ADAPTIVE MARGIN RLHF VIA PREFERENCE OVER PREFERENCES

Anonymous authors
Paper under double-blind review

ABSTRACT

Margin-based optimization is fundamental to improving generalization and robustness in classification tasks. In the context of reward model learning from preferences within Reinforcement Learning from Human Feedback (RLHF), existing methods typically rely on no margins, fixed margins, or margins that are simplistic functions of preference ratings. However, such formulations often fail to account for the varying strengths of different preferences—i.e., some preferences are associated with larger margins between responses—or they rely on noisy margin information derived from preference ratings. In this work, we argue that modeling the strength of preferences can lead to better generalization and more faithful alignment. Furthermore, many existing methods that use adaptive margins assume access to accurate preference scores, which can be difficult for humans to provide reliably. We propose a novel approach that leverages preferences over preferences—that is, annotations indicating which of two preferences reflects a stronger distinction. We use this ordinal signal to infer adaptive margins on a per-datapoint basis. We introduce an extension to Direct Preference Optimization (DPO), DPO-PoP, that incorporates adaptive margins from preference-over-preference supervision, enabling improved discriminative and generative performance. Empirically, our method outperforms vanilla DPO, DPO with fixed margins, and DPO with ground-truth margins on the UltraFeedback dataset. These results suggest that integrating preference-over-preference information, which requires less precision to be provided accurately, can improve discriminative and generative performance without adding significant complexity. Additionally, we show that there is a tradeoff between discriminative and generative performance: improving test classification accuracy, particularly by correctly labeling weaker preferences at the expense of stronger ones, can lead to a decline in generative quality. To navigate this tradeoff, we propose two sampling strategies to gather preference-over-preference labels: one favoring discriminative performance and one favoring generative performance.

1 INTRODUCTION

Margin-based approaches have been pivotal in the design and analysis of classification algorithms. In classical machine learning, the margin, defined as the distance between a decision boundary and data points, acts as a proxy for confidence and plays a critical role in improving generalization. For example, Support Vector Machines (SVMs) explicitly maximize the minimum margin, which has been shown to enhance robustness and reduce overfitting (Cortes & Vapnik, 1995). Ensemble methods like AdaBoost (Freund et al., 1996) also leverage margin-based generalization, as boosting algorithms implicitly seek to increase the margin distribution across training samples (Schapire et al., 1998).

Although fixed-margin strategies have proven effective, they assume fixed and equal margin for all training data points. This has motivated the development of adaptive margin approaches, where the margin varies across examples based on criteria such as sample difficulty, uncertainty, or class imbalance. Adaptive Margin SVMs (Herbrich & Weston, 1999) use different margin values for different training data points and provide bounds on the generalization error, justifying its robustness against outliers. Furthermore, methods such as CurricularFace (Huang et al., 2020), AdaCos (Zhang et al., 2019), and adaptive triplet losses (Ha & Blanz, 2021) have shown that adapting the margin

dynamically during training leads to more stable optimization and better generalization, particularly in settings such as face recognition or imbalanced classification.

In Reinforcement Learning from Human Feedback (RLHF), pairwise preference data from humans is used to learn a reward function or policy. The Bradley-Terry (BT) model (Bradley & Terry, 1952) is widely used to model pairwise preference data, where the probability of preferring one output over another is determined by the difference in their reward scores. This preference model is commonly used in the alignment of large language models (LLMs) (Ouyang et al., 2022; Touvron et al., 2023), in which a reward function is learned to rank outputs based on human preferences, and subsequently used to optimize the policy.

Current reward modeling approaches generally fall into two categories. Some methods treat all preferences equally by applying no margin at all (Ouyang et al., 2022). Others incorporate unequal treatment by introducing adaptive margins, which are typically derived in one of two ways: either from scalar scores assigned to preferences by human annotators or language models (Touvron et al., 2023; Wang et al., 2025), or from the outputs of learned reward models (Wang et al., 2024a; Qin et al., 2024; Amini et al., 2024; Wang et al., 2024b). Using constant or no margin information fails to account for the varying strength of different preferences—that is, the degree to which one response is favored over another within a given preference. Obtaining preference strength information from preference scores, allows us to use adaptive margin information, but requires us to collect scalar feedback from LLMs or humans.

Specifying preference strength typically requires a numerical score, which may be difficult for humans to provide accurately. For instance, when using labeling schemes such as Likert ratings, where annotators rate responses individually rather than comparatively, the scores may not be consistently calibrated. That is, even if annotators agree on which response is better in a pair, they may assign inconsistent scores due to differences in how they interpret the scale (Wadhwa et al., 2024). By contrast, preference over preference annotation requires less precision to be provided accurately, compared to assigning scores to individual responses. Comparitive annotation, particularly Best-to-Worst scaling (BWS), has been to shown to produce significantly more reliable results than rating scale annotations such as Likert scales (Kiritchenko & Mohammad, 2017; Burton et al., 2019). BWS also demonstrated greater reliability when applied to linguistically complex cases, such as phrases containing negation or modals (Kiritchenko & Mohammad, 2017). Best-to-Worst scaling (BWS) is an extension of Thurstone's method of paired comparisons (Thurstone, 2017) which is another paired comparison statistical model like Bradley-Terry (Bradley & Terry, 1952; Handley, 2001) We use this as a motivation to propose preference over preference (PoP) labeling, in which annotators compare two preferences and indicate which one reflects a stronger preference. Rather than assigning scores to individual responses (Cui et al., 2024; Wang et al., 2023), in our preference-over-preference setting, annotators compare preference pairs and select the pair for which the contrast between the chosen and rejected responses is more pronounced. More importantly, preference-over-preferences allow us to infer continuous real-valued margins for preferences, compared to rating scale annotations, which only offer discrete numerical options. Using this PoP supervision, we construct a dataset of preference over preference comparisons that enables us to infer adaptive margin information for each datapoint.

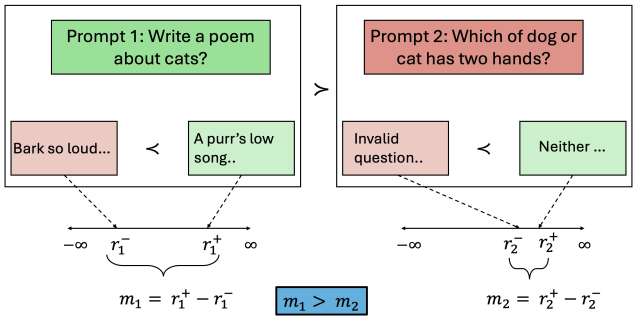


Figure 1: A pictorial illustration of the PoP framework. A preference is stronger than another when the reward difference between its preferred and dispreferred responses is larger. The reward difference of the weaker preference in the pair serves as the margin for the stronger preference.

In this work, we propose DPO-PoP, an alignment algorithm that integrates preference-over-preference (PoP) supervision into the Direct Preference Optimization (DPO) framework (Rafailov et al., 2024b), enabling margin-aware alignment of large language models (LLMs) with human preferences using only supervised learning. For each data point, we use PoP supervision to infer an adaptive margin

that reflects the relative strength of the underlying preference. A pictorial illustration of the PoP framework is presented in Figure 1. We demonstrate that collecting PoP supervision is a simple and effective way to improve both the discriminative and generative performance of LLMs. Our results show that DPO-PoP variants outperform all baselines in both respects. Moreover, we highlight a tradeoff between discriminative performance, as measured by test classification accuracy, and generative performance, as measured by win rate—where improving classification accuracy on weaker preferences at the expense of stronger ones—can lead to a decline in generative quality. To navigate this tradeoff, we propose two sampling strategies for generating preference-over-preference labels: iterative sampling, which favors discriminative performance, and random sampling, which favors generative performance.

