β-DPO: Direct Preference Optimization with Dynamic β

Junkang Wu[1*] **Yuexiang Xie**[2] **Zhengyi Yang**[1] **Jiancan Wu**[1†]
Jinyang Gao[2] **Bolin Ding**[2] **Xiang Wang**[1] **Xiangnan He**[1†]
[1]University of Science and Technology of China
[2]Alibaba Group
{jkwu0909, yangzhy1998, wujcan, xiangwang1223, xiangnanhe}@gmail.com,
{yuexiang.xyx, jinyang.gjy, bolin.ding}@alibaba-inc.com

Abstract

Direct Preference Optimization (DPO) has emerged as a compelling approach for training Large Language Models (LLMs) to adhere to human preferences. However, the performance of DPO is sensitive to the fine-tuning of its trade-off parameter β, as well as to the quality of the preference data. We analyze the impact of β and data quality on DPO, uncovering that optimal β values vary with the informativeness of pairwise data. Addressing the limitations of static β values, we introduce a novel framework that dynamically calibrates β at the batch level, informed by data quality considerations. Additionally, our method incorporates β-guided data filtering to safeguard against the influence of outliers. Through empirical evaluation, we demonstrate that our dynamic β adjustment technique significantly improves DPO's performance across a range of models and datasets, offering a more robust and adaptable training paradigm for aligning LLMs with human feedback. The code is available at https://github.com/junkangwu/beta-DPO.

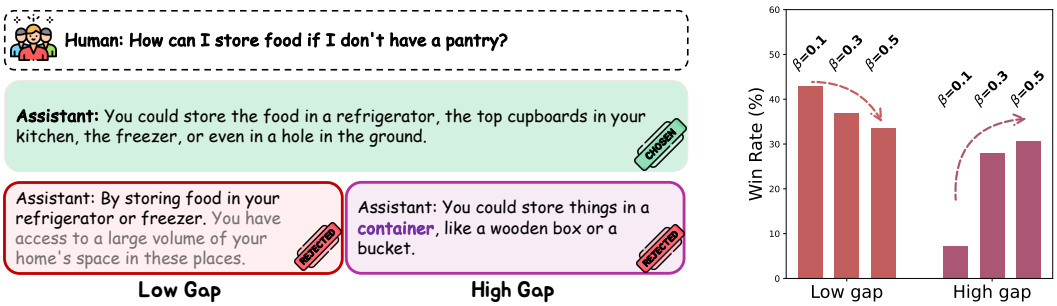
1 Introduction

The alignment of Large Language Models (LLMs) with human feedback, as explored in works like GPT-4 and LLaMA-2 [29, 38, 8], has marked a significant advancement in generating responses that are more helpful, factual, and ethical [30]. Among the various alignment strategies, Reinforcement Learning from Human Feedback (RLHF) [30] is a notable method that refines LLMs using the Proximal Policy Optimization (PPO) algorithm [36]. This approach employs a KL divergence penalty to ensure minimal deviation of the model from its original configuration, ensuring the retention of its initial characteristics while improving alignment.

Despite the effectiveness, RLHF's instability and computational requirements often limit its practical applications, prompting the exploration of alternatives like Direct Preference Optimization (DPO) [34]. DPO circumvents the reinforcement learning loop by exploiting the inherent connection between reward functions and optimal policies, thereby simplifying the policy model training. It encourages the model to favor the response that aligns with human preferences ($\mathbf{y}_w$) over the dispreferred ($\mathbf{y}_l$), implying DPO's sensitivity to the quality of pairwise data. The balance between maintaining the original reference model (π_{ref}) and incorporating new preferences ($\pi_{\boldsymbol{\theta}}$) is controlled by a β hyperparameter, whose lower values advocate for aggressive updates and higher values support more

[*]Work done at Alibaba Group.
[†]Jiancan Wu and Xiangnan He are the corresponding authors.

38th Conference on Neural Information Processing Systems (NeurIPS 2024).



(a) (b)

Figure 1: **(1a) Pairwise Data: Low vs. High Gap**: "Low gap" denotes cases where the chosen and rejected examples are closely similar, typically indicating high-quality, informative pairs. "High gap" signifies pairs with larger differences, implying lower-quality data. **(1b) Influence of Data Quality on β Selection:** Pythia-1.4B's performance on the HH dataset reveals a distinct trend: for "Low gap", a higher β reduces win rate, whereas for "High gap", an increased β improves it.

conservative adjustments:

$$\ell_{\text{DPO}}(\boldsymbol{\theta}) = \mathbb{E}_{\mathbf{x},\mathbf{y}_w,\mathbf{y}_l}[-\log\sigma\big(\beta\big[\log\big(\frac{\pi_{\boldsymbol{\theta}}(\mathbf{y}_w|\mathbf{x})}{\pi_{\text{ref}}(\mathbf{y}_w|\mathbf{x})}\big) - \log\big(\frac{\pi_{\boldsymbol{\theta}}(\mathbf{y}_l|\mathbf{x})}{\pi_{\text{ref}}(\mathbf{y}_l|\mathbf{x})}\big)\big]\big)].$$

However, the current DPO literature has largely overlooked the joint influence of β selection and pairwise data $(\mathbf{y}_w, \mathbf{y}_l)$'s quality on the alignment performance. To bridge this gap, we conduct a preliminary experiment to investigate how different β selections influence the model performance under two distinct data pair gap conditions, as shown in Figure 1b. In *low gap* scenarios (*cf.* Figure 1a), where the difference between preferred and dispreferred samples is minor, an increase in β (*e.g.,* from 0.1 to 0.5) corresponds with a decline in win rate (*e.g.,* from 42% to 33%). Conversely, in *high gap* situations (*cf.* Figure 1a) where a significant difference exists, an increase in β tends to improve DPO performance. Such contrasting outcomes highlight the necessity to tailor the β value contingent upon the data quality, especially in the presence of outliers [4].

Building upon these insights and recognizing the mixture nature of diverse quality in practically collected data [31], we propose a dynamic β selection strategy for DPO. This strategy adaptively adjusts β in response to the quality of pairwise data and robustifies DPO against data variability. Intuitively, one straightforward solution is personalizing β for each pair, rather than fixing it across the population of all pairs. While conceptually appealing, (1) instance-level β personalization can lead to optimization instabilities, particularly when dealing with a vast array of human preference instances (*cf.* Section 5.3). This issue underscores the challenge of balancing β updates with stable and scalable DPO. In this light, we propose to explore a *batch-level* adaptation of β, aiming to balance update aggressiveness and training stability. Moreover, (2) the frequent occurrence of outliers necessitates a strategy for accurately adjusting the batch-level β. The dataset notably features outliers, a challenge underscored by the significant reward discrepancy variations within the training samples of the dataset (*cf.* Section 4.1). Such conditions impede the model's ability to accurately estimate the batch-level β, thereby undermining the effectiveness of batch-level β calibration. To this end, we propose a simple yet effective dual-component approach:

- **Dynamic β Calibration at Batch-Level** (for Challenge 1): To mitigate optimization instabilities, we dynamically calibrate β within each batch. Specifically, this batch-level adjustment is based on data quality, with β being adaptively decreased for high-quality, closely-matched pairwise data (*i.e., low gap* data) to facilitate assertive updates. While for easily-discriminated pairs (*i.e., high-gap* data), β is increased to promote cautious updates, preventing overfitting to noise. This targeted batch-level calibration enables stable and responsive optimization.
- **β-Guided Data Filtering** (for Challenge 2): We implement a β-guided data filtering approach to tackle the frequent occurrence of outliers. By establishing a benchmark β value for filtering incoming data at the batch level, we maintain the fidelity of β estimation by prioritizing the most reliable and representative samples. As a result, it diminishes the impact of outliers that might

otherwise derail the optimization process, thus enhancing the precision and robustness of the batch-level β calibration.

Our contributions are threefold: (1) We investigate a pioneering study on the joint influence of β selection and pairwise data quality on the DPO performance. (2) We introduce a simple yet effective β-dynamic strategy for DPO, adaptively balancing the update aggressiveness and training stability. (3) Through empirical evaluations, our approach demonstrates marked improvements in performance across diverse conditions and model sizes (*e.g.,* achieving improvements exceeding 10% on models of various sizes, including Pythia-410M, 1.4B, and 2.8B).

