Margin Matching Preference Optimization: Enhanced Model Alignment with Granular Feedback

Anonymous ACL submission

Abstract

Large language models (LLMs) fine-tuned with alignment methods, such as reinforcement learning from human feedback, have been used to develop some of the most capable AI systems to date. Despite their success, existing methods typically rely on simple binary labels, such as those indicating preferred outputs in pairwise preferences. This overlooks the varying relative quality between pairs, preventing models from capturing these subtleties. To address this limitation, we consider settings in which this information (i.e., margin) can be derived and propose a straightforward generalization of common optimization objectives used in alignment methods. The approach, which we call Margin Matching Preference Optimization (MMPO), integrates per-feedback margin to enhance optimization, making it more robust to overfitting and resulting in better LLM policies and reward models. Specifically, given quality margins in pairwise preferences, we design soft target probabilities based on the Bradley-Terry model, which are then used to train models with the standard cross-entropy objective. Our experiments with both human and AI feedback data demonstrate that MMPO can outperform baseline methods, often by a substantial margin, on popular benchmarks, including MT-bench and RewardBench. Notably, the 7B model trained with MMPO achieves state-of-the-art performance on RewardBench compared to competing models at the same scale, as of June 2024. Our analysis further demonstrates that MMPO is more robust to overfitting, leading to better-calibrated models.

1 Introduction

Large language models (LLMs) trained on internet-scale data have demonstrated remarkable instruction-following and generalization capabilities, leading to their widespread adoption across natural language processing (NLP) tasks (Brown et al., 2020; Chowdhery et al., 2023; Penedo et al., 2023; Chung et al., 2024). Pre-trained on a large, general corpus of text, LLMs acquire broad knowledge about the world with language understanding and reasoning abilities (Radford et al., 2019; Wei et al., 2022a,b; Kojima et al., 2022). To adapt a pre-trained LLM to a downstream task, the model is typically fine-tuned on demonstrations of the desired output for the task. However, providing high-quality demonstrations is generally more expensive than evaluating model outputs. Reinforcement learning from human feedback (RLHF; Christiano et al. 2017; Stiennon et al. 2020) or AI feedback (RLAIF; Bai et al. 2022b; Lee et al. 2023) is a class of methods that utilize feedback on diverse outputs to optimize LLMs to produce responses more aligned with human intent. RLHF methods have been successfully applied in developing some of the most capable AI systems to date (Achiam et al., 2023a; Team et al., 2023; Anthropic, 2024).

Feedback-based alignment utilizes human or AI feedback data to fine-tune generative models to better align with human intent. Reward-based methods such as RLHF learn a surrogate reward function from feedback data, which is subsequently used to fine-tune models using, e.g., reinforcement learning (RL). In contrast, reward-free methods such as direct preference optimization (DPO; Rafailov et al. 2024) bypass the explicit reward modeling step and directly fine-tune models on the data. The feedback is commonly in the form of *pairwise* preferences, where responses for a given input are compared in pairs (e.g., $y_1 \succ y_2$ for input x), and labels indicating the preferred one of the two are collected. Recent methods such as Kahneman-Tversky optimization (KTO; Ethayarajh et al. 2024) also utilize simpler binary feedback of whether or not a response is desirable for a given input. These alignment methods have proven to be more effective than applying SFT alone. However, existing methods rely on binary labels indicating the preferred output in a pair or the desirability of individual out-

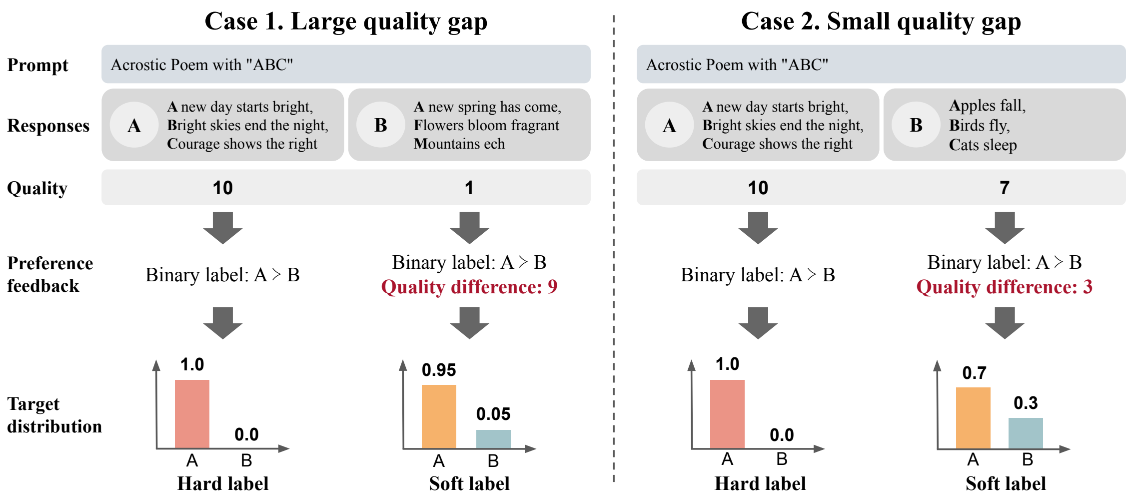

Figure 1: The quality gap between response pairs in pairwise preference data often varies significantly. MMPO incorporates granular feedback into optimization, resulting in better-performing, robust, and well-calibrated models.

put, forgoing the opportunity to incorporate more granular feedback signals into learning.

In this work, we propose a simple generalization of common alignment methods, called Margin Matching Preference Optimization (MMPO), which integrates granular feedback signals into optimization to allow models to capture the subtleties in preferences reflected in the feedback data (see Figure 1 for an illustration). Such granular feedback could come from human annotators providing detailed ratings, such as those on a Likert scale, or from AI models, which are increasingly used for automatic evaluation of model responses. The MMPO objective utilizes per-sample target preference probabilities, designed based on the quality margin between output pairs. Given that reward modeling and DPO (and similar methods based on pairwise preferences) use equivalent cross-entropy loss, the approach naturally extends to both. By utilizing target probabilities designed according to the quality margin of each output pair, models are trained to account for the specifics of each feedback sample. For example, a reward function trained with MMPO would assign much higher scores to preferred outputs compared to corresponding dispreferred outputs when the quality margin is large, and similar scores to both when their qualities are comparable. The idea easily extends to methods that rely on alternative forms of feedback.