2 BACKGROUND

2.1 REWARD MODELING

In the reward modeling stage of Reinforcement Learning from Human Feedback (RLHF), a reward model is trained to assign scalar scores to prompt-response pairs, indicating how well a response aligns with human preferences. This process relies on a preference dataset $\mathcal{D}_{\text{pref}} = (x_i, y_i^+, y_i^-)_{i=1}^N$, where x_i is a prompt, y_i^+ is the preferred response, and y_i^- is the dispreferred response. The Bradley-Terry (BT) model (Bradley & Terry, 1952) is commonly used to model preference likelihoods.

$$P(y^+ \succ y^-) = \frac{e^{r(x,y^+)}}{e^{r(x,y^+)} + e^{r(x,y^-)}} = \sigma(r(x,y^+) - r(x,y^-)) \tag{1}$$

Here, r denotes the reward assigned to a prompt-response pair, and σ denotes the sigmoid function. We parameterize the reward function as r_ϕ, and use it to approximate the ground-truth reward function by maximizing the likelihood of the observed preference data under the Bradley-Terry model. For more details on the RLHF pipeline, refer to Appendix C

$$\min_\phi -\mathbb{E}_{(x,y^+,y^-)\sim\mathcal{D}_{\text{pref}}}[\log \sigma(r_\phi(x,y^+) - r_\phi(x,y^-))] \tag{2}$$

2.2 DIRECT PREFERENCE OPTIMIZATION

Direct Preference Optimization (DPO) (Rafailov et al., 2024b) belongs to a class of algorithms, called Direct Alignment Algorithms (DAAs) (Rafailov et al., 2024a), which aim to directly align a policy from preference data via supervised learning, without having to learn a reward model or use reinforcement learning. DPO utilizes the closed form solution of the optimal KL regularized reward policy (Peters & Schaal, 2007; Peng et al., 2019), and expresses the rewards in the Bradley-Terry preference model (Bradley & Terry, 1952), directly in terms of the optimal policy. This allows us to learn a parameterized optimal policy directly from the preference data, using Equation 3

$$\mathcal{L}_{DPO}(\pi_\theta; \pi_{ref}) = \mathbb{E}_{(x,y^+,y^-)\sim\mathcal{D}_{\text{pref}}}\left[-\log\sigma\left(\beta\log\frac{\pi_\theta(y^+|x)}{\pi_{\text{ref}}(y^+|x)} - \beta\log\frac{\pi_\theta(y^-|x)}{\pi_{\text{ref}}(y^-|x)}\right)\right] \tag{3}$$

The implicit reward assigned by the DPO model to a response y given a prompt x is $\beta\log\frac{\pi_\theta(y|x)}{\pi_{\text{ref}}(y|x)}$.

2.3 MARGINS IN REWARD MODELING

Margins can be incorporated into the reward modeling phase of the RLHF pipeline to enforce not only that the reward model ranks the preferred response higher than the dispreferred one, but also that it assigns a sufficiently large difference in reward scores—either through fixed or adaptive margins.

The margin-based reward modeling loss can be expressed as:

$$\min_\phi -\mathbb{E}_{(x,y^+,y^-)\sim\mathcal{D}_{\text{pref}}}[\log \sigma(r_\phi(x,y^+) - r_\phi(x,y^-) - m(x,y^+,y^-)] \tag{4}$$

Here $m(x, y^+, y^-)$ denotes the margin term. In the fixed margin setting this can be a constant. In the adaptive-margin setting, it can be defined as a function of the preference instance, for example, based on the degree of discrepancy between the preferred and dispreferred responses.

3 Method: Adaptive Margin DPO with Preferences over Preferences

To obtain adaptive margin information, in which each preference datapoint is assigned a different margin, and stronger preferences are associated with larger margins than weaker ones, we propose preferences over preferences (PoP) supervision. Given two standard preference comparisons, such as $A \succ B$ and $C \succ D$, we collect a label indicating which of the two preferences is stronger, from a labeler. For example, if the supervision indicates that $(A \succ B) \succ (C \succ D)$, this means that the discrepancy between A and B is greater than that between C and D under the ground-truth reward function r. Formally, this implies:

$$r(A) - r(B) > r(C) - r(D)$$

This insight allows us to treat the margin from the weaker preference (e.g., $r(C) - r(D)$) as a lower bound on the margin for the stronger preference (e.g., $A \succ B$). Rather than regressing to a specific value, we enforce that the margin for the stronger preference must be at least as large as that of the weaker one.

We assume access to a dataset of preference over preference examples:

$$\mathcal{D}_{\text{PoP}} = \left\{ \left((x_{s_i}, y^+_{s_i}, y^-_{s_i}),\ (x_{w_i}, y^+_{w_i}, y^-_{w_i}) \right) \right\}_{i=1}^{N}$$

Here, $(x_{s_i}, y^+_{s_i}, y^-_{s_i})$ represents the stronger preference in the pair, where x_{s_i} is the prompt, $y^+_{s_i}$ is the preferred response, and $y^-_{s_i}$ is the dispreferred response. Similarly, $(x_{w_i}, y^+_{w_i}, y^-_{w_i})$ denotes the weaker preference, where x_{w_i} is the prompt, $y^+_{w_i}$ is the preferred response, and $y^-_{w_i}$ is the dispreferred response. Note that, unlike in standard reward modeling datasets, the prompts x_{s_i} and x_{w_i} can differ within a single PoP example, as PoP supervision compares the strength of entire preference instances, not individual responses.

We can express the adaptive margin reward modelling objective on a dataset of preferences over preferences as follows

$$\min_{\phi} \ \mathbb{E}_{\mathcal{D}_{\text{PoP}}} \Big[-\log \sigma \big(r_\phi(x_s, y^+_s) - r_\phi(x_s, y^-_s) \\ - \text{sg}\left[r_\phi(x_w, y^+_w) - r_\phi(x_w, y^-_w) \right] \big) \Big] \tag{5}$$

Here, sg$[\cdot]$ denotes the stop-gradient operator. Although the adaptive margin is computed using the reward model r_ϕ, we treat the margin derived from the weaker preference as a *fixed reference* during optimization. Applying the stop-gradient operator ensures that gradients do not propagate through this margin term, thereby preventing it from influencing updates to the reward model parameters ϕ. Without the stop-gradient operator, the objective would incentivize parameters that invert the weaker preference to minimize the loss.