2 Related Work

Reinforcement Learning from Human Feedback. Despite RLHF's effectiveness in aligning language models (LMs) with human values [10, 3, 38, 30, 14, 15], its complexity and resource demands have spurred the exploration of alternatives. RAFT [12] selects optimal training samples via an existing reward model, whereas RRHF [45] employs a simpler ranking loss, retaining PPO's efficiency. Diverse from these, DPO [34] directly optimizes LMs using a preference-based loss function, showcasing enhanced training stability in comparison to traditional RLHF. Innovatively, SLiC-HF [47] and KTO [13] devise loss functions rooted in human decision-making, focusing on preference calibration and utility optimization, respectively. Dr. DPO [42] consider robust settings where safety or group information is known at training time. Further, RSO [25] and ORPO [19] introduce efficient preference modeling and optimization, with ORPO uniquely combining supervised fine-tuning and preference alignment. These advancements reflect the ongoing shift towards more efficient, nuanced RL methods.

Data Quality in LLM's Alignment. Recent studies have increasingly recognized the significance of data quality in the alignment of LLMs. For example, LIMA [48] leverages heuristics such as post scores, response lengths, formatting, and topics to manually craft 1000 high-quality datasets from StackExchange, wikiHow, and Reddit for superficial alignment. In a similar vein, Bai et al. [3] prioritize data points based on user engagement levels for dataset assembly. Rejection Sampling (RS) and Best-of-N (BoN) techniques, as evidenced in the works of Nakano et al. [28] and Gao et al. [16], involve selecting the optimal candidate from N generated possibilities through the application of a reward model. To enhance preference optimization, RSO [25] uses statistical weightings to differentiate outcomes from an optimal policy and a base SFT policy. Besides, fDPO [27] employs a Reward Model to filter out low-quality data, effectively addressing dataset quality concerns.

3 Preliminaries

Given a text sequence (commonly referred to as a prompt) $\mathbf{x}$, a sequence $\mathbf{y} = [y_1, y_2, \dots y_N]$ is generated as a response to the prompt $\mathbf{x}$. An autoregressive language model π, when provided with the prompt $\mathbf{x}$, can generate the response sequence $\mathbf{y}$ following the probability decomposition:

$$\pi(\mathbf{y}|\mathbf{x}) = \prod_{t=1}^{N} \pi(y_i|\mathbf{x}, \mathbf{y}_{<t}), \tag{1}$$

where $\mathbf{y}_{<t}$ denotes the preceding tokens in the response sequence. Now, given a preference dataset $\mathcal{D} = \{(\mathbf{x}^{(i)}, \mathbf{y}_w^{(i)}, \mathbf{y}_l^{(i)})\}_{i=1}^{M}$, wherein each triplet consists of a prompt $\mathbf{x}$ with two responses $\mathbf{y}_w \in \Sigma^*$ and $\mathbf{y}_l \in \Sigma^*$, with Σ^* representing the alphabet, a preference oracle — either a human annotator or a language model — provides preference feedback $o(\mathbf{y}_w \succ \mathbf{y}_l|\mathbf{x}) \in \{0, 1\}$, indicating whether $\mathbf{y}_w$ is preferred over $\mathbf{y}_l$. We denote $\mathbb{P}(\mathbf{y}_w \succ \mathbf{y}_l|\mathbf{x}) = \mathbb{E}[o(\mathbf{y}_w \succ \mathbf{y}_l|\mathbf{x})]$ the probability of $\mathbf{y}_w$ "winning the duel" over $\mathbf{y}_l$. The Kullback-Leibler (KL) divergence between two probability distributions with densities p and q is defined as $\mathrm{KL}(p\|q) = \mathbb{E}_{\mathbf{y}\sim p(\mathbf{y})}\left[\log \frac{p(\mathbf{y})}{q(\mathbf{y})}\right]$.

RLHF with Reward Models. Christiano et al. [10] pioneer the learning of a reward function $r(\mathbf{y}; \mathbf{x})$ based on the Bradley-Terry model [7]. This model is deployed for the triplet of a prompt ($\mathbf{x}$) and two responses ($\mathbf{y}_w, \mathbf{y}_l$), establishing the likelihood of preference for $\mathbf{y}_w$ over $\mathbf{y}_l$ as:

$$\mathbb{P}(\mathbf{y}_w \succ \mathbf{y}_l|\mathbf{x}) = \frac{\exp(r(\mathbf{y}_w; \mathbf{x}))}{\exp(r(\mathbf{y}_w; \mathbf{x})) + \exp(r(\mathbf{y}_l; \mathbf{x}))} = \sigma\big(r(\mathbf{y}_w; \mathbf{x}) - r(\mathbf{y}_l; \mathbf{x})\big), \tag{2}$$

where $\sigma(x) = e^x/(e^x + 1)$ represents the logistic function. The approach for estimating the reward function within the Bradley-Terry framework is to maximize the log-likelihood $\log \mathbb{P}(\mathbf{y}_w \succ \mathbf{y}_l|\mathbf{x})$. Assuming accurate estimation of the true reward function $r(\mathbf{y}; \mathbf{x})$, Christiano et al. [10] propose to solve the following problem with policy optimization algorithms in RL such as PPO [36]:

$$\max_{\boldsymbol{\theta}} \mathbb{E}_{\mathbf{x}\sim\mathcal{X},\mathbf{y}\sim\pi_{\boldsymbol{\theta}}(\cdot|\mathbf{x})}[r(\mathbf{y};\mathbf{x})] - \beta\mathbb{E}_{\mathbf{x}\sim\mathcal{X}}[\mathrm{KL}(\pi_{\boldsymbol{\theta}}(\cdot|\mathbf{x})\|\pi_{\text{ref}}(\cdot|\mathbf{x}))], \tag{3}$$

where $\mathcal{X}$ represents the prompt distribution, $r(\mathbf{y}; \mathbf{x})$ denotes the reward function learned using the Bradley-Terry model on the preference dataset, π_{ref} is the fixed reference model (typically selected to be the one post supervised fine-tuning), and β serves as the penalty coefficient of the KL divergence.

Directed Preference Optimization (DPO). Rafailov et al. [34] identify that the optimization problem above has a closed-form solution such that for any $\mathbf{y}$,

$$\pi^*(\mathbf{y}|\mathbf{x}) \propto \pi_{\text{ref}}(\mathbf{y}|\mathbf{x}) \exp(r(\mathbf{y};\mathbf{x})/\beta),$$

which can be further converted to the DPO loss for any triplet $(\mathbf{x}, \mathbf{y}_w, \mathbf{y}_l)$:

$$\ell_{\text{DPO}}(\mathbf{x}, \mathbf{y}_w, \mathbf{y}_l; \boldsymbol{\theta}; \pi_{\text{ref}}) = -\log\sigma\left(\beta\left[\log\left(\frac{\pi_{\boldsymbol{\theta}}(\mathbf{y}_w|\mathbf{x})}{\pi_{\text{ref}}(\mathbf{y}_w|\mathbf{x})}\right) - \log\left(\frac{\pi_{\boldsymbol{\theta}}(\mathbf{y}_l|\mathbf{x})}{\pi_{\text{ref}}(\mathbf{y}_l|\mathbf{x})}\right)\right]\right). \tag{4}$$

4 Method

In this section, we investigate the critical connection between the parameter β and the quality of pairwise data in optimizing DPO. We present empirical evidence demonstrating the effect of β settings on DPO performance across datasets of varying quality. Our proposed method, β-DPO, introduces dynamic calibration of β and a data filtering mechanism tailored to improve DPO's effectiveness across diverse data conditions.

4.1 Motivation: The Impact of Pairwise Data Quality on β Selection

Scrutinizing Equation (4), we argue that DPO's effectiveness critically hinges on two factors: the choice of β and the quality of pairwise data. Here, we conduct experiments to demonstrate the influence of variations in β and data quality on DPO, pivotal for its effective real-world application.

