, Yue Gong, Yuheng Zou, Yujia He, Yunfan Xiong, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuyang Zhou, Y. X. Zhu, Yanhong Xu, Yanping Huang, Yaohui Li, Yi Zheng, Yuchen Zhu, Yunxian Ma, Ying Tang, Yukun Zha, Yuting Yan, Z. Z. Ren, Zehui Ren, Zhangli Sha, Zhe Fu, Zhean Xu, Zhenda Xie, Zhengyan Zhang, Zhewen Hao, Zhicheng Ma, Zhigang Yan, Zhiyu Wu, Zihui Gu, Zijia Zhu, Zijun Liu, Zilin Li, Ziwei Xie, Ziyang Song, Zizheng Pan, Zhen Huang, Zhipeng Xu, Zhongyu Zhang, and Zhen Zhang. 2025. [Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning](#). *Preprint*, arXiv:2501.12948.
- Ning Ding, Yulin Chen, Bokai Xu, Yujia Qin, Shengding Hu, Zhiyuan Liu, Maosong Sun, and Bowen Zhou. 2023. [Enhancing chat language models by scaling high-quality instructional conversations](#). In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 3029–3051, Singapore. Association for Computational Linguistics.
- Yann Dubois, Balázs Galambosi, Percy Liang, and Tatsunori B Hashimoto. 2024. [Length-controlled alpacaeval: A simple way to debias automatic evaluators](#). *arXiv preprint arXiv:2404.04475*.
- Kawin Ethayarajh, Winnie Xu, Niklas Muennighoff, Dan Jurafsky, and Douwe Kiela. 2024. [Model alignment as prospect theoretic optimization](#). In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 12634–12651. PMLR.
- Leo Gao, John Schulman, and Jacob Hilton. 2022. [Scaling laws for reward model overoptimization](#). *Preprint*, arXiv:2210.10760.
- Aryo Pradipta Gema, Joshua Ong Jun Leang, Gihwon Hong, Alessio Devoto, Alberto Carlo Maria Mancino, Rohit Saxena, Xuanli He, Yu Zhao, Xiaotang Du, Mohammad Reza Ghasemi Madani, Claire Barale, Robert McHardy, Joshua Harris, Jean Kaddour, Emile van Krieken, and Pasquale Minervini. 2025. [Are we done with mmlu?](#) *Preprint*, arXiv:2406.04127.
- 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 *Proceedings of The 27th International Conference on Artificial Intelligence and Statistics*, volume 238 of *Proceedings of Machine Learning Research*, pages 4447–4455. PMLR.
- Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. [Generative adversarial networks](#). *Preprint*, arXiv:1406.2661.
- 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 Esibou, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael Smith, Filip Radenovic, Francisco Guzmán, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Govind Thattai, Graeme Nail, Gregoire Mialon, Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel Kloumann, Ishan Misra, Ivan Evtimov, Jack Zhang, Jade Copet, Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde, Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, Joshua Johnstun, Joshua Saxe, Junteng Jia, Kalyan Vasuden Alwala, Karthik Prasad, Kartikeya Upasani, Kate Plawiak, Ke Li, Kenneth Heafield, Kevin Stone, Khalid El-Arini, Krithika Iyer, Kshitiz Malik, Kuenley Chiu, Kunal Bhalla, Kushal Lakhotia, Lauren Rantala-Yeary, Laurens van der Maaten, Lawrence Chen, Liang Tan, Liz Jenkins, Louis Martin, Lovish Madaan, Lubo Malo, Lukas Blecher, Lukas Landzaat, Luke de Oliveira, Madeline Muzzi, Mahesh Pasupuleti, Mannat Singh, Manohar Paluri, Marcin Kardas, Maria Tsimpoukelli, Mathew Oldham, Mathieu Rita, Maya Pavlova, Melanie Kam-bador, Mike Lewis, Min Si, Mitesh Kumar Singh, Mona Hassan, Naman Goyal, Narjes Torabi, Nikolay Bashlykov, Nikolay Bogoychev, Niladri Chatterji, Ning Zhang, Olivier Duchenne, Onur Çelebi, Patrick Alrassy, Pengchuan Zhang, Pengwei Li, Petar Vasic, Peter Weng, Prajjwal Bhargava, Pratik Dubal, Praveen Krishnan, Punit Singh Koura, Puxin Xu, Qing He, Qingxiao Dong, Ragavan Srinivasan, Raj Ganapathy, Ramon Calderer, Ricardo Silveira Cabral, Robert Stojnic, Roberta Raileanu, Rohan Maheswari, Rohit Girdhar, Rohit Patel, Romain Sauvestre, Ronnie Polidoro, Roshan Sumbaly, Ross Taylor, Ruan Silva, Rui Hou, Rui Wang, Saghar Hosseini, Sahana Chennabasappa, Sanjay Singh, Sean Bell, Seohyun Sonia Kim, Sergey Edunov, Shaoliang Nie, Sharan Narang, Sharath Rapparthi, Sheng Shen, Shengye Wan, Shruti Bhosale, Shun Zhang, Simon Vandenhende, Soumya Batra, Spencer Whitman, Sten Sootla, Stephane Collot, Suchin Gururangan, Sydney Borodinsky, Tamar Herman, Tara Fowler, Tarek Sheasha, Thomas Georgiou, Thomas Scialom, Tobias Speckbacher, Todor Mihaylov, Tong Xiao, Ujjwal