We use the closed-form solution for the optimal policy of a KL regularized reward problem to express the rewards directly in terms of the optimal policy, as in DPO (Rafailov et al., 2024b). Parameterizing the optimal policy by θ, we end up with the DPO Preference-over-Preference loss

$$\min_{\theta} \ \mathbb{E}_{\mathcal{D}_{\text{PoP}}} \Bigg[-\log \sigma \Bigg(\beta \left(\log \frac{\pi_\theta(y^+_s \mid x_s)}{\pi_{\text{ref}}(y^+_s \mid x_s)} - \log \frac{\pi_\theta(y^-_s \mid x_s)}{\pi_{\text{ref}}(y^-_s \mid x_s)} \right) \\ - \text{sg} \left[\beta \left(\log \frac{\pi_\theta(y^+_w \mid x_w)}{\pi_{\text{ref}}(y^+_w \mid x_w)} - \log \frac{\pi_\theta(y^-_w \mid x_w)}{\pi_{\text{ref}}(y^-_w \mid x_w)} \right) \right] \Bigg) \Bigg] \tag{6}$$

The DPO Preference-over-Preference (DPO-PoP) objective enables margin-aware alignment directly from PoP data using supervised learning, without requiring an explicit reward modeling stage or reinforcement learning. However, Equation 6 suffers from unstable gradients due to unbounded margins, resulting in a rapidly fluctuating loss that can explode during training. To mitigate this, we clip the margin values to lie within a fixed interval $[0, M_{\text{max}}]$, where M_{max} is a user-specified constant. Margin values outside this range are clipped to the nearest endpoint, using a clipping function $\text{clip}_{[0,M_{\text{max}}]}$, which improves optimization stability. Additionally, to further stabilize training, we compute the margins using a slowly-updated target policy $\pi_{\hat{\theta}}$, whose parameters $\hat{\theta}$ track the policy π via Polyak averaging over the model parameters θ. This prevents the margin estimates from changing too rapidly across training steps. With these modifications, our final DPO-PoP objective is given by Equation 7

$$\min_{\theta} \mathbb{E}_{\mathcal{D}_{\text{PoP}}} \Bigg[-\log \sigma \Big(\beta \left(\log \frac{\pi_\theta(y_s^+ \mid x_s)}{\pi_{\text{ref}}(y_s^+ \mid x_s)} - \log \frac{\pi_\theta(y_s^- \mid x_s)}{\pi_{\text{ref}}(y_s^- \mid x_s)} \right) - \text{sg} \left[\text{clip}_{[0,M_{max}]} \left(\beta \left(\log \frac{\pi_{\hat{\theta}}(y_w^+ \mid x_w)}{\pi_{\text{ref}}(y_w^+ \mid x_w)} - \log \frac{\pi_{\hat{\theta}}(y_w^- \mid x_w)}{\pi_{\text{ref}}(y_w^- \mid x_w)} \right) \right) \right] \Big) \Bigg] \quad (7)$$

4 RESULTS

We focus on the following research questions: **[Q1]** Does using DPO-PoP lead to models with improved discriminative ability? **[Q2]** Does using DPO-PoP lead to models with improved generative ability? We investigate these questions by evaluating the performance of our models on the test split of the UltraFeedback dataset (Cui et al., 2024) and external benchmarks such as RewardBench (Lambert et al., 2024) and AlpacaEval-2 (Dubois et al., 2025).

4.1 GENERATING THE PREFERENCE OVER PREFERENCE DATA

We use the UltraFeedback (Cui et al., 2024) binarized dataset [1] to evaluate our research questions. This dataset provides scalar scores for each of the chosen and rejected responses in the dataset. These scores are computed by aggregating the scores from the feedback of multiple LLMs across different evaluation axes. We compute the ground-truth margin for each preference datapoint as the difference between the scores of the chosen and rejected responses. These margins can also be used to construct our preference over preference dataset. Ideally, we would want to use human feedback for generating preferences over preferences. But, gathering annotator labels for an entire dataset is an expensive task. Hence, we use the scores for the responses in the UltraFeedback dataset to simulate annotator preferences, and generate our synthetic PoP dataset.

Given a preference dataset of size $|\mathcal{D}_{\text{pref}}|$, we can create a PoP dataset of size up to $|\mathcal{D}_{\text{PoP}}| = \frac{|\mathcal{D}_{\text{pref}}|(|\mathcal{D}_{\text{pref}}|-1)}{2}$, which is significantly larger than the size of the original preference dataset itself. Instead, we choose a size of the PoP dataset that is a small multiple of the size of the preference dataset i.e $|\mathcal{D}_{\text{PoP}}| = k|\mathcal{D}_{\text{pref}}|$, so that the size of the PoP dataset does not become too large. Additional experiments showing how performance varies with k are provided in Appendix E. To avoid constructing nearly indistinguishable preference pairs, we ensure that the margin difference between the stronger and weaker preferences in the preference pair in the PoP dataset is at least one. This is consistent with annotation practices, where preference judgments are typically withheld when the comparative strength is marginal.

We evaluate two strategies for constructing the PoP dataset: one that represents each preference from the original dataset equally, and one that represents preferences in proportion to preference strength. We do this to explore the impact of different sampling strategies used to generate the PoP dataset, on downstream discriminative and generative performance. In the **iterative sampling** approach, each preference data point is equally represented by comparing it against k weaker preferences (as judged by their margins). In practice, without ground-truth margin data, we could choose a preference and provide comparison preferences, asking the user for a label. We only choose k preference pairs in

[1] `HuggingFaceH4/ultrafeedback_binarized`

which our chosen preference is judged to be stronger than the comparative preference. In contrast, the **random sampling** approach constructs the PoP dataset by randomly selecting pairs of preferences and labeling them based on their margins. This results in stronger preferences appearing more frequently in the PoP dataset than weaker ones. Furthermore, the **random sampling** approach is straightforward to implement in practice, in comparison to the iterative sampling approach, as this would only involve randomly sampling pairs of preferences and asking the annotator for a label. After generating the PoP dataset, we discard the original scalar scores and do not use them at any stage of model training. Additional experiments showing how performance of DPO-PoP algorithms is impacted by preference-over-preference labeling noise are provided in Appendix F

4.2 Experimental Setup

We consider two models in our experiments: LLama3.2-3B and LLama3.1-8B (Grattafiori et al., 2024). Following the standard direct alignment pipeline, we align these models using the UltraFeedback preference dataset (Cui et al., 2024). We begin with a pretrained model and fine-tune it on the supervised fine-tuning (SFT) partition of the UltraFeedback dataset. Next, we align the models using the preference data from the same dataset. For further experimental details, refer to Appendix B

We evaluate the following variants of Direct Preference Optimization (DPO):