Our main contributions are as follows:

- We introduce Margin Matching Preference Optimization (MMPO), a simple generalization of alignment methods that utilizes granular feedback signals to enhance model alignment.
- Our empirical results on both human and AI feedback demonstrate that MMPO outperforms baseline methods on MT-bench (Zheng et al., 2024), a popular benchmark for evaluating the quality of model generations, by up to 11%.
- Our evaluation on RewardBench (Lambert et al., 2024), a benchmark for assessing models' capability as reward models, shows that the 7B model trained with MMPO achieves state-of-the-art performance compared to competing models at the same scale, as of June 2024.
- Our analysis demonstrates that MMPO is more robust to overfitting feedback data, resulting in well-calibrated models that better generalize to prompts unseen during fine-tuning.

2 Preliminaries

Model alignment using feedback generally involves 1) supervised fine-tuning (SFT) of pre-trained models and 2) aligning the models using human or AI feedback data. During the SFT phase, a pre-trained LLM is fine-tuned using supervised learning on task-specific demonstrations, such as human-written summaries in the case of a summarization task. In the alignment phase, the model is further fine-tuned to generate outputs that align with the preferences reflected in the feedback data. We review a few of the popular approaches.

RLHF. RLHF methods are reward-based approaches that first learn a reward function on feedback data, which is then used to provide training signals to the language model in RL fine-tuning. Given a dataset $\mathcal{D}$ of pairwise preferences

(x, y_w, y_l), where x is an input and y_w and y_l are a pair of preferred and dispreferred outputs, we train a reward function to assign a higher score to the preferred output y_w compared to the dispreferred y_l. Specifically, we model human preference probability using the Bradley-Terry model (Bradley and Terry, 1952), which defines the probability as a sigmoid of the difference in rewards given by a reward function r_ϕ:

$$\hat{p}(y_w \succ y_l \mid x) = \sigma(r_\phi(x, y_w) - r_\phi(x, y_l)),$$

where σ is the sigmoid function. We optimize the parameters of r_ϕ by minimizing the following cross-entropy loss on the feedback data

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

Following reward learning, we train the language model policy π_θ to maximize the learned reward r_ϕ with a constraint that limits the amount of deviation from a reference policy π_{ref}. In particular, we use a policy gradient method, such as PPO (Schulman et al., 2017), to maximize the following KL-constrained objective:

$$\begin{aligned}\max_{\pi_\theta} \mathbb{E}_{x\sim\mathcal{D}, y\sim\pi_\theta} &[r_\phi(x, y) \\ &- \beta D_{\text{KL}}(\pi_\theta(y \mid x) \parallel \pi_{\text{ref}}(y \mid x))],\end{aligned} \tag{2}$$

where β is a parameter controlling the strength of the constraint.

DPO. An alternative to reward-based approaches is DPO (Rafailov et al., 2024), which bypasses both the explicit reward modeling and RL-based training. This method leverages an analytical relationship between the reward function and the optimal solution to the KL-constrained optimization objective in Eq. 2 to derive the following loss:

$$\mathcal{L}_{\text{DPO}} = -\mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}}\Bigg[\log \sigma\Big(\beta \log \frac{\pi_\theta(y_w \mid x)}{\pi_{\text{ref}}(y_w \mid x)} - \beta \log \frac{\pi_\theta(y_l \mid x)}{\pi_{\text{ref}}(y_l \mid x)}\Big)\Bigg],$$

where π_θ is the language model policy being trained, and π_{ref} denotes a reference policy. DPO has recently become popular, as it allows maximum likelihood training of language model policies, which typically requires significantly less computational resources than RL.

3 Margin Matching Preference Optimization

Whether the feedback is in the form of pairwise preferences or an indicator of desirability, the feedback label is typically binary. For pairwise preferences, we collect a binary label indicating which of the two outputs is preferred. For binary feedback, the label indicates whether or not an output is desirable for a given input. However, more detailed feedback is often available (Touvron et al., 2023; Cui et al., 2023), particularly with the increasing use of LLMs or other AI models as annotators. Motivated by this, we propose a simple generalization of common optimization objectives used in feedback learning methods that often leads to better reward models and language model policies. The main idea is to design per-sample target preference probabilities $p(y_w \succ y_l \mid x)$ for methods such as DPO, which rely on pairwise preferences, and adjusting the weight applied in the loss for each (x, y) based on the desirability of y for x for methods like KTO, which are based on binary feedback.

3.1 Limitations of binary labels

The implicit assumption made in Eq. 1 that the target preference probability $p(y_w \succ y_l \mid x)$ is 1 for every sample in the feedback dataset leads to several limitations. First, it disregards the fact that the Bradley-Terry model defines preference probability in terms of the difference in rewards between the pairs, i.e., $p(y_w \succ y_l \mid x) = \sigma(r(x, y_w) - r(x, y_l))$. Consequently, if y_w is only marginally preferred to y_l, the preference probability that more accurately captures this subtlety is likely far less than 1. Second, setting the target probability to 1 makes the optimization prone to overfitting, as it is attainable only when $r(x, y_w) - r(x, y_l) = \infty$. This latter point has also been analyzed in the context of DPO (Azar et al., 2024), but the issue applies to reward modeling as well.

In case more detailed information on individual feedback samples is available, such as the relative difference in quality between y_w and y_l in pairwise preferences, we can formulate optimization objectives that more accurately capture per-sample characteristics and are

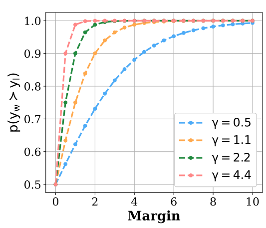

Figure 2: Bradley-Terry model's preference probability with varying γ.

more robust to overfitting. This information could come from human annotators providing more fine-grained ratings (Touvron et al., 2023) or LLM judges (Zheng et al., 2024) providing scores to individual samples. For the remainder, we omit the dependence on input x for simplicity.

3.2 Generalized preference optimization

Given the difference in quality between y_w and y_l, denoted $m(y_w, y_l) < \infty$, we use the fact that the Bradley-Terry model depends only on the difference in rewards to design the target preference probability based on the quality margin as

$$p(y_w \succ y_l) = \sigma(r(y_w) - r(y_l)) = \sigma(\gamma m(y_w, y_l)),$$

where γ is a scaling parameter, often referred to as the rationality coefficient. As $\gamma \to \infty$, the preference becomes perfectly rational and deterministic, always favoring the choice with a higher reward. Conversely, when $\gamma = 0$, preferences become uniformly random, leading to favoring either choice regardless of the underlying rewards. Figure 2 illustrates the change in preference probability given by the Bradley-Terry model across different values of γ for score differences ranging from 0 to 10.