Karn, Vedanuj Goswami, Vibhor Gupta, Vignesh Ramanathan, Viktor Kerkez, Vincent Gonguet, Virginie Do, Vish Vogeti, Vitor Albiero, Vladan Petrovic, Weiwei Chu, Wenhan Xiong, Wenyin Fu, Whitney Meers, Xavier Martinet, Xiaodong Wang, Xiaofang Wang, Xiaoqing Ellen Tan, Xide Xia, Xinfeng Xie, Xuchao Jia, Xuewei Wang, Yaelle Goldschlag, Yashesh Gaur, Yasmine Babaei, Yi Wen, Yiwen Song, Yuchen Zhang, Yue Li, Yuning Mao, Zacharie Delpierre Coudert, Zheng Yan, Zhengxing Chen, Zoe Papakipos, Aaditya Singh, Aayushi Srivastava, Abha Jain, Adam Kelsey, Adam Shajnfeld, Adithya Gangidi, Adolfo Victoria, Ahuva Goldstand, Ajay Menon, Ajay Sharma, Alex Boesenberg, Alexei Baevski, Allie Feinstein, Amanda Kallet, Amit Sangani, Amos Teo, Anam Yunus, Andrei Lupu, Andres Alvarado, Andrew Caples, Andrew Gu, Andrew Ho, Andrew Poulton, Andrew Ryan, Ankit Ramchandani, Annie Dong, Annie Franco, Anuj Goyal, Aparajita Saraf, Arkabandhu Chowdhury, Ashley Gabriel, Ashwin Bharambe, Assaf Eisenman, Azadeh Yazdan, Beau James, Ben Maurer, Benjamin Leonhardi, Bernie Huang, Beth Loyd, Beto De Paola, Bhargavi Paranjape, Bing Liu, Bo Wu, Boyu Ni, Braden Hancock, Bram Wasti, Brandon Spence, Brani Stojkovic, Brian Gamido, Britt Montalvo, Carl Parker, Carly Burton, Catalina Mejia, Ce Liu, Changhan Wang, Changkyu Kim, Chao Zhou, Chester Hu, Ching-Hsiang Chu, Chris Cai, Chris Tindal, Christoph Feichtenhofer, Cynthia Gao, Damon Civin, Dana Beaty, Daniel Kreymur, Daniel Li, David Adkins, David Xu, Davide Testuggine, Delia David, Devi Parikh, Diana Liskovich, Didem Foss, Dingkan Wang, Duc Le, Dustin Holland, Edward Dowling, Eissa Jamil, Elaine Montgomery, Eleonora Presani, Emily Hahn, Emily Wood, Eric-Tuan Le, Erik Brinkman, Esteban Arcaute, Evan Dunbar, Evan Smothers, Fei Sun, Felix Kreuk, Feng Tian, Filippos Kokkinos, Firat Ozgenel, Francesco Caggioni, Frank Kanayet, Frank Seide, Gabriela Medina Florez, Gabriella Schwarz, Gada Badeer, Georgia Swee, Gil Halpern, Grant Herman, Grigory Sizov, Guangyi, Zhang, Guna Lakshminarayanan, Hakan Inan, Hamid Shojanazeri, Han Zou, Hannah Wang, Hanwen Zha, Haroun Habeeb, Harrison Rudolph, Helen Suk, Henry Aspegren, Hunter Goldman, Hongyuan Zhan, Ibrahim Damlaj, Igor Molybog, Igor Tufanov, Ilias Leontiadis, Irina-Elena Veliche, Itai Gat, Jake Weissman, James Geboski, James Kohli, Janice Lam, Japhet Asher, Jean-Baptiste Gaya, Jeff Marcus, Jeff Tang, Jennifer Chan, Jenny Zhen, Jeremy Reizenstein, Jeremy Teboul, Jessica Zhong, Jian Jin, Jingyi Yang, Joe Cummings, Jon Carvill, Jon Shepard, Jonathan McPhie, Jonathan Torres, Josh Ginsburg, Junjie Wang, Kai Wu, Kam Hou U, Karan Saxena, Kartikay Khanelwal, Katayoun Zand, Kathy Matosich, Kaushik Veeraraghavan, Kelly Michelena, Keqian Li, Kiran Jagadeesh, Kun Huang, Kunal Chawla, Kyle Huang, Lailin Chen, Lakshya Garg, Lavender A, Leandro Silva, Lee Bell, Lei Zhang, Liangpeng Guo, Licheng Yu, Liron Moshkovich, Luca Wehrstedt, Madian Khabsa, Manav Avalani, Manish Bhatt, Martynas Mankus, Matan Hasson, Matthew Lennie, Matthias Reso, Maxim Groshev, Maxim Naumov,

Maya Lathi, Meghan Keneally, Miao Liu, Michael L. Seltzer, Michal Valko, Michelle Restrepo, Mihir Patel, Mik Vyatskov, Mikayel Samvelyan, Mike Clark, Mike Macey, Mike Wang, Miquel Jubert Hermoso, Mo Metanat, Mohammad Rastegari, Munish Bansal, Nandhini Santhanam, Natascha Parks, Natasha White, Navyata Bawa, Nayan Singhal, Nick Egebo, Nicolas Usunier, Nikhil Mehta, Nikolay Pavlovich Laptev, Ning Dong, Norman Cheng, Oleg Chernoguz, Olivia Hart, Omkar Salpekar, Ozlem Kalinli, Parkin Kent, Parth Parekh, Paul Saab, Pavan Balaji, Pedro Rittner, Philip Bontrager, Pierre Roux, Piotr Dollar, Polina Zvyagina, Prashant Ratanchandani, Pritish Yuvraj, Qian Liang, Rachad Alao, Rachel Rodriguez, Rafi Ayub, Raghotham Murthy, Raghu Nayani, Rahul Mitra, Rangrabhu Parthasarathy, Raymond Li, Rebekkah Hogan, Robin Battey, Rocky Wang, Russ Howes, Ruty Rinott, Sachin Mehta, Sachin Siby, Sai Jayesh Bondu, Samyak Datta, Sara Chugh, Sara Hunt, Sargun Dhillon, Sasha Sidorov, Satadru Pan, Saurabh Mahajan, Saurabh Verma, Seiji Yamamoto, Sharadh Ramaswamy, Shaun Lindsay, Shaun Lindsay, Sheng Feng, Shenghao Lin, Shengxin Cindy Zha, Shishir Patil, Shiva Shankar, Shuqiang Zhang, Shuqiang Zhang, Sinong Wang, Sneha Agarwal, Soji Sajuyigbe, Soumith Chintala, Stephanie Max, Stephen Chen, Steve Kehoe, Steve Satterfield, Sudarshan Govindaprasad, Sumit Gupta, Summer Deng, Sungmin Cho, Sunny Virk, Suraj Subramanian, Sy Choudhury, Sydney Goldman, Tal Rezem, Tamar Glaser, Tamara Best, Thilo Koehler, Thomas Robinson, Tianhe Li, Tianjun Zhang, Tim Matthews, Timothy Chou, Tzook Shaked, Varun Vontimitta, Victoria Ajayi, Victoria Montanez, Vijai Mohan, Vinay Satish Kumar, Vishal Mangla, Vlad Ionescu, Vlad Poenaru, Vlad Tiberiu Mihalescu, Vladimir Ivanov, Wei Li, Wenchen Wang, Wenwen Jiang, Wes Bouaziz, Will Constable, Xiao Cheng Tang, Xiaoqian Wu, Xiaolan Wang, Xilun Wu, Xinbo Gao, Yaniv Kleinman, Yanjun Chen, Ye Hu, Ye Jia, Ye Qi, Yenda Li, Yilin Zhang, Ying Zhang, Yossi Adi, Youngjin Nam, Yu, Wang, Yu Zhao, Yuchen Hao, Yundi Qian, Yunlu Li, Yuzi He, Zach Rait, Zachary DeVito, Zef Rosnbrick, Zhaoduo Wen, Zhenyu Yang, Zhiwei Zhao, and Zhiyu Ma. 2024. [The llama 3 herd of models](#). *Preprint*, arXiv:2407.21783.