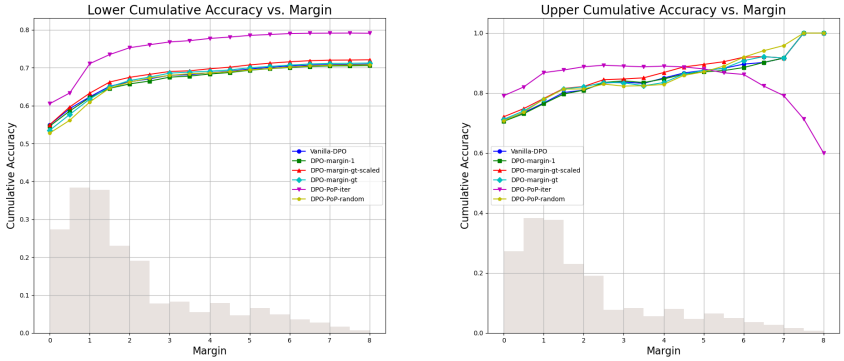
1. **Vanilla DPO**: No margin is used in the loss function.
2. **DPO-margin-1**: A fixed margin of 1 is applied to all preferences.
3. **DPO-margin-gt**: Ground-truth margin values from the UltraFeedback dataset are used.
4. **DPO-margin-gt-scaled**: This corresponds to the Scaled Bradley-Terry loss from Wang et al. (2025). The loss incorporates ground-truth margin information outside the log-sigmoid function rather than inside, effectively placing greater weight on preferences with larger margins. This can be interpreted as repeatedly sampling stronger preferences. The loss is defined as:
$$\mathcal{L}_{\text{SBT}} = -m \log \sigma \left(\beta \log \frac{\pi_\theta(y^+|x)}{\pi_{\text{ref}}(y^+|x)} - \beta \log \frac{\pi_\theta(y^-|x)}{\pi_{\text{ref}}(y^-|x)} \right) \tag{8}$$
5. **DPO-PoP-iter**: Margins are inferred from preference-over-preference (PoP) supervision, using a PoP dataset constructed via iterative sampling.
6. **DPO-PoP-random**: Margins are inferred from PoP supervision, using a PoP dataset constructed via random sampling. This strategy can be interpreted as a bootstrapped version of the loss employed in DPO-margin-gt-scaled, along with a margin term (inside the log-sigmoid) that is inferred from preference-over-preference supervision.

We provide the results for LLama3.2-3b here. Results for LLama3.1-8b are provided in Appendix D

4.3 Discriminative Ability

First, we evaluate whether DPO-PoP improves the discriminative (i.e., classification) capabilities of the LLM. In addition to accuracy, we assess how well the model captures the strength of preferences by comparing the predicted and ground truth margins. For example, given a preference $A \succ B$, the ground truth margin is defined as the score difference between the preferred and dispreferred responses in the UltraFeedback dataset. The predicted margin is computed as the difference between the implicit rewards assigned to the two responses by the trained DPO model. A strong correlation between true and predicted margins indicates that the model generalizes well to unseen samples and is well-calibrated in its estimation of preference strength. We report both Spearman and Pearson correlation between predicted and ground truth margins. It should be noted that we would be unable to calculate the correlation metrics when PoP labels are collected from annotators. However, in this experimental setting, since we are using ground-truth preference scores from the UltraFeedback dataset, we are able to compute these metrics. This analysis is conducted solely to facilitate a better understanding of the algorithms.

We see from Table 1, that DPO-PoP-iter outperforms all other variants in test classification accuracy. Interestingly, even though DPO-margin-gt has access to the actual ground-truth margin values, it performs worse in terms of test classification accuracy. Looking at the correlation results, we observe



(a) Lower Cumulative Accuracy vs Margin (b) Upper Cumulative Accuracy vs Margin

Figure 2: Cumulative Accuracy vs Margin for the different DPO variants considered. Lower Cumulative Accuracy at margin m indicates the accuracy of predicting preference labels using only datapoints with ground-truth margin less than or equal to m. Conversely, Upper Cumulative Accuracy reflects prediction accuracy on datapoints with ground-truth margin greater than or equal to m. The dark grey histogram shows the distribution (density) of margin values in the test set. In plot (a), DPO-PoP-Iter achieves higher accuracy on datapoints with lower margins, while in plot (b), its performance drops for higher margin datapoints.

an interesting pattern: DPO-PoP-random achieves the highest Spearman and Pearson correlations, with DPO-PoP-iter closely matching it in terms of Spearman correlation. Notably, DPO-PoP-iter exhibits the lowest Pearson correlation, suggesting that while it captures the ordinal structure of the ground-truth margins well (reflected in high Spearman), its predicted margins vary non-linearly with the true margins, resulting in low Pearson correlation.

We also find that DPO-PoP-random has lower accuracy than DPO-PoP-iter, but higher correlations overall. On closer inspection, as shown in Figure 2, we find that DPO-PoP-iter tends to correctly classify more of the weaker preferences, which improves overall accuracy, though it sacrifices some accuracy on stronger preferences. In contrast, DPO-PoP-random more accurately captures stronger preferences at the expense of weaker ones, leading to lower classification accuracy. We hypothesize that by avoiding overfitting to potentially noisier weaker preferences, DPO-PoP-random better preserves the linear and ordinal relationships between the predicted and ground-truth margins, resulting in higher Pearson and Spearman correlations compared to the other methods.

We also report performance on RewardBench (Lambert et al., 2024) in Table 2. The DPO-PoP variants outperform all baselines, including those with access to ground-truth margins. Examining the Overall score, we observe that DPO-PoP-random achieves the highest performance. Notably, DPO-PoP-iter outperforms all methods on the Chat and Safety splits but underperforms on the Reasoning split—which comprises a larger portion of the dataset—resulting in a lower Overall score compared to DPO-PoP-random. In contrast, DPO-PoP-random delivers stable performance across all categories, securing the highest Overall score.

Algorithm	**Pearson Correlation**	**Spearman Correlation**	**Accuracy**
Vanilla DPO	0.3018	0.3082	0.71
DPO-margin-1	0.3019	0.3079	0.71
DPO-margin-gt	0.3489	0.3512	0.71
DPO-margin-gt-scaled	0.3463	0.3525	0.72
DPO-PoP-iter	0.2471	**0.3644**	**0.79**
DPO-PoP-random	**0.3598**	**0.3674**	0.71

Table 1: Comparison of DPO variants on classification accuracy and Spearman, Pearson correlation with ground-truth margins for LLaMA3.2-3b.

Model	Chat	Chat Hard	Safety	Reasoning	Overall
Vanilla-DPO	75.14	**64.69**	71.22	76.25	75.51
DPO-margin-1	77.65	64.25	72.57	76.83	76.18
DPO-margin-gt	80.17	63.60	75.54	77.38	77.12
DPO-margin-gt-scaled	80.45	63.60	75.54	76.07	76.85
DPO-PoP-iter	**87.99**	59.21	**80.14**	72.08	77.09
DPO-PoP-random	81.01	62.72	79.59	**77.75**	**78.66**

Table 2: Performance of LLaMA3.2-3b DPO variants on RewardBench. Higher is better.

4.4 Generative Ability

Next, we use UltraRM (Cui et al., 2024) to evaluate the responses of each of the aligned models and compare the quality of their generations. We use Vanilla-DPO as the reference model against which the other DPO variants are judged. We calculate the win rate and the median advantage of each model vs Vanilla DPO, as judged by UltraRM. The advantage of a datapoint is the difference between the UltraRM rewards of the response generated by the test model and the reference model, for a given prompt. The median advantage of a model is computed as the median of these per-prompt advantages over the entire test set. The results are displayed in the Table 3. We observe that DPO-PoP-random outperforms all other baselines in terms of win rate and median advantage. DPO-PoP-random which infers margins from PoP supervision, outperforms DPO variants that have access to ground truth margins.

Method	Median Advantage	Win Rate %
DPO-margin-1	0.1992	55%
DPO-margin-gt	0.6875	63%
DPO-margin-gt-scaled	0.1875	54%
DPO-PoP-iter	0.3281	57%
DPO-PoP-random	**0.7344**	**64%**

Table 3: Comparison of margin-based DPO variants against Vanilla DPO on median advantage and win rate for LLaMA3.2-3b.