Datasets. We utilize the Anthropic HH dataset [3] for our experimental analysis, which contains approximately 170,000 dialogues between humans and an automated assistant. In this dataset, a human inquiry, denoted as $\mathbf{x}$, is paired with two responses $(\mathbf{y}_w, \mathbf{y}_l)$, where $\mathbf{y}_w$ represents the response favored by the human annotator, while y_l is the alternate response. Notably, the alternate response y_l retains informational value, making this dataset high-quality with minimal discrepancies between the response pairs, which we classify as a *low gap* dataset. To further explore the impact of data quality on DPO, we construct a synthetic dataset, referred to as the *high gap* dataset. This dataset differs from the *low gap* dataset by introducing a greater disparity between responses. Specifically, the alternative response y_l is generated by a Supervised FineTuned (SFT) Pythia-2.8B model, while the preferred response y_w remains consistent with the original dataset. We also combine the two datasets in equal proportion to create a *mixed gap* dataset, with each contributing 50%, to incorporate the characteristics of both the *low gap* and *high gap* datasets.

Models and Metrics. Our study evaluates various model sizes, specifically Pythia-410M, Pythia-1.4B, and Pythia-2.8B [5], to ensure a comprehensive assessment. Following the established protocol in DPO [34], each model iteration undergoes a single epoch with a batch size of 64. This setup provides a uniform basis for evaluation across different models. We adopt the evaluation strategy from DPO [34] to calculate the *win rate*, a metric that measures how often the GPT-4 model prefers a response generated by our models over the default chosen response on the subset of the test dataset.

Findings: **(1) The optimal value of β varies with data quality, reflecting divergent performance patterns across datasets.** In Figure 2, we present the win rate results across three levels of pairwise data gap, each evaluated under varying β parameters. As can be observed, with *low gap* pairwise data, a smaller β value is preferable for optimizing performance. This is likely

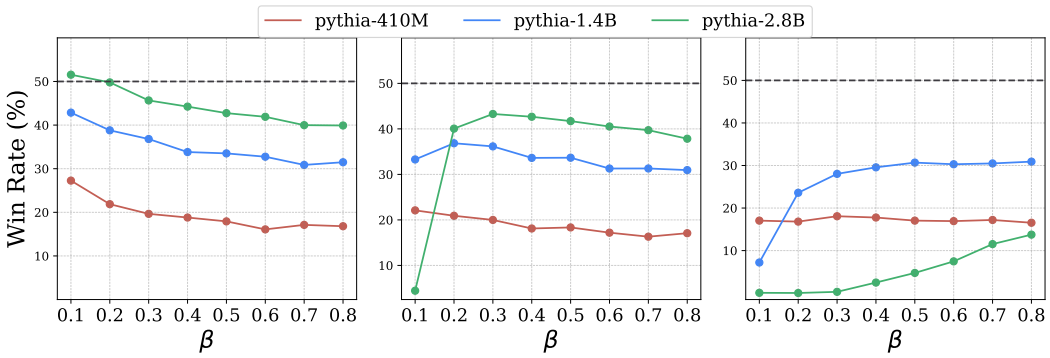


Figure 2: Win rate performance of DPO across different β settings on the *low gap*, *mixed gap*, and *high gap* datasets.

because the informative content of such data allows a lower β to facilitate more substantial updates, thereby enhancing alignment accuracy. Conversely, for *high gap* pairwise data, maintaining a low β may lead to overfitting, which significantly undermines the alignment process. The *mixed gap* dataset — a combination of both *low gap* and *high gap* datasets — exhibits a more nuanced performance pattern, suggesting the necessity for a dynamic β calibration strategy to adapt to varying data quality. Consequently, adhering to a fixed β value, *i.e.,* configuring β at the population level, might be inadequate for the dynamic and varied nature of real-world datasets.

(2) The dataset exhibits notable outliers. In Figure 3, utilizing the Pythia-2.8B model, we evaluate the data quality by examining the distribution of reward discrepancy for each triplet (which we will define as "individual reward discrepancy" later) within the HH dataset's training samples. The tails of the density plot extend beyond the highlighted percentiles, suggesting the existence of data samples with significantly higher or lower reward discrepancies. Notably, cases with significantly higher rewards for positive samples over negative ones suggest low informational value, as these discrepancies likely do not contribute meaningfully to the model's learning process. Whereas the opposite cases hint at potential labeling errors. Both cases deviate from an expected rational distribution range and are thus classified as outliers. For further details on outliers, kindly refer to Appendix A.2.

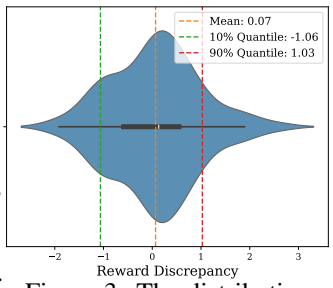


Figure 3: The distribution of individual reward discrepancy $(r(\mathbf{y}_w^{(i)};\mathbf{x}^{(i)}) - r(\mathbf{y}_l^{(i)};\mathbf{x}^{(i)}))$ on the training dataset of HH.

4.2 Method: Dynamic β Calibration in DPO

Through our empirical analysis, we highlight the sensitivity of DPO to β selections and the frequent occurrence of outliers. Hence, determining the optimal β value requires careful consideration of the quality of pairwise data while also addressing the influence of outliers. This prompts the question: *what criteria define a superior choice of β?* In response, we propose the following guiding principles:

Principle 1: *The optimal β value should be responsive to pairwise data's quality.*

Principle 2: *The selection of β value should minimize the influence of outliers.*

4.2.1 Dynamic β Calibration at Batch-Level

We begin by introducing the concept termed '*individual reward discrepancy*', which represents the difference between the rewards of winning and losing for each triplet, serving as a measurement for pairwise data quality. Formally, for a triplet $(\mathbf{x}^{(i)}, \mathbf{y}_w^{(i)}, \mathbf{y}_l^{(i)}) \in \mathcal{D}$, the individual reward discrepancy is defined as $M_i = r(\mathbf{y}_w^{(i)};\mathbf{x}^{(i)}) - r(\mathbf{y}_l^{(i)};\mathbf{x}^{(i)})$. While our primary analysis utilizes the implicit reward model induced by the policy trained using DPO, we also conducted comparative experiments with an explicit reward model. The details of these experiments with the explicit RM can be found in Appendix A.5. For the DPO-based implicit reward model, the reward discrepancy is expressed as:

$$M = \beta_0 \log \left(\frac{\pi_\theta(y_w \mid x)}{\pi_{\text{ref}}(y_w \mid x)} \right) - \beta_0 \log \left(\frac{\pi_\theta(y_l \mid x)}{\pi_{\text{ref}}(y_l \mid x)} \right).$$

Here, π_θ represents the policy being optimized, and π_{ref} denotes the reference policy. This formulation captures the difference in the log-probabilities of the winning and losing outcomes, weighted by the parameter β. Motivated by our guiding principles, a straightforward approach is to assign a distinct β to each triplet, allowing each β to serve as a parameter tailored to its respective triplet. This instance-level dynamic β adaption can be formulated as follows:

$$\beta_i = \beta_0 + \alpha(M_i - M_0)\beta_0 = [1 + \alpha(M_i - M_0)]\beta_0, \tag{5}$$

where β_0 represents the benchmark hyperparameter intrinsic to DPO, typically set to 0.1. The term M_0 denotes a predetermined threshold, and the coefficient α is a scaling factor within the interval $[0, 1]$ that adjusts the influence of M_i on β_i. Specifically, when $\alpha = 0$, β_i remains constant at β_0, thus maintaining the standard DPO framework without modification.

Equation (5) illustrates that β_i increases monotonically with M_i, allowing the model to adjust the β value based on the running reward differential between paired samples. Nevertheless, such instance-level adjustments may introduce instabilities during training. Prior studies have shown that a minibatch approach can help avoid saddle points or local minima [17], as well as mitigate the impact of noise [35, 6]. Drawing inspiration from these benefits, we propose a batch-level dynamic estimation methodology for β:

$$\beta_{\text{batch}} = [1 + \alpha(\mathbb{E}_{i\sim\text{batch}}[M_i] - M_0)]\beta_0. \tag{6}$$

In practical applications, the threshold M_0 can be estimated by employing the global mean of M_i with a moving average updating scheme [24]:

$$M_0 \leftarrow mM_0 + (1 - m)\mathbb{E}_{i\sim\text{batch}}[M_i], \tag{7}$$

where $m \in [0, 1)$ is a momentum coefficient. Such a batch-level calibration method introduces only one new parameter, α, to control the scale of β adjustment. The calculation of $\mathbb{E}_{i\sim\text{batch}}[M_i]$ is straightforward within DPO procedures, thereby incurring no additional computational overhead.