Alex Gu, Baptiste Rozière, Hugh Leather, Armando Solar-Lezama, Gabriel Synnaeve, and Sida I. Wang. 2024. [Cruxeval: A benchmark for code reasoning, understanding and execution](#). *Preprint*, arXiv:2401.03065.

Alex Havrilla, Yuqing Du, Sharath Chandra Rapparthi, Christoforos Nalmpantis, Jane Dwivedi-Yu, Maksym Zhuravinskiy, Eric Hambro, Sainbayar Sukhbaatar, and Roberta Raileanu. 2024. [Teaching large language models to reason with reinforcement learning](#). *Preprint*, arXiv:2403.04642.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. 2021. [Measuring mathematical problem solving with the math dataset](#). *Preprint*, arXiv:2103.03874.

- Jiwoo Hong, Noah Lee, and James Thorne. 2024. [ORPO: Monolithic preference optimization without reference model](#). In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 11170–11189, Miami, Florida, USA. Association for Computational Linguistics.
- Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, L  lio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timoth  e Lacroix, and William El Sayed. 2023. [Mistral 7b](#). *Preprint*, arXiv:2310.06825.
- Dahyun Kim, Yungi Kim, Wonho Song, Hyeonwoo Kim, Yunsu Kim, Sanghoon Kim, and Chanjun Park. 2025. [sDPO: Don't use your data all at once](#). In *Proceedings of the 31st International Conference on Computational Linguistics: Industry Track*, pages 366–373, Abu Dhabi, UAE. Association for Computational Linguistics.
- Andreas K  pf, Yannic Kilcher, Dimitri von R  tte, Sotiris Anagnostidis, Zhi-Rui Tam, Keith Stevens, Abdullah Barhoum, Nguyen Minh Duc, Oliver Stanley, Rich  rd Nagyfi, Shahul ES, Sameer Suri, David Glushkov, Arnav Dantuluri, Andrew Maguire, Christoph Schuhmann, Huu Nguyen, and Alexander Mattick. 2023. [Openassistant conversations – democratizing large language model alignment](#). *Preprint*, arXiv:2304.07327.
- Xin Lai, Zhuotao Tian, Yukang Chen, Senqiao Yang, Xiangru Peng, and Jiaya Jia. 2024. [Step-dpo: Step-wise preference optimization for long-chain reasoning of llms](#). *Preprint*, arXiv:2406.18629.
- Tianle Li, Wei-Lin Chiang, Evan Frick, Lisa Dunlap, Tianhao Wu, Banghua Zhu, Joseph E. Gonzalez, and Ion Stoica. 2024. [From crowdsourced data to high-quality benchmarks: Arena-hard and benchbuilder pipeline](#). *Preprint*, arXiv:2406.11939.
- 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.
- Bill Yuchen Lin. 2024. [ZeroEval: A Unified Framework for Evaluating Language Models](#).
- Bill Yuchen Lin, Ronan Le Bras, and Yejin Choi. 2024. [ZebraLogic: Benchmarking the logical reasoning ability of language models](#).
- Haipeng Luo, Qingfeng Sun, Can Xu, Pu Zhao, Jianguang Lou, Chongyang Tao, Xiubo Geng, Qingwei Lin, Shifeng Chen, Yansong Tang, and Dongmei Zhang. 2025. [Wizardmath: Empowering mathematical reasoning for large language models via reinforced evol-instruct](#). *Preprint*, arXiv:2308.09583.
- Yu Meng, Mengzhou Xia, and Danqi Chen. 2024. [Simpo: Simple preference optimization with a reference-free reward](#). In *Advances in Neural Information Processing Systems (NeurIPS)*.
- Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll 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. 2022. [Training language models to follow instructions with human feedback](#). In *Advances in Neural Information Processing Systems*, volume 35, pages 27730–27744. Curran Associates, Inc.
- Richard Yuanzhe Pang, Weizhe Yuan, Kyunghyun Cho, He He, Sainbayar Sukhbaatar, and Jason Weston. 2024a. [Iterative reasoning preference optimization](#). *Preprint*, arXiv:2404.19733.
- Richard Yuanzhe Pang, Weizhe Yuan, He He, Kyunghyun Cho, Sainbayar Sukhbaatar, and Jason E Weston. 2024b. [Iterative reasoning preference optimization](#). In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.
- 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, Bangkok, Thailand. Association for Computational Linguistics.
- Xue Bin Peng, Aviral Kumar, Grace Zhang, and Sergey Levine. 2019. [Advantage-weighted regression: Simple and scalable off-policy reinforcement learning](#). *Preprint*, arXiv:1910.00177.
- Jan Peters and Stefan Schaal. 2007. [Reinforcement learning by reward-weighted regression for operational space control](#). In *Proceedings of the 24th International Conference on Machine Learning, ICML '07*, page 745–750, New York, NY, USA. Association for Computing Machinery.
- Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2024. [Direct preference optimization: Your language model is secretly a reward model](#). *Advances in Neural Information Processing Systems*, 36.
- Rajkumar Ramamurthy, Prithviraj Ammanabrolu, Kiant   Brantley, Jack Hessel, Rafet Sifa, Christian Bauckhage, Hannaneh Hajishirzi, and Yejin Choi. 2023. [Is reinforcement learning \(not\) for natural language processing: Benchmarks, baselines, and building blocks for natural language policy optimization](#). *Preprint*, arXiv:2210.01241.
- Corby Rosset, Ching-An Cheng, Arindam Mitra, Michael Santacrose, Ahmed Awadallah, and Tengyang Xie. 2024. [Direct nash optimization: Teaching language models to self-improve with general preferences](#). *Preprint*, arXiv:2404.03715.