We also report the performance of all the DPO variants on the AlpacaEval 2.0 benchmark (Dubois et al., 2025) in Table 4. DPO-PoP-random outperforms all other baselines both in terms of win-rate and length controlled win-rate.

Experiment	Length-Controlled Win Rate	Win Rate	Avg Length
Vanilla-DPO	12.68	12.05	1828
DPO-margin-1	12.07	11.80	1844
DPO-margin-gt	11.86	11.80	1876
DPO-margin-gt-scaled	11.42	11.43	1857
DPO-PoP-iter	11.03	11.18	1917
DPO-PoP-random	**14.68**	**14.41**	1858

Table 4: Performance of LLaMA3.2-3b DPO variants on the AlpacaEval 2.0 benchmark.

In both Tables 3 and 4, we observe that DPO-PoP-iter underperforms compared to DPO-PoP-random and DPO-margin-gt. We hypothesize that this is due to correctly classifying weaker preferences at the expense of stronger preferences, as discussed in Section 4.3. By potentially overfitting to noisy weaker preferences, DPO-PoP-iter suffers a drop in generative performance.

4.5 Discrimination vs Generation

We observe a trade-off between discriminative and generative performance. To improve generative performance, models should avoid overfitting to weaker preferences in the preference dataset. DPO-

PoP-iter offers good discriminative performance on test data that is in-distribution with respect to the training data, while it performs worse in terms of generative quality. DPO-PoP-random achieves good generative performance and is also robust in terms of discriminative performance, as supported by the RewardBench results in Table 2. These results enable informed choices: practitioners should use DPO-PoP-iter when the target is discriminative evaluation in a fixed domain and DPO-PoP-random when generative quality and robustness are priority. Furthermore, preference over preference annotations lead to significant generative performance gains when the size of the preference dataset is small, as seen in Appendix E

5 Related Work

Techniques that employ margins have largely been employed in the reward modeling phase of the RLHF pipeline. Touvron et al. (2023) used margins derived from preference ratings given by human annotators, in order to train reward models, and showed that the margin term can help the helpfulness reward model accuracy, especially when the two responses are more separable. Wang et al. (2025) propose Scaled Bradley-Terry loss, a margin based reward modeling objective that uses the margins derived from preference ratings in order to scale the loss for each datapoint. This can be seen as upsampling preferences for which the margin is higher. They show that the scaled loss variant leads to better performance that the margin loss variant proposed in Touvron et al. (2023). Wang et al. (2024b) propose Reward Difference Optimization, that also uses a scaled loss, but uses margins computed from a learned reward model to scale each data point. DPO-PoP-random can be interpreted as a bootstrapped variant of the Scaled Bradley-Terry loss(Wang et al., 2025; 2024b). Other approaches compute margins in different ways. Qin et al. (2024) define the margin as the average difference between the rewards of the chosen and rejected responses within each training batch. Wang et al. (2024a) use an ensemble of reward models and calculate the margin as the average reward difference across the ensemble for each preference.

In the case of Direct Alignment Algorithms (Rafailov et al., 2024a), IPO (Azar et al., 2023) and SLiC (Zhao et al., 2023) can also be interpreted in terms of margin, wherein IPO regresses the difference of implicit rewards to a fixed margin, whereas SLiC uses hinge loss with a fixed margin. Amini et al. (2024), propose ODPO, which is a variant of DPO with an offset. They use a reward model to label the preference data and also to provide the margin values to be used in the ODPO loss. Another approach, α-DPO (Wu et al., 2024a), redefines the reference policy $\hat{\pi}_{\text{ref}}$, to blend between the policy π and the reference policy π_{ref}, to achieve personalized reward margins. Wu et al. (2024b) observe that the optimal β value for the DPO loss depends on the informativeness of the pairwise preference data, and they propose β-DPO, which dynamically calibrates β at the batch level based on data quality. Our approach, DPO-PoP, on the other hand, gathers preference over preference information from an annotator to infer the margin values.

6 Conclusion

We introduced DPO-PoP, a framework that integrates adaptive margins into the DPO loss using preference-over-preference (PoP) supervision. Unlike prior approaches that derive margins from scalar preference ratings—whether provided by annotators or estimated via reward models—DPO-PoP infers margins directly from ordinal comparisons between preferences. We explored two PoP data sampling strategies: random and iterative. Our results show that improving discriminative performance by better modeling weaker preferences, as in DPO-PoP-iter, can come at the expense of generative quality. Furthermore, we show that DPO-PoP-random achieves stronger generative performance than DPO baselines using fixed or score-derived margins, while maintaining robust discriminative accuracy, as demonstrated on RewardBench.

These findings offer a practical takeaway for RLHF applications: DPO-PoP provides a way to perform margin-aware alignment using preference-over-preference annotation that is fine-grained in terms of resolution, compared to providing numerical scores. Practitioners can choose the sampling strategy based on their goals—favoring iterative sampling when discriminative performance is critical in-domain, and random sampling when prioritizing general-purpose generation and robustness

REFERENCES

Afra Amini, Tim Vieira, and Ryan Cotterell. Direct preference optimization with an offset. *arXiv preprint arXiv:2402.10571*, 2024.

Mohammad Gheshlaghi Azar, Mark Rowland, Bilal Piot, Daniel Guo, Daniele Calandriello, Michal Valko, and Rémi Munos. A general theoretical paradigm to understand learning from human preferences, 2023. URL https://arxiv.org/abs/2310.12036.

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

Nichola Burton, Michael Burton, Dan Rigby, Clare AM Sutherland, and Gillian Rhodes. Best-worst scaling improves measurement of first impressions. *Cognitive research: principles and implications*, 4(1):36, 2019.

Yaswanth Chittepu, Blossom Metevier, Will Schwarzer, Austin Hoag, Scott Niekum, and Philip S. Thomas. Reinforcement learning from human feedback with high-confidence safety constraints, 2025. URL https://arxiv.org/abs/2506.08266.

Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine learning*, 20:273–297, 1995.

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. Ultrafeedback: Boosting language models with scaled ai feedback, 2024. URL https://arxiv.org/abs/2310.01377.

Josef Dai, Xuehai Pan, Ruiyang Sun, Jiaming Ji, Xinbo Xu, Mickel Liu, Yizhou Wang, and Yaodong Yang. Safe rlhf: Safe reinforcement learning from human feedback, 2023. URL https://arxiv.org/abs/2310.12773.

Yann Dubois, Balázs Galambosi, Percy Liang, and Tatsunori B. Hashimoto. Length-controlled alpacaeval: A simple way to debias automatic evaluators, 2025. URL https://arxiv.org/abs/2404.04475.

Yoav Freund, Robert E Schapire, et al. Experiments with a new boosting algorithm. In *icml*, volume 96, pp. 148–156. Citeseer, 1996.

Aaron Grattafiori et al. The llama 3 herd of models, 2024. URL https://arxiv.org/abs/2407.21783.

Mai Lan Ha and Volker Blanz. Deep ranking with adaptive margin triplet loss. *arXiv preprint arXiv:2107.06187*, 2021.

John C Handley. Comparative analysis of bradley-terry and thurstone-mosteller paired comparison models for image quality assessment. In *PICS*, volume 1, pp. 108–112, 2001.