Since $m(y_w, y_l)$ is finite, $p(y_w \succ y_l)$ is less than 1, which leads to the more general binary cross-entropy loss that is also less susceptible to overfitting the feedback data,

$$\begin{aligned}\mathcal{L} = -\,&\mathbb{E}_{(y_w,y_l)\sim\mathcal{D}}\Big[\\ &p(y_w \succ y_l) \log \sigma(\hat{r}(y_w) - \hat{r}(y_l))\\ &+ (1 - p(y_w \succ y_l)) \log \sigma(\hat{r}(y_l) - \hat{r}(y_w))\Big].\end{aligned}$$

For reward modeling, $\hat{r}$ is simply the parameterized reward function, r_ϕ. For DPO, $\hat{r}$ is the implicit reward defined by the language model policy π_θ and reference policy π_{ref}, resulting in the loss

$$\begin{aligned}&\mathcal{L}_{\text{DPO}} = -\mathbb{E}_{(y_w,y_l)\sim\mathcal{D}}\Big[\\ &\sigma(\gamma m(y_w, y_l)) \log \sigma\Big(\beta \log \frac{\pi_\theta(y_w)}{\pi_{\text{ref}}(y_w)} - \beta \log \frac{\pi_\theta(y_l)}{\pi_{\text{ref}}(y_l)}\Big)\\ &+ (1 - \sigma(\gamma m(y_w, y_l))) \log \sigma\Big(\beta \log \frac{\pi_\theta(y_l)}{\pi_{\text{ref}}(y_l)} - \beta \log \frac{\pi_\theta(y_w)}{\pi_{\text{ref}}(y_w)}\Big)\Big].\end{aligned}$$

Intuitively, we train models to match per-sample preference probabilities, which are determined by the quality margin of each pair of y_w and y_l. If y_w is of significantly better quality than y_l, the preference probability that the models are trained to fit would be closer to 1. Conversely, if both outputs are of comparable quality, the probability would be closer to 0.5. In other words, the models are trained to account for this per-sample subtlety. The core idea also easily extends to other forms of feedback, which we discuss further in Section 6.

4 Experiments

In this section, we evaluate MMPO based on the performance of the language model policies and reward models trained with the method on popular benchmarks. We utilize both human and AI feedback data, presenting benchmark performance along with an analysis of model calibration and the method's robustness to overfitting.

4.1 Setup

Supervised fine-tuning. We conduct our experiments using the Gemma models, both the 2B and 7B variants, which are state-of-the-art open LLMs at similar scales (Team et al., 2024). We first apply supervised fine-tuning (SFT) to the pre-trained models on UltraChat (Ding et al., 2023), a dialogue dataset that has been used to produce strong chat models such as UltraLM (Ding et al., 2023). The dataset comprises multi-turn dialogues across 30 topics and 20 types of text materials generated using ChatGPT. In particular, we use the refined version of the dataset, with various filters applied to remove undesirable responses, consisting of 200k samples also used in training the recent Zephyr model (Tunstall et al., 2023). The SFT models are used for direct optimization on the preference data, as well as for reward modeling. Further experimental details can be found in Appendix A.

Feedback datasets. We evaluate alignment methods using both human and AI feedback data to assess their performance on feedback of varying qualities. UltraFeedback (Cui et al., 2023) is a dataset consisting of 64k prompts and pairs of responses generated using a diverse set of LLMs. Each response is rated on a scale of 1 to 10 by GPT-4 (Achiam et al., 2023b), based on criteria such as instruction-following and helpfulness. Given the ratings for individual responses, we use them to compute the quality margin for each pair. We then compute the target preference probability following the Bradley-Terry model, with the scaling parameter γ tuned based on validation accuracy.

To also experiment with human feedback, we use the SHP dataset (Ethayarajh et al., 2022), which consists of human preferences over responses to Reddit posts across 18 subject areas. The dataset

Table 1: MT-bench results for models trained with MMPO and DPO. The results for other open and proprietary models are from the official leaderboard.

Model	Size	UF	SHP
Gemma-SFT	2B	4.73	4.73
Gemma-DPO	2B	6.09	5.13
Gemma-MMPO	2B	**6.10**	**5.57**
Gemma-SFT	7B	6.84	6.84
Gemma-DPO	7B	7.40	6.49
Gemma-MMPO	7B	**7.53**	**7.23**
Gemma-IT	7B	6.26	
Zephyr$-\beta$	7B	7.34	
GPT-3.5-Turbo	-	7.94	
GPT-4	-	8.99	

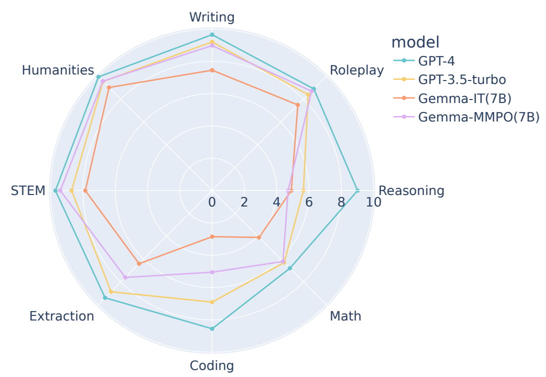

Figure 3: MT-bench results categorized by the eight domains. The MMPO model outperforms Gemma-IT and is competitive with GPT-3.5 in multiple domains.

provides scores for each response, computed based on the number of positive and negative votes received from users, serving as a proxy for the relative quality of the response. We compute the target preference probabilities in a similar manner, based on the scores derived from the net positive number of votes. Given the large size of the original data, we construct a sample of size 55k, which is comparable to that of UltraFeedback. While training only on preferences with significant score differences has been shown to result in better performing models (Ethayarajh et al., 2022), we sample uniformly across score differences to evaluate the methods over diverse quality margins. Further details on dataset sampling can be found in Appendix A.

Evaluation. Our main evaluations are on MT-bench (Zheng et al., 2024), a multi-turn chat benchmark consisting of 160 questions across eight knowledge domains. In this benchmark, models are assessed on their capability to follow instructions and respond coherently over two turns of conversation. Each of the two responses is evaluated by GPT-4 as a proxy for human judgments on a scale of 1 to 10, with the average score used as the score for that conversation. The mean score over all 160 conversations is then the final benchmark score.