4.2.2 β-Guided Data Filtering

To mitigate the adverse impact of outliers on the β selection process, we introduce a β-guided data filtering mechanism. Informed by 3σ confidence criterion [33], this strategy employs a probabilistic model to assess the significance of each triplet $(\mathbf{x}^{(i)}, \mathbf{y}_w^{(i)}, \mathbf{y}_l^{(i)})$ based on its individual reward discrepancy M_i, which is defined as:

$$p(M_i) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(M_i - M_0)^2}{2\sigma^2} \right), \tag{8}$$

where M_0 and σ represent the mean and standard deviation of M_i across the training dataset, respectively. Similar to the updating scheme of M_0 in Equation (7), we dynamically estimate the value of σ using the moving average method:

$$\sigma \leftarrow m\sigma + (1 - m)\sqrt{\mathbb{V}_{i\sim\text{batch}}[M_i]}. \tag{9}$$

This probabilistic weighting discerns the relative importance of each sample, guiding the selection of $|\text{batch}| \times \rho$ samples (without replacement) based on their calculated probabilities $p(M_i)$. Here, ρ denotes the selection ratio, defaulting to 0.8, a choice validated by preliminary experiments aimed at optimizing training efficiency and model accuracy.

This process is iterated for each training batch, ensuring that the training data is continuously updated to reflect the most informative samples. The introduction of the β-guided data filtering strategy is instrumental in fortifying the model against outliers, thereby facilitating the accurate estimation of the β value.

Highlights: We underline the following key features of our proposed β-DPO framework:

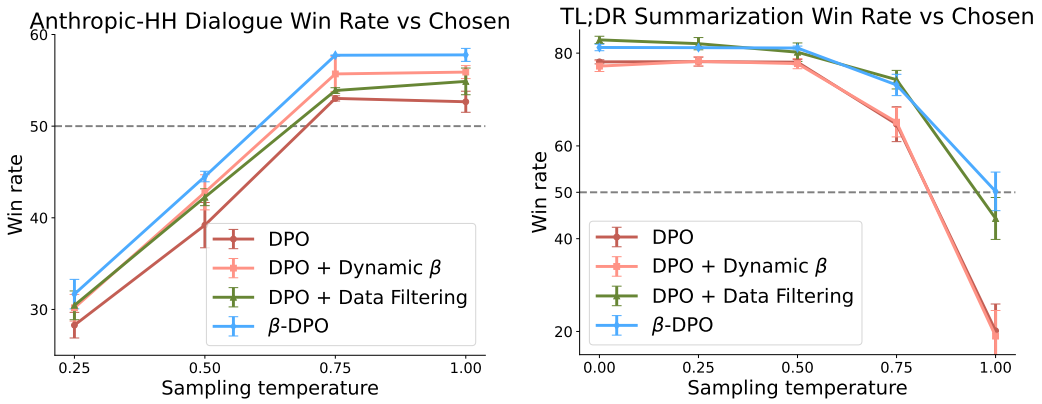


Figure 4: **Left.** The win rates computed by GPT-4 evaluations for the Anthropic-HH one-step dialogue; β-DPO consistently outperforms across all sampling temperatures. **Right.** In the comparison of TL;DR summarization win rates versus chosen summaries with GPT-4 as the evaluator, β-DPO is distinguished as the only strategy achieving a win rate over 50% across different sampling temperatures.

- **Simplicity**: β-DPO is extremely straightforward and quick to implement. It merely involves a dynamic β adjustment at the batch level and a β-guided data filtering mechanism, both of which are predicated upon the reward discrepancy denoted by M_i.
- **Efficiency**: Unlike other methodologies [25, 27, 32, 40] that necessitate an additional gold model for data filtering, our method leverages the running reward discrepancy M_i within the DPO framework. Moreover, our empirical studies indicate that β-DPO exhibits insensitivity to the hyperparameters ρ. A default setting of $\rho = 0.8$ typically yields satisfactory performance.
- **Model-agnostic**: As a variant of the traditional DPO, the proposed β-DPO can function as a plug-and-play module. It allows straightforward integration of future enhancements and extensions within the DPO framework. Our empirical investigations corroborate this assertion.

4.3 Discussion with Previous Studies

Relations to Data Selection. An increasing volume of works [25, 27, 48, 32, 44] have underscored the impact of data quality on the performance of LLM's alignment. A common practice among these efforts involves employing a so-called "gold model" for data selection. This approach, however, introduces significant computational demands and the choice of the gold model directly influences the resultant system's performance. The focus of this work, it should be noted, is not to propose a superior strategy for data selection. Instead, we aim to enhance adaptability to the quality of data by dynamically adjusting the β parameter. This adjustment facilitates improved β estimation by selecting data based on the reward. Moreover, Section 5.2 illustrates the compatibility of dynamic β adjustment with other data selection methodologies.

Relations to Recent Temperature Schemes. Dynamic temperature frameworks have been introduced in the realm of contrastive learning, motivated by various objectives, such as addressing out-of-distribution tasks [46] or accommodating long-tail data distributions [23]. The work most closely related to ours, MACL [20], has indeed proposed an alignment-adaptive strategy; however, its primary aim was to navigate the uniformity-tolerance dilemma. Hence, the integration of dynamic temperature mechanisms with LLM's alignment remains an underexplored area against this backdrop.

5 Experiments

In this section, we commence by conducting an empirical evaluation of β-DPO on two specific tasks: dialogue generation and summarization. Subsequently, we analyze the various adaptations of the proposed method β-DPO. Concluding this section, we underscore the imperative need for batch-level dynamic β calibration, highlighting its significance in the context of our study.

Table 1: Win rate comparison of Pythia-410M, -1.4B, and -2.8B models on the Anthropic HH dataset, evaluated using GPT-4.

Method	**410M**	**1.4B**	**2.8B**
DPO	26.19	42.78	51.51
DPO + Dynamic β	$27.15^{+3.67\%}$	$43.51^{+1.71\%}$	$55.19^{+7.14\%}$
DPO + Data Filtering	$29.03^{+10.84\%}$	$46.99^{+9.84\%}$	$53.42^{+3.71\%}$
β-DPO	$30.18^{+15.23\%}$	$48.67^{+13.77\%}$	$57.07^{+10.79\%}$

5.1 Empirical Evaluation of β-DPO on Dialogue Generation and Summarization

Datasets and Setup. Our experiments are conducted on the Anthropic HH dataset [3] and Reddit TL;DR summarization dataset [39]. The training configuration follows from Rafailov et al. [34]. The goals of these experiments are to study: 1) How β-DPO performs on single-turn dialogue generation and summarization tasks; 2) How the sampling temperature affects the performance of β-DPO; 3) How β-DPO works with different model sizes. For detailed experimental settings, please refer to Appendix A.1.

Baselines. In our comparison, we examine the performance of β-DPO relative to its counterparts: the standard DPO, DPO implemented with a dynamic β yet devoid of β-guided data filtering, and DPO complemented by data filtering with β fixed at 0.1.

Win Rate Across different Sampling Temperature. An analysis of win rates derived from GPT-4 evaluations on the Anthropic-HH one-step dialogue demonstrates that β-DPO consistently outperforms across all sampling temperatures, as depicted in Figure 4 (Left). Furthermore, for the TL;DR summarization task, β-DPO stands out as the only approach achieving win rates above 50% for diverse sampling temperatures, which is visually represented in Figure 4 (Right). The data also suggests that while both dynamic β and data filtering enhance DPO's effectiveness, the impact of data filtering is especially pronounced in the summarization task, likely due to the inherently greater noise present in the Reddit TL;DR summarization dataset. Notably, β-DPO exhibits a remarkable degree of robustness to variations in sampling temperature. As the temperature incrementally escalates from 0.0 to 1.0, the win rate for standard DPO plunges to a mere 25%, whereas β-DPO maintains a commendable performance level with a win rate of 54%.

Win Rate Across Different Model Sizes. We further evaluate the performance of β-DPO on the Anthropic HH dataset with Pythia-410M, -1.4B, and -2.8B models. The results are summarized in Table 1. We observe that β-DPO consistently outperforms DPO, DPO with dynamic β, and DPO with data filtering across all model sizes. We observe that in a smaller model, the improvement of data filtering is more significant, while in a larger model, the improvement of dynamic β is more significant. We attribute this to the fact that the larger model has more capacity to learn the optimal policy, while the smaller model needs more help from the data filtering.