R. Herbrich and J. Weston. Adaptive margin support vector machines for classification. In *1999 Ninth International Conference on Artificial Neural Networks ICANN 99. (Conf. Publ. No. 470)*, volume 2, pp. 880–885 vol.2, 1999. doi: 10.1049/cp:19991223.

Yuge Huang, Yuhan Wang, Ying Tai, Xiaoming Liu, Pengcheng Shen, Shaoxin Li, Jilin Li, and Feiyue Huang. Curricularface: adaptive curriculum learning loss for deep face recognition. In *proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 5901–5910, 2020.

Natasha Jaques, Asma Ghandeharioun, Judy Hanwen Shen, Craig Ferguson, Àgata Lapedriza, Noah J. Jones, Shixiang Shane Gu, and Rosalind W. Picard. Way off-policy batch deep reinforcement learning of implicit human preferences in dialog. *ArXiv*, abs/1907.00456, 2019. URL https://api.semanticscholar.org/CorpusID:195766797.

Svetlana Kiritchenko and Saif M Mohammad. Best-worst scaling more reliable than rating scales: A case study on sentiment intensity annotation. *arXiv preprint arXiv:1712.01765*, 2017.

Nathan Lambert, Valentina Pyatkin, Jacob Morrison, LJ Miranda, Bill Yuchen Lin, Khyathi Chandu, Nouha Dziri, Sachin Kumar, Tom Zick, Yejin Choi, Noah A. Smith, and Hannaneh Hajishirzi. Rewardbench: Evaluating reward models for language modeling, 2024. URL https://arxiv.org/abs/2403.13787.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744, 2022.

Xue Bin Peng, Aviral Kumar, Grace Zhang, and Sergey Levine. Advantage-weighted regression: Simple and scalable off-policy reinforcement learning, 2019. URL https://arxiv.org/abs/1910.00177.

Jan Peters and Stefan Schaal. Reinforcement learning by reward-weighted regression for operational space control. In *Proceedings of the 24th international conference on Machine learning*, pp. 745–750, 2007.

Bowen Qin, Duanyu Feng, and Xi Yang. Towards understanding the influence of reward margin on preference model performance, 2024. URL https://arxiv.org/abs/2404.04932.

Rafael Rafailov, Yaswanth Chittepu, Ryan Park, Harshit Sikchi, Joey Hejna, Bradley Knox, Chelsea Finn, and Scott Niekum. Scaling laws for reward model overoptimization in direct alignment algorithms, 2024a. URL https://arxiv.org/abs/2406.02900.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model, 2024b. URL https://arxiv.org/abs/2305.18290.

Robert E Schapire, Yoav Freund, Peter Bartlett, and Wee Sun Lee. Boosting the margin: A new explanation for the effectiveness of voting methods. *The annals of statistics*, pp. 1651–1686, 1998.

Nisan Stiennon, Long Ouyang, Jeff Wu, Daniel M. Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul Christiano. Learning to summarize from human feedback, 2022. URL https://arxiv.org/abs/2009.01325.

Louis L Thurstone. A law of comparative judgment. In *Scaling*, pp. 81–92. Routledge, 2017.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.

Manya Wadhwa, Jifan Chen, Junyi Jessy Li, and Greg Durrett. Using natural language explanations to rescale human judgments, 2024. URL https://arxiv.org/abs/2305.14770.

Binghai Wang, Rui Zheng, Lu Chen, Yan Liu, Shihan Dou, Caishuang Huang, Wei Shen, Senjie Jin, Enyu Zhou, Chenyu Shi, Songyang Gao, Nuo Xu, Yuhao Zhou, Xiaoran Fan, Zhiheng Xi, Jun Zhao, Xiao Wang, Tao Ji, Hang Yan, Lixing Shen, Zhan Chen, Tao Gui, Qi Zhang, Xipeng Qiu, Xuanjing Huang, Zuxuan Wu, and Yu-Gang Jiang. Secrets of rlhf in large language models part ii: Reward modeling, 2024a. URL https://arxiv.org/abs/2401.06080.

Shiqi Wang, Zhengze Zhang, Rui Zhao, Fei Tan, and Cam Tu Nguyen. Reward difference optimization for sample reweighting in offline rlhf, 2024b. URL https://arxiv.org/abs/2408.09385.

Zhilin Wang, Yi Dong, Jiaqi Zeng, Virginia Adams, Makesh Narsimhan Sreedhar, Daniel Egert, Olivier Delalleau, Jane Polak Scowcroft, Neel Kant, Aidan Swope, and Oleksii Kuchaiev. Helpsteer: Multi-attribute helpfulness dataset for steerlm, 2023. URL https://arxiv.org/abs/2311.09528.

Zhilin Wang, Alexander Bukharin, Olivier Delalleau, Daniel Egert, Gerald Shen, Jiaqi Zeng, Oleksii Kuchaiev, and Yi Dong. Helpsteer2-preference: Complementing ratings with preferences, 2025. URL https://arxiv.org/abs/2410.01257.

Junkang Wu, Xue Wang, Zhengyi Yang, Jiancan Wu, Jinyang Gao, Bolin Ding, Xiang Wang, and Xiangnan He. α-dpo: Adaptive reward margin is what direct preference optimization needs, 2024a. URL https://arxiv.org/abs/2410.10148.

Junkang Wu, Yuexiang Xie, Zhengyi Yang, Jiancan Wu, Jinyang Gao, Bolin Ding, Xiang Wang, and Xiangnan He. β-dpo: Direct preference optimization with dynamic β, 2024b. URL https://arxiv.org/abs/2407.08639.

Xiao Zhang, Rui Zhao, Yu Qiao, Xiaogang Wang, and Hongsheng Li. Adacos: Adaptively scaling cosine logits for effectively learning deep face representations, 2019. URL https://arxiv.org/abs/1905.00292.

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

A Large Language Model Usage

Large Language Models (LLMs) were used solely for grammatical editing and improving writing flow. The research methodology, experimental design, data analysis, and all scientific conclusions are entirely the work of the human authors.

B Experiment Details

The hyperparameters used in our experiments for SFT and DPO are provided in Table 5 and Table 6 respectively. For DPO-PoP, we used the same hyperparameters used for DPO. For the DPO-PoP specific hyperparameters we set the clipping threshold $M_{\text{max}} = 10$ and the size of the PoP dataset to $120,000$ (twice the size of the preference dataset in UltraFeedback, i.e $k = 2$). All models were trained using 4 Nvidia A100 80G GPUs. The code is available at *removed for review*

Hyperparameter	Value
Epochs	1
Max Sequence Length	2048
Per-device Train Batch Size	2
Per-device Eval Batch Size	2
Gradient Accumulation Steps	8
Gradient Checkpointing	True
Num GPUs	4
Learning Rate	2e-5
Learning Rate Scheduler	Cosine
Weight Decay	0

Table 5: Training hyperparameters used for SFT

C Reinforcement Learning from Human Feedback

Reinforcement Learning from Human Feedback (RLHF) (Ouyang et al., 2022) is the predominant paradigm for aligning language models with human intent. The RLHF pipeline typically begins with a pre-trained language model trained on an internet-scale corpus and proceeds through three stages. We briefly describe each stage below:

Supervised Fine Tuning In the SFT stage, the model is fine-tuned to follow instructions by autoregressively predicting the next token in a sequence using Maximum Likelihood Estimation (MLE). This stage uses a dataset $\mathcal{D}_{\text{SFT}}$ consisting of prompt-response pairs (x, y), where x is a prompt and y is a high-quality response. These responses are either human-annotated or generated by large language models.