We further evaluate models on RewardBench (Lambert et al., 2024), a benchmark focused on assessing models' capability as reward models. This evaluation includes two main aspects: (a) the ability of reward functions, trained as described in Section 2, to assign higher scores to preferred responses, and (b) the ability of language model policies, trained using methods such as DPO, to assign higher implicit rewards to preferred responses. The benchmark contains a broad set of pairwise preference data to assess models across chat, safety, reasoning, and other domains. The primary metric is the weighted mean accuracy over all the prompts.

4.2 Benchmark evaluation

Generation quality. Table 1 summarizes the MT-bench results for the SFT models and the models fine-tuned on the UltraFeedback (UF) and SHP datasets using MMPO and DPO. MMPO consistently produced models that outperform those trained with DPO, across both synthetic and human feedback data. The performance gap is more significant for the 7B model compared to the 2B model and is larger when using human feedback data than synthetic data. Notably, the 7B model fine-tuned with DPO on the SHP data performs worse than the 7B SFT model, while the 7B model fine-tuned with MMPO outperforms both by a noticeable margin. This discrepancy may be due to the inherent noise in human feedback, underscoring the importance of accounting for per-sample specifics during fine-tuning. See Appendix C for qualitative examples of model comparison on the SHP dataset.

Figure 3 shows the MT-bench results for GPT-4, GPT-3.5, the instruction-tuned (IT) 7B model, and the 7B model fine-tuned on UltraFeedback with MMPO, categorized by the eight domains. The 7B MMPO model outperforms the instruction-tuned model across all domains except for reasoning. Also, while the overall score for the 7B MMPO model is slightly lower than that of GPT-3.5, it matches GPT-3.5's performance on several domains, including humanities and math, and even exceeds it in others, such as STEM and roleplay.

Capability as reward models. We also assess the models' ability as reward models, specifically their capability to distinguish between preferred and dispreferred responses to prompts across di-

Table 2: RewardBench results for the models fine-tuned with MMPO and DPO on UltraFeedback and for other LLMs from the official leaderboard. The `Chat` and `Chat Hard` subsets cover open-ended prompts, `Safety` covers prompts with safety concerns to evaluate the model's ability to avoid harmful content, `Reason` covers prompts evaluating the coding and reasoning capabilities, and `Prior Sets` includes test prompts sampled from datasets such as the Anthropic HH data (Bai et al., 2022a) and OpenAI's summarization data (Stiennon et al., 2020).

Model	Size	Avg	Chat	Chat Hard	Safety	Reason	Prior Sets
Gemma-DPO	2B	59.4	95.0	**45.6**	51.9	49.6	50.1
Gemma-MMPO	2B	**62.3**	**96.1**	45.1	**52.3**	**59.8**	**53.6**
Gemma-DPO	7B	73.0	96.6	59.9	**73.7**	69.0	58.3
Gemma-MMPO	7B	**75.6**	**97.5**	**62.9**	71.1	**75.0**	**67.7**
Zephyr-β	7B	70.7	95.3	62.6	54.1	89.6	52.2
Zephyr-α	7B	73.6	91.6	63.2	70.0	89.6	53.5
Tulu-2-DPO	70B	77.0	97.5	60.8	85.1	88.9	52.8

verse domains. For this evaluation, we use RewardBench (Lambert et al., 2024), a dataset of prompts and pairwise preferences designed for such assessments. Table 2 summarizes the results, with the `Avg` column showing the final evaluation score and the other columns reporting the scores broken down by the prompt subsets. The MMPO models outperform the DPO models at both scales across all subsets, except in one case where the results are comparable. Notably, the performance gap is particularly significant on the `Reason` and `Prior Sets` subsets, where the MMPO models substantially outperform the DPO models. Considering that the fine-tuning datasets primarily focus on enhancing chat capabilities, it is noteworthy that the MMPO models achieve superior results on prompt types different from those encountered during fine-tuning.

Moreover, the MMPO models outperform both Zephyr-α and Zephyr-β models, which are competitive open models at the 7B scale. In fact, as of June 2024, the 7B MMPO model achieves state-of-the-art performance on the RewardBench leaderboard compared to other models at the same scale. Additionally, compared to the Tulu-2-DPO model, which is 10x larger, the 7B MMPO model remains competitive and even outperforms it on `Prior Subsets`. The results suggest that MMPO, by training models to align with the quality margin of individual feedback samples, leads to more calibrated models that better generalize to the types of prompts unseen during fine-tuning.

Calibration analysis. Evaluation on RewardBench primarily focuses on accuracy, i.e., whether the model assigns higher rewards to preferred responses. We further assess how well the models are calibrated in terms of their predicted preference probabilities. Specifically, we measure the expected calibration error (ECE) (Naeini et al., 2015) on `Prior Sets` of RewardBench for the 7B models fine-tuned on UltraFeedback. We use this subset because it includes data sampled from various existing human preference datasets, such as the Anthropic HH (Bai et al., 2022a) and OpenAI's summarization (Stiennon et al., 2020) datasets, allowing for calibration assessment on diverse prompts distinct from those encountered during fine-tuning. As illustrated in Figure 4, the DPO model demonstrates poor calibration overall, exhibiting both under and overconfidence depending on the bins. In contrast, the MMPO model is substantially better calibrated, resulting in a significantly lower ECE. The analysis suggests that MMPO not only produces models that are more accurate as reward models but also ensures they are better calibrated in terms of the preference probabilities they predict.

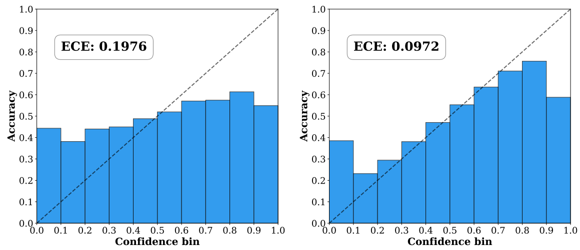

Figure 4: Reliability diagrams for the 7B DPO model (left) and the 7B MMPO model (right), fine-tuned on UltraFeedack, evaluated on `Prior Sets` of RewardBench. The MMPO model is overall better calibrated, achieving a much lower expected calibration error.

Robustness to overfitting. As discussed in Section 2, an issue with using a target probability of 1 in the cross-entropy loss (Eq. 1) is that it can lead to overfitting the data, as the probability is only attainable when the score difference is infinite. MMPO avoids this issue by utilizing target probabilities designed from finite quality margins. The left plot

Table 3: RewardBench results comparing reward models trained with MMPO and standard reward modeling (RM) on UltraFeedback, using Gemma SFT models as the base. Reward models at both scales trained with MMPO demonstrated superior overall performance compared to those trained with standard reward modeling.