- John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. [Proximal policy optimization algorithms](#). *Preprint*, arXiv:1707.06347.
- Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. 2024. [Deepseekmath: Pushing the limits of mathematical reasoning in open language models](#). *Preprint*, arXiv:2402.03300.
- Junkang Wu, Xue Wang, Zhengyi Yang, Jiancan Wu, Jinyang Gao, Bolin Ding, Xiang Wang, and Xiangnan He. 2024a. [\$\alpha\$ -dpo: Adaptive reward margin is what direct preference optimization needs](#). *Preprint*, arXiv:2410.10148.
- Junkang Wu, Yuexiang Xie, Zhengyi Yang, Jiancan Wu, Jinyang Gao, Bolin Ding, Xiang Wang, and Xiangnan He. 2024b. [\$\beta\$ -DPO: Direct preference optimization with dynamic \$\beta\$](#) . In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.
- Teng Xiao, Yige Yuan, Zhengyu Chen, Mingxiao Li, Shangsong Liang, Zhaochun Ren, and Vasant G Honavar. 2025. [SimPER: A minimalist approach to preference alignment without hyperparameters](#). In *The Thirteenth International Conference on Learning Representations*.
- 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](#). *Preprint*, arXiv:2401.08417.
- Hongyi Yuan, Zheng Yuan, Chuanqi Tan, Wei Wang, Songfang Huang, and Fei Huang. 2023. [Rrhf: Rank responses to align language models with human feedback](#). In *Advances in Neural Information Processing Systems*, volume 36, pages 10935–10950. Curran Associates, Inc.
- Yao Zhao, Rishabh Joshi, Tianqi Liu, Misha Khalman, Mohammad Saleh, and Peter J. Liu. 2023. [Slic-hf: Sequence likelihood calibration with human feedback](#). *Preprint*, arXiv:2305.10425.
- Chunting Zhou, Pengfei Liu, Puxin Xu, Srini Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, Lili Yu, Susan Zhang, Gargi Ghosh, Mike Lewis, Luke Zettlemoyer, and Omer Levy. 2023. [Lima: Less is more for alignment](#). *Preprint*, arXiv:2305.11206.

A Experimental Details

A.1 Training Objectives and Hyperparameters for Baseline Algorithms

This section provides details of our baseline algorithms. Table 4 summarizes the training objectives and training hyperparameters for each algorithm. For the learning rate, each method searches within the range of $[3e^{-7}, 5e^{-7}, 6e^{-7}, 1e^{-6}]$, following the same search strategy as in SimPO (Meng et al., 2024). For method-specific hyperparameters, we followed the search strategy from SimPO (Meng et al., 2024) and SimPER (Xiao et al., 2025). For discriminator training in all IRPO experiments, we used the same learning rate as that of policy training without conducting a separate hyperparameter search.

Method	Objective	Hyperparameters
SimPER (Xiao et al., 2025)	$-\exp\left(\frac{1}{ y_w } \log \pi_\theta(y_w x)\right) + \exp\left(\frac{1}{ y_l } \log \pi_\theta(y_l x)\right)$	-
ORPO (Hong et al., 2024)	$-\log p_\theta(y_w x) - \lambda \log \sigma\left(\log \frac{p_\theta(y_w x)}{1-p_\theta(y_w x)} - \log \frac{p_\theta(y_l x)}{1-p_\theta(y_l x)}\right)$, where $p_\theta(y x) = \exp\left(\frac{1}{ y } \log \pi_\theta(y x)\right)$	$\lambda \in [0.1, 0.5, 1.0, 2.0]$
RRHF (Yuan et al., 2023)	$\max\left(0, -\frac{1}{ y_w } \log \pi_\theta(y_w x) + \frac{1}{ y_l } \log \pi_\theta(y_l x)\right) - \lambda \log \pi_\theta(y_w x)$	$\lambda \in [0.1, 0.5, 1.0, 10.0]$
IPO (Gheshlaghi Azar et al., 2024)	$\left(\log \frac{\pi_\theta(y_w x)}{\pi_{\text{ref}}(y_w x)} - \log \frac{\pi_\theta(y_l x)}{\pi_{\text{ref}}(y_l x)} - \frac{1}{2\tau}\right)^2$	$\tau \in [0.01, 0.1, 0.5, 1.0]$
DPO (Rafailov et al., 2024)	$-\log \sigma\left(\beta \log \frac{\pi_\theta(y_w x)}{\pi_{\text{ref}}(y_w x)} - \beta \log \frac{\pi_\theta(y_l x)}{\pi_{\text{ref}}(y_l x)}\right)$	$\beta \in [0.01, 0.05, 0.1]$
SLiC-HF (Zhao et al., 2023)	$\max\left(0, \delta - \log \pi_\theta(y_w x) + \log \pi_\theta(y_l x)\right) - \lambda \log \pi_\theta(y_w x)$	$\lambda \in [0.1, 0.5, 1.0, 10.0]$ $\beta \in [0.1, 0.5, 1.0, 2.0]$
CPO (Xu et al., 2024)	$-\log \sigma\left(\beta \log \pi_\theta(y_w x) - \beta \log \pi_\theta(y_l x)\right) - \lambda \log \pi_\theta(y_w x)$	$\lambda = 1.0$ $\beta \in [0.01, 0.05, 0.1]$
R-DPO (Park et al., 2024)	$-\log \sigma\left(\beta \log \frac{\pi_\theta(y_w x)}{\pi_{\text{ref}}(y_w x)} - \beta \log \frac{\pi_\theta(y_l x)}{\pi_{\text{ref}}(y_l x)} + (\alpha y_w - \alpha y_l)\right)$	$\alpha \in [0.05, 0.1, 0.5, 1.0]$ $\beta \in [0.01, 0.05, 0.1]$
SimPO (Meng et al., 2024)	$-\log \sigma\left(\frac{\beta}{ y_w } \log \pi_\theta(y_w x) - \frac{\beta}{ y_l } \log \pi_\theta(y_l x) - \gamma\right)$	$\beta \in [2.0, 2.5, 10.0]$ $\gamma \in [0.3, 0.5, 1.0, 1.2, 1.4, 1.6]$
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 $z_{\text{ref}} = \mathbb{E}_{(x,y) \sim \mathcal{D}}[\beta \text{KL}(\pi_\theta(y x) \pi_{\text{ref}}(y x))]$	$\lambda_l = \lambda_w = 1.0$ $\beta \in [0.01, 0.05, 0.1]$
α -DPO (Wu et al., 2024a)	$-\log \sigma\left(u(x, y_w, y_l) - \text{sg}[\gamma + \alpha M^*(x, y_w, y_l)]\right)$, where $u(x, y_w, y_l) = \frac{\beta}{ y_w } \log \pi_\theta(y_w x) - \frac{\beta}{ y_l } \log \pi_\theta(y_l x)$	$\beta \in [2.5, 10.0]$ $\gamma \in [0.1, 0.3, 0.5]$ $\alpha \in [0.01, 0.05, 0.1, 0.2]$
IRPO	$-\log \sigma\left(\beta \log \pi_\theta(y_w x) - \beta \log \pi_\theta(y_l x) - \beta \log\left(\frac{d(x, y_w)}{1-d(x, y_w)} \cdot \frac{1-d(x, y_l)}{d(x, y_l)}\right)\right)$	$\beta \in [0.01, 0.05, 0.1]$
IRPO w/ LN	$-\log \sigma\left(\frac{\beta}{ y_w } \log \pi_\theta(y_w x) - \frac{\beta}{ y_l } \log \pi_\theta(y_l x) - \frac{\beta}{ y_w } \log \frac{d(x, y_w)}{1-d(x, y_w)} - \frac{\beta}{ y_l } \log \frac{1-d(x, y_l)}{d(x, y_l)}\right)$	$\beta \in [2.0, 2.5, 10.0]$