Hyperparameter	Value
Epochs	1
Max Sequence Length	2048
Per-device Train Batch Size	2
Per-device Eval Batch Size	2
Gradient Accumulation Steps	8
Gradient Checkpointing	True
Num GPUs	4
Learning Rate	1e-6
Learning Rate Scheduler	Cosine
Learning Rate Warmup Ratio	0.03
Weight Decay	0.05
Beta	0.1

Table 6: Training hyperparameters used for DPO

Reward Modeling In the reward modeling stage, a reward model is trained to assign scalar scores to prompt-response pairs, indicating how well a response aligns with human preferences. This process relies on a preference dataset $\mathcal{D}_{\text{pref}} = (x_i, y_i^+, y_i^-)_{i=1}^N$, where x_i is a prompt, y_i^+ is the preferred response, and y_i^- is the dispreferred response. Preference labels are typically provided by human annotators or large language models. The Bradley-Terry (BT) model (Bradley & Terry, 1952) is commonly used to model the likelihood of observed preferences.

$$P(y^+ \succ y^-) = \frac{e^{r(x,y^+)}}{e^{r(x,y^+)} + e^{r(x,y^-)}} = \sigma(r(x,y^+) - r(x,y^-)) \tag{9}$$

Here, r denotes the reward assigned to a prompt-response pair, and σ denotes the logistic (sigmoid) function. We parameterize the reward function as r_ϕ, where ϕ represents the model parameters, and use it to approximate the ground-truth reward function. The reward model is trained by maximizing the likelihood of the observed preference data under the Bradley-Terry model.

$$\min_\phi -\mathbb{E}_{(x,y^+,y^-)\sim\mathcal{D}_{\text{pref}}}[\log \sigma(r_\phi(x,y^+) - r_\phi(x,y^-))] \tag{10}$$

Reinforcement Learning In the reinforcement learning stage, the language model is optimized to generate responses that maximize the reward assigned by the learned reward model r_ϕ. However, directly optimizing for this reward can degrade response quality, as the policy may overfit to imperfections in the learned reward function and begin producing unnatural outputs (Jaques et al., 2019; Stiennon et al., 2022).

To mitigate this, a KL divergence constraint is added to ensure that the updated policy does not deviate too far from a reference policy, usually taken to be the supervised fine-tuning (SFT) policy. The resulting RL objective, with a KL penalty coefficient β, is given by:

$$\max_\theta \mathbb{E}_{x\sim\mathcal{D},y\sim\pi_\theta(.|x)}[r_\phi(x,y)] - \beta\mathbb{D}_{KL}[\pi_\theta(y|x)||\pi_{ref}(y|x)] \tag{11}$$

Additionally, some approaches (Chittepu et al., 2025; Dai et al., 2023) enforce safety and harmlessness by augmenting the objective in Equation 11 with an explicit cost constraint.

D Results for LLama3.1-8B

D.1 Discriminative Performance

The results showing the test classification accuracy on the UltraFeedback dataset (Cui et al., 2024) and RewardBench (Lambert et al., 2024) scores are in Tables 7 and 8 respectively.

Algorithm	Pearson Correlation	Spearman Correlation	Accuracy
Vanilla DPO	0.3151	0.3244	0.69
DPO-margin-1	0.3161	0.3243	0.69
DPO-margin-gt	0.3791	0.3715	0.70
DPO-margin-gt-scaled	0.3633	0.3669	0.71
DPO-PoP-iter	0.2183	**0.3868**	**0.82**
DPO-PoP-random	**0.3962**	**0.3871**	0.71

Table 7: Comparison of DPO variants on classification accuracy and Spearman, Pearson correlation with ground-truth margins for LLaMA3.1-8b.

Model	Chat	Chat Hard	Safety	Reasoning	Overall
Vanilla-DPO	73.46	63.60	57.03	76.69	71.59
DPO-margin-1	71.23	62.94	57.16	**77.07**	71.39
DPO-margin-gt	79.05	65.79	60.95	76.84	73.67
DPO-margin-gt-scaled	76.26	62.28	62.43	76.11	72.96
DPO-PoP-iter	**86.59**	61.84	**72.03**	72.05	75.41
DPO-PoP-random	81.56	**66.89**	68.51	76.95	**76.25**

Table 8: Performance of LLaMA3.1-8b DPO variants on RewardBench. Higher is better.

D.2 Generative Performance

The results displaying the win rate of the model responses as judged by UltraRM (Cui et al., 2024) and AlpacaEval 2.0 win rates (Dubois et al., 2025) are in Tables 9 and 10 respectively.

Method	Median Advantage	Win Rate %
DPO-margin-1	0.2813	55%
DPO-margin-gt	0.5000	59%
DPO-margin-gt-scaled	0.0938	52%
DPO-PoP-iter	0.3496	56%
DPO-PoP-random	**0.7500**	**63%**

Table 9: Comparison of margin-based DPO variants against Vanilla DPO on median advantage and win rate for LLaMA3.1-8b.

Experiment	Length-Controlled Win Rate	Win Rate	Avg Length
Vanilla-DPO	10.38	10.56	1869
DPO-margin-1	11.07	11.06	1864
DPO-margin-gt	11.23	11.30	1825
DPO-margin-gt-scaled	10.95	11.43	1881
DPO-PoP-iter	12.89	13.42	2004
DPO-PoP-random	**14.62**	**14.78**	1909

Table 10: Performance of LLaMA3.1-8b DPO variants on the AlpacaEval 2.0 benchmark.

E Effect of PoP data scale on performance

In order to study the effect of the PoP data scale on model performance, we consider the LLaMA3.2-3B model and begin with an initial subset of preferences of size $|\mathcal{D}_{\text{pref}}| = 7500$. We then generate a Preference-over-Preference (PoP) dataset of size $k \cdot |\mathcal{D}_{\text{pref}}|$, where $k \in \{1, 2, 4, 8, 16\}$. This procedure is carried out using both iterative and random sampling strategies for generating the PoP data. The

baseline DPO variants are all trained on the same subset of 7500 preferences used to construct the PoP dataset.

E.1 Discriminative Performance

Algorithm	Pearson Correlation	Spearman's Correlation	Accuracy
Vanilla-DPO	0.1450	0.1708	0.64
DPO-margin-1	0.1374	0.1609	0.64
DPO-margin-gt	0.1855	0.2091	0.65
DPO-margin-gt-scaled	0.1441	0.1656	0.64

Table 11: Comparison of baseline DPO variants trained on a subset of preferences ($|\mathcal{D}_{\text{pref}}| = 7500$), evaluated on classification accuracy and correlation with ground-truth margins for LLaMA3.2-3b.

Data Size Multiplier k	Pearson Correlation	Spearman's Correlation	Accuracy
1	**0.2229**	**0.2463**	**0.67**
2	0.2193	0.2429	**0.67**
4	0.2127	0.2325	0.65
8	0.2183	0.2268	0.64
16	0.2223	0.2236	0.63

Table 12: Performance of DPO-PoP-iter for varying values of k, evaluated on classification accuracy and correlation with ground-truth margins for LLaMA3.2-3b.