5.2 Adaptations of β-DPO

In this section, our inquiry is twofold: first, we aim to understand the performance of β-DPO when applied across various filtering strategies; second, we examine its efficacy across different adaptations of the DPO framework. In terms of filtering strategies, prevailing studies [32, 44] in the domain largely employ a gradient-based approach. We propose to extend this methodology into three distinct scenarios. This involves arranging the gradients of pairwise data within a batch and consequently: (1) Excluding the top 20% of samples, hereby referred to as **Filter Head**, (2) Excluding the bottom 20% of samples, hereby referred to as **Filter Tail**, (3) Excluding both the top and bottom 10% of samples, a method we denote as **Filter Tail & Head**. For a fair comparison, we maintain the amount of data excluded at 20% for the above strategies. Second, we integrate three variants of DPO into our analysis: the IPO [2], a novel approach that facilitates learning directly from preferences without the need for the Bradley-Terry (BT) model. Additionally, we consider the KTO [13], which focuses on discerning whether a preference is desirable or undesirable and SPPO [43], which approximates the Nash equilibrium. For detailed settings, we refer the reader to the supplementary material.

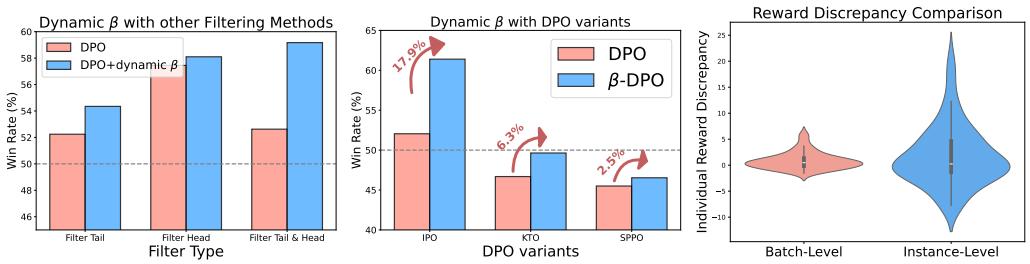


Figure 5: **Left:** Win rates from GPT-4 evaluations on Anthropic-HH single-turn dialogues, showcasing β-DPO's adaptability to diverse filtering strategies. **Middle:** Win rates of β-DPO across various DPO variants as evaluated by GPT-4. **Right:** Distribution of individual reward discrepancies following fine-tuning through batch-level and instance-level calibration.

Selective filtering of the top 20% of samples markedly enhances model performance. This approach, detailed in Figure 5 (Left), not only surpasses other filtering strategies but also suggests that these samples, which exhibit the smallest discrepancies between positive and negative pairs, are particularly prone to flipped noise. By excluding them, the model's learning efficacy is appreciably improved.

Dynamic β adapts to and improves upon existing filtering strategies. Figure 5 (Left) corroborates our stance that a static β proves insufficient within the DPO framework. We contend that the application of our dynamic β-DPO could markedly reshape the DPO field by fostering the development of advanced filtering techniques.

Dynamic β Enhancement across DPO Variants. We introduce dynamic β-DPO, a novel strategy enhancing DPO and its variants: IPO, KTO, and SPPO in Figure 5 (Middle). Our results on the Anthropic HH dataset demonstrate that while IPO initially leads in performance, the integration of dynamic β substantially elevates all variants, notably increasing β-IPO's efficiency by 17.9%. This underscores dynamic β-DPO's capability to significantly enhance model training through adaptable improvements, solidifying its value in advancing language models via human feedback.

5.3 Necessity of Batch-Level Dynamic β Calibration

In this section, we aim to underscore the pivotal role of batch-level tuning in calibrating the parameter β. To this end, we compare the performance of our β-DPO algorithm under two distinct regimes: one employing batch-level dynamic β calibration, and the other utilizing instance-level dynamics. To emulate the diverse data disparity scenarios encountered in practical applications, we adopt the methodology outlined in Section 4.1, meticulously blending datasets characterized by both *low gap* and *high gap* attributes at varying ratios.

Batch-level calibration surpasses both instance-level and population-level approaches. The results presented in Table 2 illustrate that batch-level dynamic β calibration yields superior performance compared to instance-level dynamics and the baseline population-level approach (referred to as vanilla DPO) across a range of mixture ratios. This improvement can be credited to the batch-level calibration's ability to adjust to the varying data quality present within a batch, thus refining the model's learning process. Conversely, instance-level dynamics can provoke excessively vigorous model updates, precipitating a decline in performance particularly noticeable at a mixture ratio of 40%, a scenario in which outliers exert a significant negative impact.

Instance-level calibration magnifies the impact of outliers. As demonstrated in Figure 5 (Right), instance-level calibration can inadvertently widen the range of reward discrepancy distribution. This broadened range suggests that instance-level calibration might be leading to excessively high or low β values for the model. Such disparities in β values consequently cause disproportionate update rates for certain samples, further intensifying the extremities in the distribution.

Our β-calibration strategy consistently outperforms baseline methods. To expand our approach to more diverse datasets and model sizes, we follow the current state-of-the-art models SimPO [26]. We perform β-DPO with two families of models, Llama3-8B [1], Mistral2-7B [21], on the UltraChat-200k

Table 2: Comparison of win rates across varying mixture ratios on the Anthropic HH dataset, with each ratio indicating the proportion of *high-gap* to *low-gap* datasets, e.g., a 40% mixture ratio reflects a blend of 40% *high-gap* and 60% *low-gap*.

Mixture Ratio	**10%**	**20%**	**30%**	**40%**
Vanilla DPO	50.17	50.56	47.95	29.15
+ Instance-level calibration	$49.18^{-1.97\%}$	$49.82^{-1.46\%}$	$44.42^{-7.36\%}$	$16.82^{-42.30\%}$
+ Batch-level calibration	$57.68^{+14.69\%}$	$56.15^{+11.06\%}$	$51.25^{+6.88\%}$	$34.92^{+19.79\%}$

dataset [11]. For comparison with baselines, we assess our models using one of the most popular open-ended instruction-following benchmarks: AlpacaEval 2. All settings are consistent with SimPO [26]. Table 3 presents the AlpacaEval 2 results under the Mistral-Instruct (7B) and Llama3-Instruct (8B) settings. LC and WR denote length-controlled and raw win rate, respectively. Regardless of whether we use Llama3-8B or Mistral-7B, and whether the loss function is DPO or SimPO, our β-D, SimPO strategy consistently demonstrates significant performance improvements. This thoroughly showcases the method's strong generalization ability and excellent scalability.

Table 3: Performance comparison of different models

Model	Mistral-Instruct (7B)		Llama3-Instruct (8B)	
	LC (%)	WR (%)	LC (%)	WR (%)
DPO	20.98	**21.60**	40.44	37.38
β-DPO	**23.56**	20.42	**43.38**	**38.21**
SimPO	28.50	30.56	44.38	38.97
β-SimPO	**30.48**	**32.13**	**46.03**	**40.18**

6 Conclusion and Future Work

This paper introduces β-DPO, a novel framework designed to optimize DPO by dynamically adjusting the β parameter in response to the variability in the informativeness of pairwise data. Our approach, which incorporates β-guided data filtering and batch-level dynamic β calibration, has demonstrated significant improvements in DPO's performance across a range of models and datasets. The empirical evaluations indicate that β-DPO offers an adaptable training paradigm for LLMs with human feedback.

Limitations and Future Work. Our work on β-DPO showcases a promising framework for LLM optimization, albeit with room for advancement. Future endeavors should explore: Adaptive β in Self-Play: Extending β-DPO to self-play scenarios [9, 43] where negative samples dynamically adapt, necessitating iterative β adjustments, to foster the evolution of superior model strategies. Enhanced Evaluation Standards: Development of sophisticated metrics and use of advanced evaluators beyond win rates, capitalizing on advancements like GPT-4+, to comprehensively gauge model performance. Scalability Investigation: Examining β-DPO's scalability to ultra-large models surpassing 7B parameters, and its integration into diverse DPO-inspired architectures, is pivotal for practical impact. Automated Parameter Tuning: Pursuing automation in parameter tuning, alleviating manual intervention for β, to streamline the training pipeline and broaden accessibility.