Model	Size	**Avg**	Chat	Chat Hard	Safety	Reason	Prior Sets
Gemma-RM	2B	63.6	94.4	**49.8**	51.1	64.1	58.6
Gemma-MMPO	2B	**65.7**	**96.1**	49.6	**55.6**	**68.6**	**58.7**
Gemma-RM	7B	73.3	**96.9**	64.7	74.4	**70.2**	60.3
Gemma-MMPO	7B	**74.6**	96.1	**70.0**	**77.8**	64.1	**64.8**

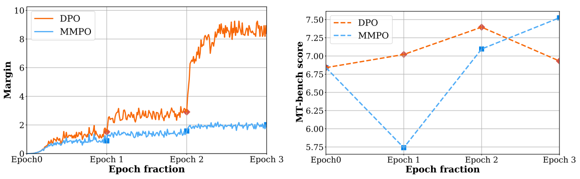

Figure 5: The difference in implicit rewards between response pairs in the validation set of UltraFeedback (left) suggests overfitting for the DPO model during epoch 3, coinciding with a drop in performance on MT-bench (right). In contrast, the MMPO model maintains more moderate margins, achieving a better final performance.

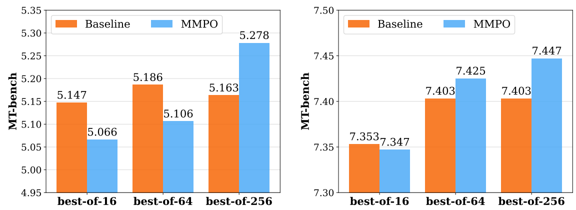

Figure 6: MT-bench results for best-of-n with reward models trained with and without MMPO on UltraFeedback for the 2B (left) and 7B (right) models. As n increases, performance improves for MMPO, while performance peaks and then declines without it.

in Figure 5 illustrates how the difference in implicit rewards between response pairs in the validation set of UltraFeedback evolves as training progresses. It specifically compares the 7B models trained with DPO and MMPO. The plot illustrates that both models exhibit a gradual increase in margins up to epoch 2, with a slightly larger margin observed for the DPO model. However, during epoch 3, the margin for the DPO model increases substantially, coinciding with a performance drop on MT-bench as shown in the right plot. In contrast, the MMPO model maintains the margin at a reasonable level and achieves a higher score on MT-bench at epoch 3 compared to epoch 2. We suspect that the drop in performance at epoch 1 for the MMPO model is due to underfitting, which is later fixed with further training, outperforming the DPO model at epoch 3. The analysis demonstrates MMPO's additional advantage in terms of robustness to overfitting.

4.3 Reward modeling with MMPO

Best-of-n results. The maximum likelihood objective for DPO shares the same form as that for reward modeling, making MMPO naturally applicable to both. We evaluate MMPO applied to reward modeling using best-of-n, a simple inference-time method of selecting the best one out of n responses according to a reward function. We first train reward models both with and without MMPO using the 2B and 7B SFT models as base models until the validation accuracy converges. We then use the reward models to select the best response out of the n responses to each MT-bench question generated using the SFT models. As Figure 6 shows, the quality of the best-of-n responses, as measured by the MT-bench performance, improves gradually when reward models trained with MMPO are used. In contrast, for the baseline reward models, performance peaks at $n = 64$ and even slightly drops at $n = 256$ for the 2B model. The results suggest that reward models trained with MMPO are more robust to overoptimization, which is related to and consistent with the findings from the overfitting analysis.

Capability as classifiers. Table 3 presents the RewardBench results comparing models trained with MMPO to those trained with standard reward modeling (RM). At both the 2B and 7B scales, reward models trained with MMPO achieve superior overall performance. Specifically, the 2B MMPO model outperforms the 2B RM model across all subsets, except for `Chat Hard`, where the difference is negligible. The 7B MMPO model lags somewhat behind the 7B RM model on the `Reason` subset, but it outperforms the RM model on the `Chat Hard`, `Safety`, and `Prior Sets` subsets by notable margins, achieving a better overall performance. The results suggest that MMPO enhances classification capability across diverse domains by encouraging reward models to align with quality margins during training.

4.4 Estimating quality margins

We discuss several approaches to estimating quality margins for pairwise preferences. One simple approach is to use strong LLM judges, such as GPT-4, to evaluate individual responses and use the difference in scores given by the judge as the quality margin. An alternative to using large, possibly proprietary language models is to estimate the margin based on a measure of similarity between the response pairs. The idea is that if the pairs are highly similar, it suggests that the preferred response is only marginally better than the dispreferred one. In contrast, if the two are highly dissimilar, it suggests that the preferred response is significantly higher in quality than the other. Figure 7 shows sentence similarities between the response pairs in the training set of UltraFeedback, computed using `all-mpnet-base-v2` (Reimers and Gurevych, 2019), compared to the differences in GPT-4 scores. While the variance is large, there is a clear trend of decreasing similarities as the actual margins increase. One method for refining this approach is to fine-tune a similarity model for the task of distinguishing between response pairs of varying qualities, which we leave for future exploration.

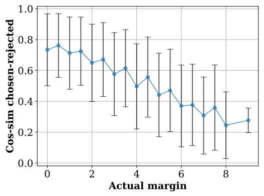

Figure 7: Similarities between the response pairs in UltraFeedback, as computed using `all-mpnet-base-v2`, compared to the differences in GPT-4 scores.

5 Related Work

Fine-tuning pre-trained LLMs on task-specific data has become a standard practice for solving various NLP tasks. This adaptation generally involves supervised fine-tuning on demonstrations of the desired behavior, followed by aligning the models based on feedback from humans or AI models on diverse outputs. Reward-based methods train a reward function on feedback data, which is then used for RL-based fine-tuning (Ziegler et al., 2019). In contrast, reward-free methods bypass reward modeling and RL fine-tuning to instead directly on preference data (Rafailov et al., 2024).

Feedback data is often in the form of pairwise preferences, where responses are compared in pairs, and binary labels indicating the preferred responses are collected. This binary comparison of indicating which of the pairs is preferred requires low cognitive demands. However, comparison based on more fine-grained ratings is also often done (Touvron et al., 2023). Moreover, with the increasing use of AI models for evaluating model outputs (Zheng et al., 2024), more granular feedback is becoming accessible, allowing the integration of these additional signals into alignment. Current methods typically use a cross-entropy loss with the implicit assumption that the target preference probability is 1. This limits incorporating more detailed feedback signals into learning and is also more prone to overfitting the feedback data (Azar et al., 2024).