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

A.2 Evaluation Hyperparameters

The evaluation hyperparameters used in this paper are the same as those employed by SimPO. For evaluation on AlpacaEval 2, we use a temperature of 0.7 for the Mistral-Base setting, a temperature of 0.5 for the Mistral-Instruct setting, and a temperature of 0.9 for both Llama-3-Instruct settings. For evaluation on Arena-Hard, we use the default greedy decoding for all settings and methods.

A.3 Computation Resources

All the experiments in this paper were conducted on $16 \times$ NVIDIA A100 (40G) GPUs with 128 batchsize, based on the SimPO repository*.

*<https://github.com/princeton-nlp/SimPO>

A.4 Detailed Results

This section provides detailed results of Table 1, including the average generation length, standard deviation of win rate, and confidence interval of win rate.

Method	AlpacaEval 2				Arena-Hard					
	LC (%)	WR (%)	STD (%)	Length	SC (%)	LC (%)	WR (%)	95 CI high	95 CI low	Length
Llama-3-Instruct (8B)										
SFT	22.9	22.6	1.26	1899	19.9	20.1	20.6	1.5	1.7	585
SimPER	37.2	35.5	1.44	1913	29.1	30.8	30.8	1.8	1.9	524
ORPO	27.0	26.4	1.31	1902	26.2	27.0	27.3	2.0	2.4	536
RRHF	30.3	28.0	1.34	1850	26.9	27.8	27.2	2.1	2.7	504
IPO	35.9	35.5	1.41	1978	30.0	30.6	30.9	2.5	1.9	545
DPO	39.0	37.4	1.42	1921	31.7	32.4	32.8	2.3	2.1	528
SLiC-HF	27.9	28.3	1.32	1985	24.1	24.8	25.7	1.9	2.5	576
CPO	28.1	31.4	1.35	2152	25.8	27.2	29.0	2.0	2.1	622
R-DPO	38.8	36.9	1.41	1900	34.0	34.8	34.9	1.8	2.2	518
SimPO	43.5	38.6	1.42	1800	32.7	34.4	33.8	1.9	2.3	500
KTO	31.0	30.4	1.36	1923	25.1	26.3	26.8	2.1	2.2	538
α -DPO	40.2	33.9	1.44	1709	23.0	24.6	23.1	1.8	2.4	434
IRPO	40.0	45.3	1.50	2240	29.1	32.7	35.7	1.8	2.1	654
IRPO w/ LN	41.3	35.2	1.44	1721	27.6	26.1	28.3	2.4	1.9	469
Llama-3-Instruct-v0.2 (8B)										
SFT	22.9	22.6	1.26	1899	19.9	20.1	20.6	1.5	1.7	585
SimPER	41.8	37.8	1.49	1827	30.0	29.1	32.9	2.0	2.0	510
ORPO	35.7	32.3	1.38	1819	27.6	30.1	29.6	1.9	2.4	517
RRHF	32.6	28.7	1.33	1756	28.0	31.1	29.5	2.0	1.5	464
IPO	48.0	43.5	1.48	1832	31.2	34.7	34.3	1.8	2.0	525
DPO	46.6	46.0	1.48	2000	30.9	35.5	36.0	1.7	2.0	586
SLiC-HF	32.6	28.7	1.33	1756	28.1	31.1	29.6	1.9	2.0	465
CPO	32.7	35.1	1.38	2093	27.4	29.3	30.6	2.1	2.1	601
R-DPO	46.2	44.8	1.47	1966	28.2	32.4	32.7	2.4	2.3	564
SimPO	50.7	45.7	1.48	1795	31.2	36.7	37.0	2.1	2.7	534
KTO	32.0	30.4	1.35	1891	26.4	28.1	28.1	1.7	2.3	545
α -DPO	52.4	48.5	1.52	1868	25.4	31.0	30.7	2.2	2.1	501
IRPO	48.3	51.2	1.54	2187	29.7	34.4	37.9	2.2	2.5	703
IRPO w/ LN	52.3	44.0	1.46	1720	30.9	36.0	33.9	2.2	2.4	452

Table 5: Detailed results of Llama-3-Instruct and Llama-3-Instruct-v0.2 on AlpacaEval 2 and Arena-Hard. The results are reported as evaluation metrics for the win rate (**WR**), length-controlled win rate (**LC**), style-controlled win rate (**SC**), and the average generation length (**Length**). **STD** denotes a standard deviation of win rate, and **95 CI high** and **95 CI low** denote the confidence interval of win rate on the Arena-Hard benchmark.