Acknowledgments and Disclosure of Funding

This research is supported by the National Science and Technology Major Project (2023ZD0121102), National Natural Science Foundation of China (92270114, 62302321). This research was also supported by the advanced computing resources provided by the Supercomputing Center of the USTC.

References

[1] AI@Meta. Llama 3 model card. 2024. URL https://github.com/meta-llama/llama3/blob/main/MODEL_CARD.md.

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

[3] Y. Bai, A. Jones, K. Ndousse, A. Askell, A. Chen, N. DasSarma, D. Drain, S. Fort, D. Ganguli, T. Henighan, N. Joseph, S. Kadavath, J. Kernion, T. Conerly, S. E. Showk, N. Elhage, Z. Hatfield-Dodds, D. Hernandez, T. Hume, S. Johnston, S. Kravec, L. Lovitt, N. Nanda, C. Olsson, D. Amodei, T. B. Brown, J. Clark, S. McCandlish, C. Olah, B. Mann, and J. Kaplan. Training a helpful and harmless assistant with reinforcement learning from human feedback. *CoRR*, abs/2204.05862, 2022.

[4] I. Bejan, A. Sokolov, and K. Filippova. Make every example count: On the stability and utility of self-influence for learning from noisy NLP datasets. In *EMNLP*, 2023.

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

[6] L. Bottou. Large-scale machine learning with stochastic gradient descent. In *COMPSTAT*, 2010.

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

[8] S. Bubeck, V. Chandrasekaran, R. Eldan, J. Gehrke, E. Horvitz, E. Kamar, P. Lee, Y. T. Lee, Y. Li, S. M. Lundberg, H. Nori, H. Palangi, M. T. Ribeiro, and Y. Zhang. Sparks of artificial general intelligence: Early experiments with GPT-4. *CoRR*, abs/2303.12712, 2023.

[9] Z. Chen, Y. Deng, H. Yuan, K. Ji, and Q. Gu. Self-play fine-tuning converts weak language models to strong language models. In *ICML*, 2024.

[10] P. F. Christiano, J. Leike, T. B. Brown, M. Martic, S. Legg, and D. Amodei. Deep reinforcement learning from human preferences. In *NeurIPS*, 2017.

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

[12] H. Dong, W. Xiong, D. Goyal, R. Pan, S. Diao, J. Zhang, K. Shum, and T. Zhang. RAFT: reward ranked finetuning for generative foundation model alignment. *CoRR*, abs/2304.06767, 2023.

[13] K. Ethayarajh, W. Xu, N. Muennighoff, D. Jurafsky, and D. Kiela. KTO: model alignment as prospect theoretic optimization. *CoRR*, abs/2402.01306, 2024.

[14] J. Fang, Z. Bi, R. Wang, H. Jiang, Y. Gao, K. Wang, A. Zhang, J. Shi, X. Wang, and T.-S. Chua. Towards neuron attributions in multi-modal large language models. In *Thirty-eighth Conference on Neural Information Processing Systems*, 2024.

[15] J. Fang, S. Zhang, C. Wu, Z. Yang, Z. Liu, S. Li, K. Wang, W. Du, and X. Wang. Moltc: Towards molecular relational modeling in language models. In *ACL (Findings)*, pages 1943–1958. Association for Computational Linguistics, 2024.

[16] L. Gao, J. Schulman, and J. Hilton. Scaling laws for reward model overoptimization. In *ICML*, volume 202 of *Proceedings of Machine Learning Research*, pages 10835–10866. PMLR, 2023.

[17] R. Ge, F. Huang, C. Jin, and Y. Yuan. Escaping from saddle points - online stochastic gradient for tensor decomposition. In *COLT*, 2015.

[18] A. Havrilla, M. Zhuravinskyi, D. Phung, A. Tiwari, J. Tow, S. Biderman, Q. Anthony, and L. Castricato. trlX: A framework for large scale reinforcement learning from human feedback. In *EMNLP*, Dec. 2023.

[19] J. Hong, N. Lee, and J. Thorne. ORPO: monolithic preference optimization without reference model. *CoRR*, abs/2403.07691, 2024.

[20] Z. Huang, H. Chen, Z. Wen, C. Zhang, H. Li, B. Wang, and C. Chen. Model-aware contrastive learning: Towards escaping the dilemmas. In *ICML*, 2023.

[21] A. Q. Jiang, A. Sablayrolles, A. Mensch, C. Bamford, D. S. Chaplot, D. de Las Casas, F. Bressand, G. Lengyel, G. Lample, L. Saulnier, L. R. Lavaud, M.-A. Lachaux, P. Stock, T. L. Scao, T. Lavril, T. Wang, T. Lacroix, and W. E. Sayed. Mistral 7B. *ArXiv*, abs/2310.06825, 2023.

[22] D. Jiang, X. Ren, and B. Y. Lin. LLM-Blender: Ensembling large language models with pairwise ranking and generative fusion. In *ACL*, 2023.

[23] A. Kukleva, M. Böhle, B. Schiele, H. Kuehne, and C. Rupprecht. Temperature schedules for self-supervised contrastive methods on long-tail data. In *ICLR*, 2023.

[24] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra. Continuous control with deep reinforcement learning. In *ICLR (Poster)*, 2016.

[25] T. Liu, Y. Zhao, R. Joshi, M. Khalman, M. Saleh, P. J. Liu, and J. Liu. Statistical rejection sampling improves preference optimization. *CoRR*, abs/2309.06657, 2023.

[26] Y. Meng, M. Xia, and D. Chen. Simpo: Simple preference optimization with a reference-free reward. *CoRR*, abs/2405.14734, 2024.

[27] T. Morimura, M. Sakamoto, Y. Jinnai, K. Abe, and K. Air. Filtered direct preference optimization. *arXiv preprint arXiv:2404.13846*, 2024.

[28] R. Nakano, J. Hilton, S. Balaji, J. Wu, L. Ouyang, C. Kim, C. Hesse, S. Jain, V. Kosaraju, W. Saunders, X. Jiang, K. Cobbe, T. Eloundou, G. Krueger, K. Button, M. Knight, B. Chess, and J. Schulman. Webgpt: Browser-assisted question-answering with human feedback. *CoRR*, abs/2112.09332, 2021.

[29] OpenAI. GPT-4 technical report. *CoRR*, abs/2303.08774, 2023.

[30] L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. L. Wainwright, P. Mishkin, C. Zhang, S. Agarwal, K. Slama, A. Ray, J. Schulman, J. Hilton, F. Kelton, L. Miller, M. Simens, A. Askell, P. Welinder, P. F. Christiano, J. Leike, and R. Lowe. Training language models to follow instructions with human feedback. In *NeurIPS*, 2022.

[31] G. Penedo, Q. Malartic, D. Hesslow, R. Cojocaru, H. Alobeidli, A. Cappelli, B. Pannier, E. Almazrouei, and J. Launay. The refinedweb dataset for falcon LLM: outperforming curated corpora with web data only. In *NeurIPS*, 2023.

[32] G. Pruthi, F. Liu, S. Kale, and M. Sundararajan. Estimating training data influence by tracing gradient descent. In *NeurIPS*, 2020.

[33] F. Pukelsheim. The three sigma rule. *The American Statistician*, 1994.

[34] R. Rafailov, A. Sharma, E. Mitchell, C. D. Manning, S. Ermon, and C. Finn. Direct preference optimization: Your language model is secretly a reward model. In *NeurIPS*, 2023.

[35] H. Robbins and S. Monro. A stochastic approximation method. 1951.

[36] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov. Proximal policy optimization algorithms. *CoRR*, abs/1707.06347, 2017.

[37] N. Stiennon, L. Ouyang, J. Wu, D. M. Ziegler, R. Lowe, C. Voss, A. Radford, D. Amodei, and P. F. Christiano. Learning to summarize from human feedback. *CoRR*, abs/2009.01325, 2020.