Data Size Multiplier k	Pearson Correlation	Spearman's Correlation	Accuracy
1	0.2386	0.2614	**0.67**
2	**0.2403**	**0.2638**	0.66
4	0.2362	0.2556	0.66
8	0.2322	0.2454	0.65
16	0.2265	0.2354	0.66

Table 13: Performance of DPO-PoP-random for varying values of k, evaluated on classification accuracy and correlation with ground-truth margins for LLaMA3.2-3b.

Comparing Table 11 with Tables 12 and 13, we observe that the DPO-PoP variants consistently outperform the DPO baselines in terms of discriminative performance, including those baselines that have access to ground-truth margins. Furthermore, increasing the data size multiplier k results in a decline in classification accuracy and correlation metrics with respect to the ground-truth margins for both DPO-PoP variants. Notably, this performance degradation is more pronounced in DPO-PoP-iter than in DPO-PoP-random. These findings suggest that, when prioritizing discriminative performance, using smaller values of k (e.g., $k = 1$ or $k = 2$) is advisable.

E.2 Generative Performance

Method	Median Advantage	Win Rate
DPO-margin-1	0.2500	0.56
DPO-margin-gt	0.4844	0.60
DPO-margin-gt-scaled	0.0313	0.51

Table 14: Median advantage and win rate of various DPO baseline variants over Vanilla-DPO, for LLaMA3.2-3b. All models are trained on a subset of preferences with $|\mathcal{D}_{\text{pref}}| = 7500$.

Data Size Multiplier k	Median Advantage	Win Rate
1	0.2813	0.55
2	1.1250	0.68
4	**1.7813**	**0.77**
8	1.7188	0.75
16	1.4629	0.69

Table 15: Median advantage and win rate of DPO-PoP-iter over Vanilla-DPO for different values of k, for LLaMA3.2-3b.

Data Size Multiplier k	Median Advantage	Win Rate
1	0.4688	0.57
2	1.2500	0.71
4	1.7969	**0.77**
8	**1.8711**	**0.77**
16	1.5547	0.72

Table 16: Median advantage and win rate of DPO-PoP-random over Vanilla-DPO for different values of k, for LLaMA3.2-3b.

Looking at Tables 15 and 16, we observe that the win rate initially increases with the data size multiplier k, before eventually declining. Additionally, DPO-PoP-random appears to be more robust to the choice of k than DPO-PoP-iter when considering win rate. When prioritizing generative ability, a moderately larger value of k (e.g., $k = 4$ or $k = 8$) is preferable. More importantly, when comparing with Table 14, we find that in a small-data regime, DPO-PoP variants achieve substantially higher win rates than the DPO baselines—including those with access to ground-truth margins.

F Effect of PoP labeling noise on performance

We investigate the sensitivity of our DPO-PoP approaches to noise in PoP labels collected from annotators. Given our PoP dataset $|\mathcal{D}_{\text{PoP}}|$, we introduce label noise by randomly flipping PoP labels with probability ϵ. We use the LLama3.2-3b model and experiment with three different noise levels: $\epsilon \in \{0.1, 0.3, 0.5\}$. We evaluate both the discriminative and generative performance of models trained on these perturbed datasets.

F.1 Discriminative Performance

We observe from Figure 3 that both the Spearman and Pearson correlations for DPO-PoP-iter and DPO-PoP-random decrease as the noise level increases. Notably, this decline in correlation is more pronounced for DPO-PoP-iter compared to DPO-PoP-random. From the accuracy plot, we surprisingly find that the test classification accuracy of DPO-PoP-iter slightly increases with added noise, while it marginally decreases for DPO-PoP-random. We hypothesize that label noise induces a regularizing effect in DPO-PoP-iter, which helps mitigate its tendency to overfit to weaker preferences.

F.2 Generative Performance

We observe from Figure 4 that both the win rate and median advantage for DPO-PoP-random decrease as the noise level increases. Similar to the trend observed in the discriminative setting, we find that the win rate and median advantage for DPO-PoP-iter initially increase before declining, suggesting that a moderate amount of label noise may have a regularizing effect, helping DPO-PoP-iter avoid overfitting to weaker preferences.

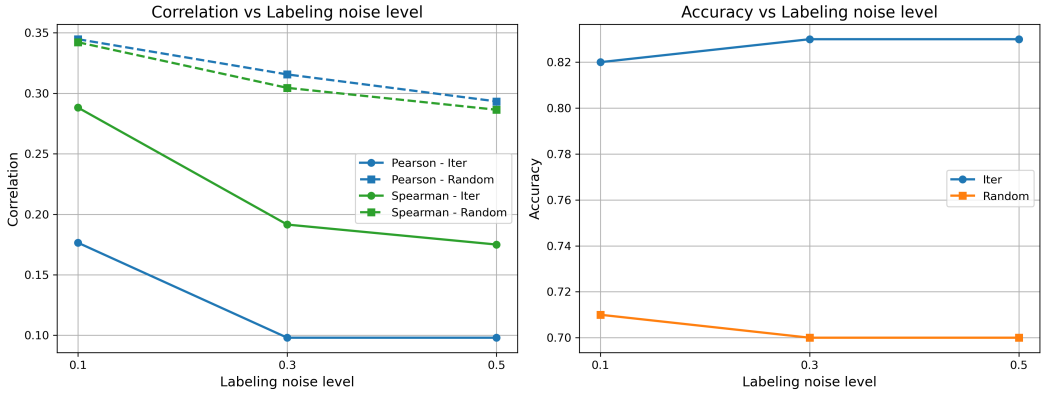


Figure 3: Spearman and Pearson correlations (left), and test classification accuracy (right) of DPO-PoP models trained with varying levels of label noise.

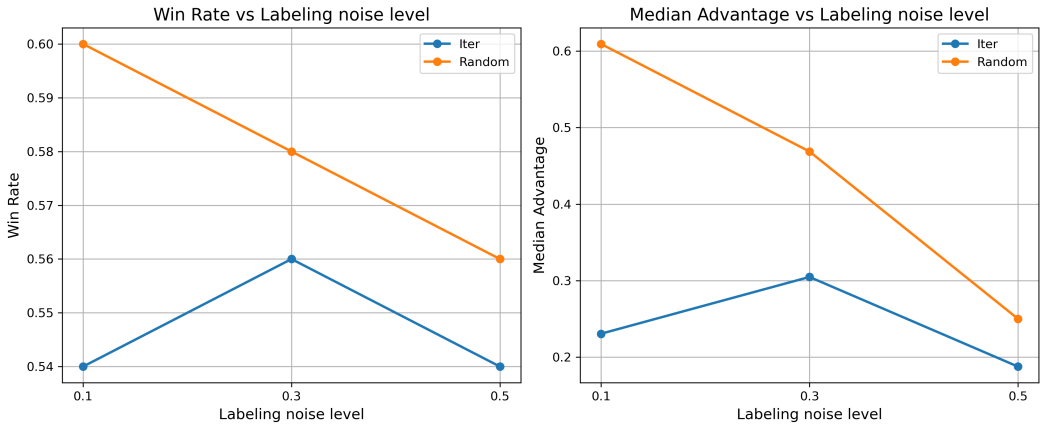


Figure 4: Win rates (left) and median advantage (right) of DPO-PoP models trained with varying levels of label noise.