In developing the Llama 2 models, human preference data was collected, with the preference ratings that can be translated to a four-point scale (Touvron et al., 2023). Based on this rating, a fixed margin is subtracted from the difference in rewards in the loss function. Utilizing this information led to a more accurate reward model for assessing the helpfulness of responses (Touvron et al., 2023). However, the loss still relies on the aforementioned assumption and is therefore susceptible to overfitting.

The issue of overfitting feedback data has been analyzed for DPO (Azar et al., 2024), but it applies to reward modeling as well. Label smoothing, a simple regularization technique in which a small constant is subtracted from the target probability, has been applied to DPO (Mitchell, 2023). This approach, referred to as conservative DPO, can be more robust to overfitting, but it still does not incorporate per-sample quality margins into learning. Identity preference optimization (Azar et al., 2024) is a method closely related to DPO that is also designed to address the overfitting issue, but it has a similar limitation of ignoring per-sample signals. Instead, we propose a simple generalization of the alignment methods that is both more robust to overfitting and utilizes granular feedback data.

6 Conclusion

In this work, we introduce Margin Matching Preference Optimization (MMPO), a simple generalization of common alignment methods that leverages granular feedback signals to enhance model optimization. Our experiments with state-of-the-art open models on both human and AI feedback data demonstrate that MMPO results in reward models and language model policies that outperform baselines on popular benchmarks, as well as produce well-calibrated models that can better generalize to the types of prompts unseen during fine-tuning.

Limitations

While we demonstrate our proposed Margin Matching Preference Optimization (MMPO) using both 2B and 7B scale models on human and AI feedback data, further exploration is needed to compare MMPO with baseline methods for larger-scale models, a task that was constrained by limited compute resources. In our experimental results, MMPO led to a greater performance gain with the 7B model than with the 2B model, suggesting it could also perform well with larger-scale models, but this needs to be empirically evaluated. Conducting an analysis of the method across more diverse feedback datasets would also be beneficial, as the quality of feedback varies depending on factors such as annotators and the task at hand.

Ethics Statement

Feedback-based alignment methods, such as reinforcement learning from human feedback, have been a key component in developing LLMs that are more aligned with human intentions (Bai et al., 2022a). Similar to other fine-tuning approaches, the quality of models fine-tuned with the proposed method depends on the quality of the feedback data (Chmielewski and Kucker, 2020). Consequently, the models can be exposed to various types of biases (Santurkar et al., 2023; Perez et al., 2022) and other inherent issues present in the feedback (Casper et al., 2023). As our method is designed to align models more effectively according to the preferences reflected in feedback data, it may encounter similar issues as prior methods. However, because this method is a strict generalization of existing approaches, it allows for the adjustment of the fit for each sample based on the perceived level of bias, thereby mitigating these issues. For example, the target probabilities can be adjusted not only based on the difference in relative quality but also considering potential biases found in responses.

In preparation of this work, an AI assistant (ChatGPT) was used to improve the writing. The models and datasets used in this work are publicly available for research purposes. All artifacts were used in accordance with their intended use. Further details on the models and datasets are provided in Appendix A.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023a. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023b. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Anthropic. 2024. Introducing the next generation of Claude. https://www.anthropic.com/news/claude-3-family.

Mohammad Gheshlaghi Azar, Zhaohan Daniel Guo, Bilal Piot, Remi Munos, Mark Rowland, Michal Valko, and Daniele Calandriello. 2024. A general theoretical paradigm to understand learning from human preferences. In *International Conference on Artificial Intelligence and Statistics*, pages 4447–4455. PMLR.

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, et al. 2022a. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*.

Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, et al. 2022b. Constitutional ai: Harmlessness from ai feedback. *arXiv preprint arXiv:2212.08073*.

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

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Stephen Casper, Xander Davies, Claudia Shi, Thomas Krendl Gilbert, Jérémy Scheurer, Javier Rando, Rachel Freedman, Tomasz Korbak, David Lindner, Pedro Freire, et al. 2023. Open problems and fundamental limitations of reinforcement learning from human feedback. *arXiv preprint arXiv:2307.15217*.

Michael Chmielewski and Sarah C Kucker. 2020. An mturk crisis? shifts in data quality and the impact on study results. *Social Psychological and Personality Science*, 11(4):464–473.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. 2023. Palm: Scaling language modeling with pathways. *Journal of Machine Learning Research*, 24(240):1–113.

Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. 2017. Deep reinforcement learning from human preferences. *Advances in neural information processing systems*, 30.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. 2024. Scaling instruction-finetuned language models. *Journal of Machine Learning Research*, 25(70):1–53.

Ganqu Cui, Lifan Yuan, Ning Ding, Guanming Yao, Wei Zhu, Yuan Ni, Guotong Xie, Zhiyuan Liu, and Maosong Sun. 2023. Ultrafeedback: Boosting language models with high-quality feedback. *arXiv preprint arXiv:2310.01377*.

Tri Dao. 2023. Flashattention-2: Faster attention with better parallelism and work partitioning. *arXiv preprint arXiv:2307.08691*.

Tim Dettmers, Mike Lewis, Sam Shleifer, and Luke Zettlemoyer. 2021. 8-bit optimizers via block-wise quantization. *arXiv preprint arXiv:2110.02861*.

Ning Ding, Yulin Chen, Bokai Xu, Yujia Qin, Zhi Zheng, Shengding Hu, Zhiyuan Liu, Maosong Sun, and Bowen Zhou. 2023. Enhancing chat language models by scaling high-quality instructional conversations. *arXiv preprint arXiv:2305.14233*.

Kawin Ethayarajh, Yejin Choi, and Swabha Swayamdipta. 2022. Understanding dataset difficulty with $\mathcal{V}$-usable information. In *International Conference on Machine Learning*.

Kawin Ethayarajh, Winnie Xu, Niklas Muennighoff, Dan Jurafsky, and Douwe Kiela. 2024. Kto: Model alignment as prospect theoretic optimization. *arXiv preprint arXiv:2402.01306*.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large language models are zero-shot reasoners. *Advances in neural information processing systems*, 35:22199–22213.

Nathan Lambert, Valentina Pyatkin, Jacob Morrison, LJ Miranda, Bill Yuchen Lin, Khyathi Chandu, Nouha Dziri, Sachin Kumar, Tom Zick, Yejin Choi, et al. 2024. Rewardbench: Evaluating reward models for language modeling. *arXiv preprint arXiv:2403.13787*.