Method	AlpacaEval 2				Arena-Hard					
	LC (%)	WR (%)	STD (%)	Length	SC (%)	LC (%)	WR (%)	95 CI high	95 CI low	Length
Mistral-Base (7B)										
SFT	7.5	4.5	0.64	924	3.9	2.7	2.2	0.5	0.6	546
SimPER	13.2	10.3	0.94	1466	11.6	11.3	8.5	1.1	1.6	518
ORPO	15.1	12.8	1.02	1576	11.3	8.9	8.6	1.3	1.3	560
RRHF	9.3	8.4	0.84	1626	8.6	6.6	6.7	1.0	0.9	592
IPO	11.5	9.5	0.92	1444	9.9	7.4	7.8	1.4	0.9	616
DPO	15.1	13.6	1.04	1712	11.5	8.7	9.3	0.9	1.3	658
SLiC-HF	11.2	8.9	0.85	1518	8.8	6.8	7.0	1.1	0.9	659
CPO	11.3	10.1	0.93	1789	7.2	5.4	6.1	1.0	1.0	764
R-DPO	17.8	13.6	1.05	1408	13.9	10.7	10.8	1.0	1.5	522
SimPO	21.8	21.6	1.23	1942	18.9	14.9	16.0	1.4	1.9	653
KTO	12.3	8.5	0.87	1200	8.4	6.4	5.6	0.9	0.9	421
α -DPO	17.1	16.5	1.13	1801	18.3	14.4	15.0	1.3	1.7	649
IRPO	12.9	13.7	1.05	3297	11.4	9.7	10.8	1.1	1.5	1240
IRPO w/ LN	17.8	15.6	1.11	1635	17.2	13.9	13.9	1.1	1.4	522
Mistral-Instruct (7B)										
SFT	18.0	15.1	1.08	1596	16.2	12.6	12.6	1.4	1.6	541
SimPER	24.5	26.8	1.35	2505	20.1	21.5	17.6	1.7	1.4	639
ORPO	24.2	23.9	1.26	1933	25.1	20.6	21.1	2.0	1.6	528
RRHF	26.0	24.1	1.26	1805	25.8	21.2	21.8	1.8	1.4	522
IPO	19.8	19.0	1.16	1930	20.7	16.8	17.0	1.4	1.6	556
DPO	25.4	22.7	1.23	1721	23.1	19.0	19.1	1.5	2.3	512
SLiC-HF	23.1	22.4	1.24	1915	21.9	17.8	18.8	2.1	1.7	593
CPO	25.1	30.5	1.35	3131	19.6	16.5	20.3	1.8	1.8	913
R-DPO	25.2	21.7	1.22	1679	22.4	17.9	18.0	1.9	1.6	504
SimPO	32.8	33.8	1.40	2074	24.8	20.5	20.8	2.1	2.0	526
KTO	24.4	22.7	1.25	1803	24.6	19.9	19.8	2.5	1.9	504
α -DPO	32.4	34.2	1.47	2085	21.5	18.6	19.1	1.5	1.7	510
IRPO	28.1	38.1	1.48	3476	18.9	16.1	21.7	1.9	1.5	1095
IRPO w/ LN	28.0	29.4	1.39	2332	22.1	22.9	19.1	1.8	2.1	544

Table 6: Detailed results of Mistral-Base and Mistral-Instruct on AlpacaEval 2 and Arena-Hard. The results are reported as evaluation metrics for the win rate (**WR**), length-controlled win rate (**LC**), style-controlled win rate (**SC**), and the average generation length (**Length**). **STD** denotes a standard deviation of win rate, and **95 CI high** and **95 CI low** denote the confidence interval of win rate on the Arena-Hard benchmark.

B Theoretical Analysis

In this section, we provide theoretical analyses of the KL-regularization with respect to the winning policy and the reward shaping in our objective Eq. (4).

B.1 KL-regularization with respect to the Winning Policy

First, we formally assume the property of winning policy π_w as follows:

Assumption B.1. For a given x and any y_w, y_l , if $r(x, y_w) \geq r(x, y_l)$, then

$$\frac{\pi_w(y_w|x)}{\pi_{\text{ref}}(y_w|x)} \geq \frac{\pi_w(y_l|x)}{\pi_{\text{ref}}(y_l|x)}$$

always holds.

Based on this assumption, we theoretically demonstrate that KL-regularization with respect to π_w is more effective than with respect to π_{ref} , as shown in the following proposition:

Proposition B.2. Let π^\dagger be the optimal solution of

$$\max_{\pi_\theta} \mathbb{E}_{\pi_\theta} [r(x, y)] - \beta \mathbb{D}_{\text{KL}}[\pi_\theta(y|x) || \pi_{\text{ref}}(y|x)],$$

and π^* be the optimal solution of

$$\max_{\pi_\theta} \mathbb{E}_{\pi_\theta} [r(x, y)] - \beta \mathbb{D}_{\text{KL}}[\pi_\theta(y|x) || \pi_w(y|x)].$$

Then,

$$\mathbb{E}_{\pi^*} [r(x, y)] \geq \mathbb{E}_{\pi^\dagger} [r(x, y)]$$

holds.

To prove this proposition, we first introduce the following lemma:

Lemma B.3. Let $p(y)$ and $q(y)$ be two probability distributions such that for any y_w and y_l with $r(y_w) \geq r(y_l)$,

$$\frac{p(y_w)}{q(y_w)} \geq \frac{p(y_l)}{q(y_l)}.$$

Then,

$$\mathbb{E}_p[r(y)] \geq \mathbb{E}_q[r(y)].$$

Proof of Lemma B.3.

$$\begin{aligned} \mathbb{E}_p[r(y)] &= \mathbb{E}_q \left[\frac{p(y)}{q(y)} r(y) \right] \\ &= \underbrace{\mathbb{E}_q \left[\frac{p(y)}{q(y)} \right]}_{=1} \mathbb{E}_q[r(y)] + \underbrace{\text{Cov}_q \left(\frac{p(y)}{q(y)}, r(y) \right)}_{\geq 0} \\ &\geq \mathbb{E}_q[r(y)]. \end{aligned}$$

Here,

$$\text{Cov}_q \left(\frac{p(y)}{q(y)}, r(y) \right) = \mathbb{E}_{y_1, y_2 \sim q} \left[\left(\frac{p(y_1)}{q(y_1)} - \frac{p(y_2)}{q(y_2)} \right) (r(y_1) - r(y_2)) \right] \geq 0,$$

since $\frac{p(y_1)}{q(y_1)} - \frac{p(y_2)}{q(y_2)}$ and $r(y_1) - r(y_2)$ have the same sign, which can be easily derived from the condition of the lemma. \square

Now, we provide the proof of Proposition B.2 using this lemma:

Proof of Proposition B.2. By the Lemma B.3, it is enough to show that for a given x and any y_w, y_l with $r(x, y_w) \geq r(x, y_l)$,

$$\frac{\pi^*(y_w|x)}{\pi^\dagger(y_w|x)} \geq \frac{\pi^*(y_l|x)}{\pi^\dagger(y_l|x)}.$$