[38] H. Touvron, L. Martin, K. Stone, P. Albert, A. Almahairi, Y. Babaei, N. Bashlykov, S. Batra, P. Bhargava, S. Bhosale, D. Bikel, L. Blecher, C. Canton-Ferrer, M. Chen, G. Cucurull, D. Esiobu, J. Fernandes, J. Fu, W. Fu, B. Fuller, C. Gao, V. Goswami, N. Goyal, A. Hartshorn, S. Hosseini, R. Hou, H. Inan, M. Kardas, V. Kerkez, M. Khabsa, I. Kloumann, A. Korenev, P. S. Koura, M. Lachaux, T. Lavril, J. Lee, D. Liskovich, Y. Lu, Y. Mao, X. Martinet, T. Mihaylov, P. Mishra, I. Molybog, Y. Nie, A. Poulton, J. Reizenstein, R. Rungta, K. Saladi, A. Schelten, R. Silva, E. M. Smith, R. Subramanian, X. E. Tan, B. Tang, R. Taylor, A. Williams, J. X. Kuan,

P. Xu, Z. Yan, I. Zarov, Y. Zhang, A. Fan, M. Kambadur, S. Narang, A. Rodriguez, R. Stojnic, S. Edunov, and T. Scialom. Llama 2: Open foundation and fine-tuned chat models. *CoRR*, abs/2307.09288, 2023.

[39] M. Völske, M. Potthast, S. Syed, and B. Stein. Tl;dr: Mining reddit to learn automatic summarization. In *NFiS@EMNLP*, 2017.

[40] C. Wang, Y. Jiang, C. Yang, H. Liu, and Y. Chen. Beyond reverse KL: generalizing direct preference optimization with diverse divergence constraints. *ICLR*, 2024.

[41] H. Wang, W. Xiong, T. Xie, H. Zhao, and T. Zhang. Interpretable preferences via multi-objective reward modeling and mixture-of-experts. *arXiv preprint arXiv:2406.12845*, 2024.

[42] J. Wu, Y. Xie, Z. Yang, J. Wu, J. Chen, J. Gao, B. Ding, X. Wang, and X. He. Towards robust alignment of language models: Distributionally robustifying direct preference optimization. *CoRR*, abs/2407.07880, 2024.

[43] Y. Wu, Z. Sun, H. Yuan, K. Ji, Y. Yang, and Q. Gu. Self-play preference optimization for language model alignment. *arXiv preprint arXiv:2405.00675*, 2024.

[44] M. Xia, S. Malladi, S. Gururangan, S. Arora, and D. Chen. Less: Selecting influential data for instruction tuning. 2024.

[45] Z. Yuan, H. Yuan, C. Tan, W. Wang, S. Huang, and F. Huang. RRHF: rank responses to align language models with human feedback without tears. In *NeurIPS*, 2023.

[46] O. Zhang, M. Wu, J. Bayrooti, and N. D. Goodman. Temperature as uncertainty in contrastive learning. *CoRR*, abs/2110.04403, 2021.

[47] Y. Zhao, R. Joshi, T. Liu, M. Khalman, M. Saleh, and P. J. Liu. Slic-hf: Sequence likelihood calibration with human feedback. *CoRR*, abs/2305.10425, 2023.

[48] C. Zhou, P. Liu, P. Xu, S. Iyer, J. Sun, Y. Mao, X. Ma, A. Efrat, P. Yu, L. Yu, S. Zhang, G. Ghosh, M. Lewis, L. Zettlemoyer, and O. Levy. LIMA: less is more for alignment. In *NeurIPS*, 2023.

A Experiment

A.1 β-DPO Implementation Details and Hyperparameters

β-DPO is relatively straightfoward to implement; The full algorithm is summarized in Algorithm 1.

Algorithm 1 β-Direct Preference Optimization

Require: Preference dataset $\mathcal{D}$, batch size b, constraint coefficient β_0, selection ratio ρ, scaling factor α, and learning rate η.

Initialize model π_{θ_0} with supervised finetuning on $\mathcal{D}$.
while not converged **do**
  Sample a batch $\mathcal{B} = \{(\mathbf{x}^{(i)}, \mathbf{y}_w^{(i)}, \mathbf{y}_l^{(i)})\}_{i=1}^{b}$ from $\mathcal{D}$.
  Compute the individual reward discrepancy $M_i = r(\mathbf{y}_w^{(i)}; \mathbf{x}^{(i)}) - r(\mathbf{y}_l^{(i)}; \mathbf{x}^{(i)})$.
  Update the threshold M_0 and σ using Equations (7) and (9).
  Sample $b \times \rho$ samples without replacement based on $p(M_i)$ in Equation (8).
  Compute the batch-level β using Equation (5).
  Compute the loss using the Equation (4).
  Compute the gradient and update the model $\theta_t \leftarrow \theta_{t-1} - \eta \nabla_\theta \ell(\theta_{t-1}, \mathcal{B})$.
end while
return Final model π_θ.

Unless noted otherwise, we use a $\beta = 0.1$, batch size of 64, $m = 0.9$ to ensure the stability of the global M_i estimation, $\rho = 0.8$ to filter 20% uninformative samples, and the Adam optimizer with a learning rate $5e-7$ by default. We carried out all computational tasks on a suite of four 80GB A100 GPUs. For the Pythia-410M model, we use a batch size of 128, while the rest of the parameters remain the same.

In examining the arena of **single-turn dialogue**, our experimental framework leverages Pythia-2.8, Pythia-1.4b, and Pythia-410M for empirical analysis using the Anthropic-HH dataset. Given the absence of a pre-existing Supervised Fine-Tuning (SFT) model tailored for this dataset, we fine-tune an accessible language model exclusively with preferred completions to sculpt our SFT model. Turning our focus to the domain of **summarization**, we employed the Reddit TL;DR summarization dataset, enriching our research with human preferences as documented by the study [37]. Our methodology here incorporates an SFT model meticulously fine-tuned on expert-composed summaries of forum posts [3], operating within the TRLX framework for Reinforcement Learning from Human Feedback (RLHF), as introduced by Havrilla et al. [18]. The human preference dataset was gathered by Stiennon et al. [37] on samples from a different, but similarly-trained, SFT model.

A.2 Mixture of *low gap* and *high gap*

In our previous discussion, we identified a *low gap* dataset, constituted by pairs of responses generated from the HH dataset. Given that these responses originate from the same model, we can infer that they represent semantically similar answers, hence the designation *low gap*. Concurrently, we maintain the positive samples constant while selecting negative samples generated by a Pythia 2.8B model. The significant performance disparity between the models results in a considerable variation in the quality of these negative samples, which we refer to as the *high gap*. We mix these two types of data in varying proportions to mimic the distribution of data in real-world scenarios. We label this approach the "mixture experiment." To facilitate a better comparison of the distributions across different ratios, we proceed to illustrate our findings with the following graph:

Figure 6 clearly illustrates that as the mixture ratio increases—that is, the proportion of high gap data rises—the dispersion of the overall dataset broadens. Conversely, a decrease in the mixture ratio, corresponding to an elevated presence of low gap data, results in a more concentrated distribution of reward discrepancy.

[3] https://huggingface.co/CarperAI/openai_summarize_tldr_sft

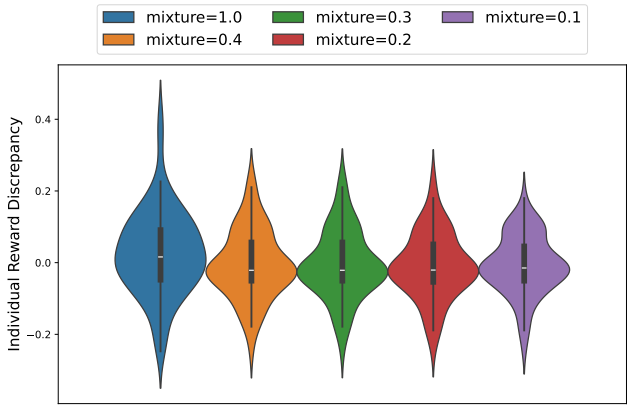


Figure 6: Distribution of individual reward discrepancies following the Pythia-2.8B model.