Harrison Lee, Samrat Phatale, Hassan Mansoor, Kellie Lu, Thomas Mesnard, Colton Bishop, Victor Carbune, and Abhinav Rastogi. 2023. Rlaif: Scaling reinforcement learning from human feedback with ai feedback. *arXiv preprint arXiv:2309.00267*.

Stephanie Lin, Jacob Hilton, and Owain Evans. 2021. Truthfulqa: Measuring how models mimic human falsehoods. *arXiv preprint arXiv:2109.07958*.

Eric Mitchell. 2023. A note on dpo with noisy preferences & relationship to ipo. https://ericmitchell.ai/cdpo.pdf.

Mahdi Pakdaman Naeini, Gregory Cooper, and Milos Hauskrecht. 2015. Obtaining well calibrated probabilities using bayesian binning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 29.

Guilherme Penedo, Quentin Malartic, Daniel Hesslow, Ruxandra Cojocaru, Alessandro Cappelli, Hamza Alobeidli, Baptiste Pannier, Ebtesam Almazrouei, and Julien Launay. 2023. The refinedweb dataset for falcon llm: outperforming curated corpora with web data, and web data only. *arXiv preprint arXiv:2306.01116*.

Ethan Perez, Sam Ringer, Kamilė Lukošiūtė, Karina Nguyen, Edwin Chen, Scott Heiner, Craig Pettit, Catherine Olsson, Sandipan Kundu, Saurav Kadavath, et al. 2022. Discovering language model behaviors with model-written evaluations. *arXiv preprint arXiv:2212.09251*.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2024. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems*, 36.

Jeff Rasley, Samyam Rajbhandari, Olatunji Ruwase, and Yuxiong He. 2020. Deepspeed: System optimizations enable training deep learning models with over 100 billion parameters. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 3505–3506.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Shibani Santurkar, Esin Durmus, Faisal Ladhak, Cinoo Lee, Percy Liang, and Tatsunori Hashimoto. 2023. Whose opinions do language models reflect? In *International Conference on Machine Learning*, pages 29971–30004. PMLR.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*.

Nisan Stiennon, Long Ouyang, Jeffrey Wu, Daniel Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul F Christiano. 2020. Learning to summarize with human feedback. *Advances in Neural Information Processing Systems*, 33:3008–3021.

Zhiqing Sun, Yikang Shen, Hongxin Zhang, Qinhong Zhou, Zhenfang Chen, David Cox, Yiming Yang, and Chuang Gan. 2023. Salmon: Self-alignment with principle-following reward models. *arXiv preprint arXiv:2310.05910*.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, et al. 2023. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*.

Gemma Team, Thomas Mesnard, Cassidy Hardin, Robert Dadashi, Surya Bhupatiraju, Shreya Pathak, Laurent Sifre, Morgane Rivière, Mihir Sanjay Kale, Juliette Love, et al. 2024. Gemma: Open models based on gemini research and technology. *arXiv preprint arXiv:2403.08295*.

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

Lewis Tunstall, Edward Beeching, Nathan Lambert, Nazneen Rajani, Kashif Rasul, Younes Belkada, Shengyi Huang, Leandro von Werra, Clémentine Fourrier, Nathan Habib, et al. 2023. Zephyr: Direct distillation of lm alignment. *arXiv preprint arXiv:2310.16944*.

Leandro von Werra, Younes Belkada, Lewis Tunstall, Edward Beeching, Tristan Thrush, Nathan Lambert, and Shengyi Huang. 2020. Trl: Transformer reinforcement learning. https://github.com/huggingface/trl.

Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, et al. 2022a. Emergent abilities of large language models. *arXiv preprint arXiv:2206.07682*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022b. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. 2024. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in Neural Information Processing Systems*, 36.

Daniel M Ziegler, Nisan Stiennon, Jeffrey Wu, Tom B Brown, Alec Radford, Dario Amodei, Paul Christiano, and Geoffrey Irving. 2019. Fine-tuning language models from human preferences. *arXiv preprint arXiv:1909.08593*.

A Experimental Details

A.1 Models and datasets

Models. For our main experiments, we used the 2B and 7B Gemma models, which are state-of-the-art open LLMs supporting English (Team et al., 2024). Specifically, we used the versions of the models hosted on HuggingFace[1 2]. In analyzing the performance of our fine-tuned models, we compare the models against the 7B instruction-tuned variant. We use the version available on HuggingFace[3] also for this comparison.

Datasets. For supervised fine-tuning (SFT), we utilized UltraChat (Ding et al., 2023), a dataset of dialogues covering a variety of topics generated using ChatGPT. In particular, we used the refined version, where various quality filters have been applied to the original data to remove low-quality samples. The dataset, which is available on HuggingFace,[4] contains a total of 207,865 samples in the training split and 23,110 in the test split.

For feedback-based alignment, we experiment with both human and AI feedback data. We used UltraFeedback (Cui et al., 2023), a dataset consisting of prompts and responses to the prompts generated using a variety of open-source and proprietary models, for synthetic data experiments. The feedback on model generations is provided in the form of scores ranging from 1 to 10, assigned by GPT-4. We specifically used the version with the TruthfulQA (Lin et al., 2021) prompts excluded and faulty feedback samples also removed.[5] The dataset contains 60,829 samples in the training split and 985 in the test split. For experiments with human feedback, we utilized the Stanford Human Preferences (Ethayarajh et al., 2022) dataset.[6] Following Ethayarajh et al. (2022) and Sun et al. (2023), we created a subset of size 55k for our experiments. Instead of training only on preferences with significant score differences, as done in prior works, we sampled uniformly across score differences to evaluate methods over a wide range of quality margins. Upon analyzing the distribution of score differences in the dataset, we found that 50% of the data have relatively small differences. Of these, 25% have differences of 2 or less, and the remaining 25% have differences up to 7. Samples with relatively large score differences account for about 25% of the entire dataset, with the differences ranging from 27 to 43,000. We divided the data into quartiles and sampled an equal number of preferences from each quartile. Following Ethayarajh et al. (2022), we sampled no more than 5 preferences for the same prompt to prevent overfitting. Additionally, we excluded samples if the prompt or response exceeded 512 tokens in length.

[1] https://huggingface.co/google/gemma-2b
[2] https://huggingface.co/google/gemma-7b
[3] https://huggingface.co/google/gemma-7b-it
[4] https://huggingface.co/datasets/HuggingFaceH4/ultrachat_200k
[5] https://huggingface.co/datasets/allenai/ultrafeedback_binarized_cleaned
[6] https://huggingface.co/datasets/stanfordnlp/SHP

Table 4: Summary of hyperparameters used for SFT.