Following prior work, the optimal solutions π^\dagger and π^* are formulated as follows:

$$\begin{aligned}\pi^\dagger(y|x) &= \frac{1}{Z^\dagger(x)} \pi_{\text{ref}}(y|x) \exp\left(\frac{1}{\beta} r(x, y)\right), \\ \pi^*(y|x) &= \frac{1}{Z^*(x)} \pi_w(y|x) \exp\left(\frac{1}{\beta} r(x, y)\right).\end{aligned}$$

Therefore, the ratio between π^* and π^\dagger is derived as

$$\frac{\pi^*(y|x)}{\pi^\dagger(y|x)} = \frac{\pi_w(y|x)}{\pi_{\text{ref}}(y|x)} \frac{Z^\dagger(x)}{Z^*(x)},$$

and therefore,

$$\frac{\pi^*(y_w|x)}{\pi^\dagger(y_w|x)} = \frac{\pi_w(y_w|x)}{\pi_{\text{ref}}(y_w|x)} \frac{Z^\dagger(x)}{Z^*(x)} \geq \frac{\pi_w(y_l|x)}{\pi_{\text{ref}}(y_l|x)} \frac{Z^\dagger(x)}{Z^*(x)} = \frac{\pi^*(y_l|x)}{\pi^\dagger(y_l|x)}.$$

Here, the inequality holds by Assumption B.1. □

B.2 Reward Shaping by Penalizing Log-probability of Losing Policy

Similar to Assumption B.1, we can assume the property of π_l as follows:

Assumption B.4. For a given x and any y_w, y_l , if $r(x, y_w) \geq r(x, y_l)$, then

$$\frac{\pi_l(y_w|x)}{\pi_{\text{ref}}(y_w|x)} \leq \frac{\pi_l(y_l|x)}{\pi_{\text{ref}}(y_l|x)}$$

always holds.

Here, we assume an additional property of π_{ref} as follows:

Assumption B.5. For a given x and any y_w, y_l , if $r(x, y_w) \geq r(x, y_l)$, then

$$\frac{\pi_{\text{ref}}(y_l|x)}{\pi_{\text{ref}}(y_w|x)} \geq 1$$

always holds.

This assumption presumes that the reference policy is less likely to generate high-quality samples that it has not observed during training than to generate low-quality samples.

Based on Assumptions B.1, B.4, and B.5, we can prove the following proposition:

Proposition B.6. Let π^* be the optimal solution of

$$\max_{\pi_\theta} \mathbb{E}_{\pi_\theta}[r(x, y)] - \beta \mathbb{D}_{KL}[\pi_\theta(y|x) || \pi_w(y|x)]$$

and π^{**} be the optimal solution of

$$\max_{\pi_\theta} \mathbb{E}_{\pi_\theta}[r(x, y) - \beta \log \pi_l(y|x)] - \beta \mathbb{D}_{KL}[\pi_\theta(y|x) || \pi_w(y|x)].$$

Then,

$$\mathbb{E}_{\pi^{**}}[r(x, y)] \geq \mathbb{E}_{\pi^*}[r(x, y)]$$

holds.

Proof. By the Lemma B.3, it is enough to show that for a given x and any y_w, y_l with $r(x, y_w) \geq r(x, y_l)$,

$$\frac{\pi^{**}(y_w|x)}{\pi^*(y_w|x)} \geq \frac{\pi^{**}(y_l|x)}{\pi^*(y_l|x)}.$$

Following prior work, the optimal solutions π^* and π^{**} are formulated as follows:

$$\begin{aligned} \pi^*(y|x) &= \frac{1}{Z^*(x)} \pi_w(y|x) \exp\left(\frac{1}{\beta} r(x, y)\right), \\ \pi^{**}(y|x) &= \frac{1}{Z^{**}(x)} \pi_w(y|x) \exp\left(\frac{1}{\beta} r(x, y) - \log \pi_l(y|x)\right) \\ &= \frac{1}{Z^{**}(x)} \frac{\pi_w(y|x)}{\pi_l(y|x)} \exp\left(\frac{1}{\beta} r(x, y)\right) \end{aligned}$$

Therefore, the ratio between π^{**} and π^* is derived as

$$\frac{\pi^{**}(y|x)}{\pi^*(y|x)} = \frac{1}{\pi_l(y|x)} \frac{Z^*(x)}{Z^{**}(x)},$$

and therefore,

$$\frac{\pi^{**}(y_w|x)}{\pi^*(y_w|x)} = \frac{1}{\pi_l(y_w|x)} \frac{Z^*(x)}{Z^{**}(x)} \geq \frac{1}{\pi_l(y_l|x)} \frac{\pi_{\text{ref}}(y_l|x)}{\pi_{\text{ref}}(y_w|x)} \frac{Z^*(x)}{Z^{**}(x)} \geq \frac{1}{\pi_l(y_l|x)} \frac{Z^*(x)}{Z^{**}(x)} = \frac{\pi^{**}(y_l|x)}{\pi^*(y_l|x)}.$$

Here, the inequality holds by Assumption B.4 and B.5. □

C Qualitative Analysis

In this section, we provide generation examples from the Llama-3-Instruct model after training with IRPO, IRPO w/ LN, and SimPO. Table 7 and 8 show that both IRPO and IRPO w/ LN generate structured responses. However, due to the effect of length normalization, IRPO w/ LN generates relatively concise responses, whereas IRPO generates longer but more detailed responses.

Input →	How can you determine if a person is genuinely interested in a conversation or simply being polite?
----------------	---

IRPO w/ LN →	Determining someone's genuine interest in a conversation versus mere politeness can be challenging, but here are some cues to help you gauge their engagement: <ol style="list-style-type: none">1. Eye contact: People who are genuinely interested usually maintain consistent eye contact, occasionally looking away to process information or re-focus. Avoidance or fleeting glances might indicate they're not fully engaged.2. Body language: Open and relaxed body language, such as uncrossed arms, leaning forward, or tilting their head, can suggest genuine interest. On the other hand, closed-off postures (e.g., crossing arms or legs) might indicate disinterest.3. Verbal cues: Listen for:<ul style="list-style-type: none">- Engaging questions: They ask follow-up questions, probe for more information, or seek clarification, showing a desire to understand your perspective.- Active listening: They paraphrase, summarize, or reflect back what you've said, demonstrating they're actively listening.- Nuanced responses: Their answers are thoughtful, specific, and show some depth of understanding.- Personal anecdotes or examples: They share relevant stories or experiences, indicating a connection to the topic.4. Pace and tone: A genuinely interested person will:<ul style="list-style-type: none">- Match their tone and pace to yours, creating a sense of mutual engagement.- Display enthusiasm, passion, or curiosity in their voice.5. Engagement with your input: They:<ul style="list-style-type: none">- Respond thoughtfully, even if they disagree or don't share the same opinion.- Build upon your ideas or add new perspectives.- Show willingness to learn or consider alternative views.6. Nonverbal feedback: Pay attention to:<ul style="list-style-type: none">- Nodding or head tilts, indicating they're following along and interested.- Smiling or laughing, which can suggest pleasure or amusement related to the conversation.7. Duration and persistence: If someone maintains a conversation over a longer period, even when it's not necessarily "important" or "productive," it may indicate genuine interest.8. Follow-up: After the conversation, they might:<ul style="list-style-type: none">- Send a summary or follow-up email to reiterate their interest.- Ask to continue the discussion at a later time.- Share relevant resources or articles. <p>Keep in mind that these cues are not foolproof, and people can still be polite without being genuinely interested. Ultimately, trust your instincts and observe the conversation dynamics to get a sense of whether someone is engaged and interested or simply being courteous.</p>
---------------------	--