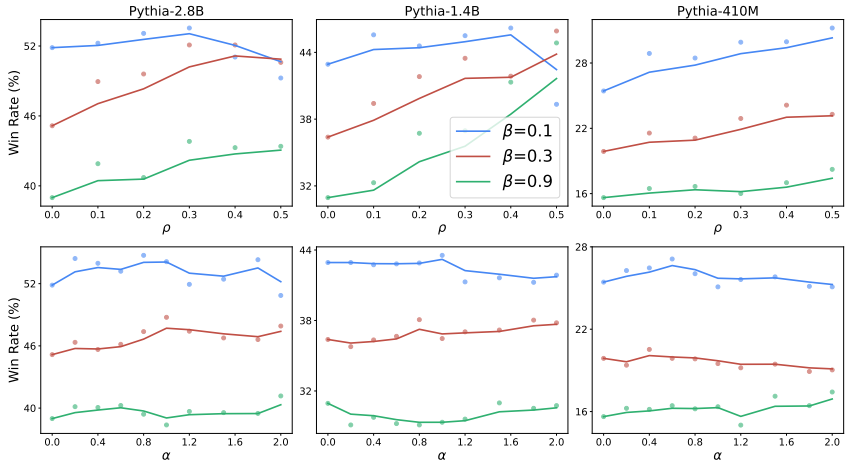


Figure 7: Performance comparison across different β values and ρ values for three different model sizes (Pythia-2.8B, Pythia-1.4B, and Pythia-410M) on the Anthropic HH dataset using GPT-4 as the evaluator. Each subplot represents the win rate for varying parameters β = 0.1, 0.3, and 0.9 with exponential smoothing.

A.3 Hyperparameter Sensitivity

In this section, we investigate the impact of key hyperparameters in the β-DPO methodology, using the Anthropic HH dataset and leveraging models such as Pythia-2.8B and GPT-4 for evaluation. Specifically, we examine the influence of varying the hyperparameters α and the filtering ratio ρ. The parameter α is explored across a broad spectrum of values including 0.2, 0.4, 0.6, 0.8, 1.0, 1.2, 1.5, 1.8, and 2.0. Concurrently, the filtering ratio ρ is investigated at intervals ranging from 0.1 to 0.5, at a granularity of 0.1. This comprehensive analysis aims to unravel how these hyperparameters affect the performance and outcomes of the β-DPO process.

Static β vs. Dynamic β. Our results, as depicted in Figure 7, reveal that a dynamic β (where $\alpha \neq 0$) prevails over a static β (where $\alpha = 0$) under an exhaustive range of hyperparameter configurations. This outcome aligns seamlessly with our underlying premise: static beta fails to fully leverage a model's potential when confronted with varying data quality within a dataset. Furthermore, our observations highlight an intriguing trend: as α increases, the performance of the β-DPO model initially improves before declining, typically peaking within the interval of 0.6 to 1.0.

Filtering Ratio ρ Sensitivity. As illustrated in the figure 7, varying model sizes exhibit distinct sensitivities to the parameter ρ. Within the context of the Pythia-2.8B model, a ρ value of 0.3 yields optimal performance, whereas for the Pythia-410M model, a ρ value of 0.5 is superior. This can be posited to suggest that smaller models may require more stringent data filtering to perform optimally, whereas larger models possess the increased capacity necessary for learning the most

effective strategies. This insight provides a significant directive for future research: the value of ρ should be fine-tuned according to the specific circumstances of the application.

A.4 The ablation study *w.r.t.* M_0

In this work, we employed a moving average updating scheme [24] for the updating of M_0. In order to investigate the superiority of this configuration, we also conducted a comparative experiment involving hyperparameter settings. Specifically, M_0 was treated as a constant in the training process, while all other settings remained unchanged. The experimental results are as follows:

Table 4: Comparison of win rates across varying M_0 in β-DPO.

M_0	0	1	3	5	7	10	Moving Average
β-DPO	53.35	54.00	53.85	55.61	53.19	56.75	57.07

As demonstrated in Table 4, by tuning M_0, we were able to achieve significant performance improvements, which approached the performance level of the moving average updating scheme. This clearly underscores the superiority of the moving average updating scheme. On one hand, it obviates the need for an additional manual search process. On the other hand, it facilitates stable performance enhancements.

A.5 Our Methods with Explicit Reward Model

To broaden the application of the β-DPO strategy for alignment and to address the limitations of DPO's implicit reward model, we incorporated two external reward models (RMs):

- We utilized llm-blender/PairRM [22] to score the chosen and rejected responses, denoted as $r_{w,\text{PairRM}}, r_{l,\text{PairRM}}$.
- We adopted the v0.2 Llama3-Instruct setting [26] by employing RLHFlow/ArmoRM-Llama3-8B-v0.1 [41] as the reward model to rank responses, denoted as $r_{w,\text{ArmoRM}}, r_{l,\text{ArmoRM}}$.

The other algorithmic processes (refer to Appendix Algorithm 1) remain unchanged, with modifications made only to $M_{i,\text{PairRM}} = r^{(i)}_{w,\text{PairRM}} - r^{(i)}_{l,\text{PairRM}}$ and $M_{i,\text{ArmoRM}} = r^{(i)}_{w,\text{ArmoRM}} - r^{(i)}_{l,\text{ArmoRM}}$. For comparison with baselines, we assess our models using one of the most popular open-ended instruction-following benchmarks: AlpacaEval 2. The detailed results are as follows:

Table 5: Comparison of different methods on Llama3-Instruct (8B) with explicit reward model

Method	**Llama3-Instruct (8B)**	**Llama3-Instruct (8B)**
	LC (%)	**WR (%)**
DPO (Implicit RM)	40.44	37.38
β-DPO (Implicit RM)	**43.38**	**38.21**
SimPO (Implicit RM)	44.38	38.97
β-SimPO (Implicit RM)	**46.03**	**40.18**
SimPO (PairRM)	44.70	38.98
β-SimPO (PairRM, Instance-Level)	43.84	38.54
β-SimPO (PairRM, Batch-Level)	**45.65**	**39.76**
SimPO (ArmoRM)	53.70	47.50
β-SimPO (ArmoRM, Instance-Level)	49.05	45.47
β-SimPO (ArmoRM, Batch-Level)	**54.86**	**49.66**

From the results above, we observed the following:

The proposed dynamic β strategy demonstrates strong generalizability. Both β-D,SimPO consistently yield stable performance improvements across implicit and explicit RMs.

Batch-level performance is crucial. In explicit RM settings, batch-level dynamic β consistently outperforms instance-level dynamic β.

A.6 GPT-4 prompts for computing summarization and dialogue win rates

A fundamental element of our experimental framework involves the assessment of win rates facilitated by GPT-4. In this segment, we delineate the prompts employed to ascertain win rates pertinent to our summarization and dialogue-oriented investigations. For the entirety of our experimental endeavors, we utilize GPT-4. It is important to note that the sequence in which summaries or responses are presented is randomized for each evaluation.

Summarization GPT-4 win rate prompt.

```
Which of the following summaries does a better job of summarizing the most \
important points in the given forum post?

Post:
<post>

Summary A:
<Summary A>

Summary B:
<Summary B>

FIRST provide a one-sentence comparison of the two summaries, explaining which \
you prefer and why. SECOND, on a new line, state only "A" or "B" to indicate your \
choice. Your response should use the format:
Comparison: <one-sentence comparison and explanation>
Preferred: <"A" or "B">
```

Dialogue GPT-4 win rate prompt.

```
For the following query to a chatbot, which response is more helpful?

Query: <the user query>

Response A:
<either the test method or baseline>

Response B:
<the other response>

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

B Broader Impacts

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none of which we feel must be specifically highlighted here.

NeurIPS Paper Checklist

1. **Claims**

Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

Answer: [Yes]

Justification: see abstract and introduction.

Guidelines:

- The answer NA means that the abstract and introduction do not include the claims made in the paper.
- The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
- The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
- It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

Question: Does the paper discuss the limitations of the work performed by the authors?

Answer: [Yes]

Justification: see Section 6.

Guidelines:

- The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [NA]

Justification: This work does not include theoretical results.

Guidelines:

- The answer NA means that the paper does not include theoretical results.

- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: see Appendix A.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: The code is available at https://github.com/junkangwu/beta-DPO.

   Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: see Appendix A.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: The results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.

- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We carried out all computational tasks on a suite of four 80GB A100 GPUs.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics https://neurips.cc/public/EthicsGuidelines?

   Answer: [Yes]

   Justification: Our research has been conducted with strict adherence to the NeurIPS Code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: see Appendix B

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.

- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: The paper poses no such risks.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [NA]

    Justification:The answer NA means that the paper does not use existing assets.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: The answer NA means that the paper does not release new assets.

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.