Table 7: An AlpacaEval 2 generation example from the Llama-3-Instruct model after training with IRPO w/ LN.

Input → How can you determine if a person is genuinely interested in a conversation or simply being polite?

IRPO → Determining whether someone is genuinely interested in a conversation or just being polite can be challenging, but here are some nonverbal and verbal cues to help you make an educated guess:

Nonverbal Cues:

1. **Eye contact:** If someone is engaged, they tend to maintain consistent eye contact or occasionally glance away and then return their gaze. Avoidance of eye contact or a lack of interest in maintaining eye contact may indicate politeness rather than interest.
2. **Body language:** Open and relaxed posture, such as uncrossed arms, leaning forward, or facing the conversation partner, can suggest genuine interest. Tense or closed-off body language, like crossed arms or legs, might indicate disinterest.
3. **Facial expressions:** A person genuinely interested in the conversation may display subtle, attentive facial expressions like raised eyebrows, nodding, or a smile. Lack of facial movement or a "mask-like" expression could indicate politeness rather than engagement.
4. **Proximity and distance:** If someone is interested, they may lean in slightly or move closer to the conversation, indicating a desire to participate more fully. Maintaining a fixed distance or staying back may suggest politeness rather than interest.

Verbal Cues:

1. **Ask open-ended questions:** Genuinely interested individuals tend to ask questions that encourage further discussion, while polite individuals might ask yes/no questions or ones that can be answered briefly.
2. **Active listening:** People interested in the conversation will often paraphrase or summarize what you've said, demonstrating they're actively listening and processing the information.
3. **Follow-up questions:** If someone shows interest, they may ask additional questions to delve deeper into a topic or seek clarification. Polite individuals might not pursue further questions or may change the subject.
4. **Vocabulary and tone:** Engaged individuals may use more descriptive language, excited tone, or emotional language to convey their interest. Polite responses might be more formal, neutral, or lacking in enthusiasm.
5. **Initiating topics or switching topics:** Genuinely interested individuals may initiate new topics or suggest changing the subject to keep the conversation going. Polite people might stick to the original topic or avoid changing the subject.

Behavioral Patterns:

1. **Consistency:** Pay attention to whether someone consistently shows interest in conversations or only participates when forced to.
2. **Depth of engagement:** If someone only engages in superficial conversations or only asks brief, surface-level questions, it may indicate politeness rather than genuine interest.
3. **Follow-through:** If someone expresses interest in a specific topic or activity, do they follow through on their commitments or show enthusiasm when discussing it later?

Final Tips:

1. Observe the person's behavior over time to get a better sense of their conversational style.
2. Pay attention to inconsistencies in their behavior or verbal cues.
3. Don't assume someone is only being polite; people can be both polite and genuinely interested in a conversation.
4. Practice active listening yourself, as it can help you better detect genuine interest in others.

Remember, these cues are not foolproof, and individuals can vary in their behavior. However, by paying attention to these signs, you can increase your chances of determining whether someone is genuinely interested in a conversation or simply being polite.

Table 8: An AlpacaEval 2 generation example from the Llama-3-Instruct model after training with IRPO.

Input	→ How can you determine if a person is genuinely interested in a conversation or simply being polite?
--------------	---

SimPO	→ To determine whether someone is genuinely interested in a conversation or just being polite, observe their verbal and nonverbal cues. Here are some signs to look out for:
--------------	--

Genuinely interested:

1. **Active listening:** They maintain eye contact, nod their head, and ask follow-up questions to clarify or seek more information.
2. **Engaged body language:** Their posture is open, facing you, and they lean in slightly to show enthusiasm. Hands may be relaxed or gesturing emphatically.
3. **Probing questions:** They ask thoughtful, specific questions that demonstrate curiosity about your thoughts, experiences, or opinions.
4. **Sharing personal anecdotes:** They volunteer stories or experiences related to the topic, indicating a personal investment in the conversation.
5. **Varying tone and inflection:** Their voice exhibits enthusiasm, surprise, or humor, conveying genuine emotions.
6. **Maintaining the conversation's pace:** They don't rush to change the subject or dominate the conversation, allowing for a natural flow.

Polite but not interested:

1. **Limited eye contact:** They may glance away or avoid direct eye contact, indicating a lack of engagement.
2. **Monosyllabic responses:** They provide brief, curt answers without elaborating or showing interest.
3. **Superficial comments:** Their remarks are shallow or generic, lacking depth or insight.
4. **Changing the subject quickly:** They abruptly shift the conversation to avoid discussing topics that aren't familiar or comfortable for them.
5. **Failure to ask follow-up questions:** They don't demonstrate curiosity or seek further information.
6. **Distracted behavior:** They're easily sidetracked by their phone, surroundings, or other stimuli, indicating their mind isn't fully present in the conversation.

Contextual clues:

1. Frequency and consistency: If someone consistently shows interest in conversations over time, it's more likely a genuine trait.
2. Topic selection: If they initiate conversations or choose topics that align with their passions or expertise, it suggests a genuine interest.
3. Body language during pauses: If they use pauses to process information or think about their response, rather than simply waiting for their turn to speak, it indicates engagement.

Remember that people can exhibit a mix of both interested and polite behaviors, depending on the situation and their personality. By paying attention to these cues, you can generally determine whether someone is invested in the conversation or simply being courteous.

Table 9: An AlpacaEval 2 generation example from the Llama-3-Instruct model after training with SimPO.