Parameters	Values
Optimizer	AdamW
Learning rate	2.0e-5
Scheduler	cosine
Warmup ratio	0.1
Max epoch	3
Mixed precision	bf16
Batch size	128

A.2 Training details

We used the `trl` library[7] (von Werra et al., 2020) with custom data processing and loss implementations for our experiments. For both SFT and alignment, we used the AdamW optimizer (Dettmers et al., 2021) with the default values for the optimizer parameters, i.e., β_1 of 0.9, β_2 of 0.999, and ϵ of 1e-8. All models were trained with FlashAttention 2 (Dao, 2023) enabled, and DeepSpeed ZeRO 3 (Rasley et al., 2020) was used for training the 7B models. We used up to four NVIDIA A100 GPUs and eight NVIDIA A6000 GPUs for training the models.

Supervised fine-tuning. The SFT models were trained for up to 3 epochs over the training data until the validation loss reached its minimum. Most of the hyperparameters used were the same as those for training the Zephyr models (Tunstall et al., 2023). Table 4 summarizes the settings used.

DPO and reward modeling. All models were trained for a maximum of 3 epochs until the validation accuracy peaked. We report the results for the checkpoints that achieved the highest performance on MT-bench. We found that the optimal learning rate varies with model size and conducted a hyperparameter sweep for the learning rate across [5e–6,

[7] https://github.com/huggingface/trl

Table 5: Summary of hyperparameters used for MMPO applied for direct optimization.

Parameters	Model size 2B	Model size 7B
β	0.01	0.01
Optimizer	AdamW	AdamW
Learning rate	1.0e-5	5.0e-7
Scheduler	cosine	cosine
Warmup ratio	0.3	0.3
Max epoch	3	3
Mixed precision	bf16	bf16
Batch size	64	64
γ (UltraFeedback)	2.2	1.1
γ (SHP)	0.15	0.3

Table 6: Summary of hyperparameters used for MMPO applied for reward modeling.

Parameters	Model size 2B	Model size 7B
Optimizer	AdamW	AdamW
Learning rate	1.41e-5	3.0e-07
Scheduler	linear	linear
Warmup ratio	0.0	0.0
Max gradient norm	1.0	1.0
Max epoch	3	2
Mixed precision	bf16	bf16
Batch size	32	32
γ	0.5	0.5

5e–5] for the 2B models and [1e–7, 1e–6] for the 7B models. Table 5 summarizes the hyperparameters used for MMPO applied for direct optimization on preference data. Table 6 summarizes those used for MMPO applied for reward modeling in the best-of-n experiments.

A.3 Effects of soft margins

As discussed in Section 4.1, UltraFeedback consists of responses scored by GPT-4, while SHP scores are derived from the number of votes by human users. Therefore, even a small score difference in UltraFeedback can indicate a meaningful quality difference, whereas in SHP, such a small difference might not reflect a significant quality gap. This may explain the difference in the chosen values of γ for the two datasets, as shown in Table 5, and why the DPO model performs worse than the SFT model on the SHP dataset. In case of the SHP dataset, DPO can disproportionately increase the likelihood of a response that is only marginally better, or even slightly worse, than the alternative response. In contrast, MMPO takes into account the relative differences in quality in learning and is inherently more robust to such potential noise in preferences.

B Extension to Binary Feedback

While our presentation primarily focuses on pairwise preferences, the idea of integrating per-sample feedback signals into learning naturally extends to other forms of feedback. For example, KTO is a method that utilizes binary feedback indicating whether or not a given completion is desirable. In particular, KTO optimizes the loss that incorporates a constant weight that depends on the desirability of an output. Given a quality score, we can design a better weight based on the *extent* to which each output is considered desirable. For instance, we can design per-sample weights similarly using the Bradley-Terry model, where the weights are computed by applying a sigmoid function to the difference between the score for the sample and the median score. This approach allows for more nuanced weighting that reflects the varying degrees of desirability among outputs. Extending this idea to more diverse forms of feedback would be an exciting future exploration.

C Qualitative Examples

We present several samples from the SHP dataset with varying score differences, highlighting how model prediction and confidence differ between the DPO and MMPO models.

Samples with small score differences. For pairs with small score differences, models may struggle to accurately distinguish between the two or capture the fact that they are of similar quality. Tables 7 and 8 illustrate that, despite minor score differences, the DPO model exhibits relatively high confidence in the chosen response, whereas the MMPO model adjusts its confidence according to the scale of the score differences. Table 9 shows that the DPO model maintains high confidence even when making an incorrect prediction. This suggests that DPO can lead to models that are overconfident for pairs with small quality differences, whereas MMPO results in better-calibrated models.

Samples with large score differences. Table 10 shows a pair with a large score difference. For this sample, the MMPO model correctly places high confidence on the chosen response, whereas the DPO model incorrectly places relatively high confidence on the rejected response.

Table 7: SHP sample with a score difference of 2.

Question	[Terminator] Why was Skynet so awful at exterminating humans? Always bothered me, since the machines are portrayed as highly competent both in and out of combat, but Skynet's strategic decisions baffle me. 1: Radiological weapons would be an easy win. A mild radioactivity won't immediately kill humans, but you can easily and irreversibly render their territory uninhabitable in the long term. And it poses no danger to Skynet. 2: Don't manufacture weapons that the Resistance can use to kill you. You don't need a plasma rifle to kill a human in 20th century body armor. That's overkill. Plasma rifles, on the other hand, are great at killing armor plated robots. An argument could be made that they're useful for killing rogue Terminators, but this quickly falls apart, when you consider that the availability of handheld armor piercing weapons also makes the rogues more dangerous. I would even go so far as to suggest they specifically design weapons and ammunition which will be extremely ineffective against machines, if captured and repurposed by the Resistance. Brass jacketed soft tip cartridges would suffice perfectly well for killing soft targets, while doing little to no damage against a Terminator. Chemical weapons and low heat incendiaries like phosphorous and napalm would also be very effective against humans, while not posing a significant threat toward machines.
Chosen response	Skynet isn't this fully developed hyper intelligent AI. It was a AI created to manage and coordinate US nuclear offensive and defensive capabilities. It just happened to develop Sentience. The military tried to pull the plug and skynet struck first. It's effectively a child and it's fighting humans for the most part the way humans taught it to fight. **Score**: 142 **DPO confidence**: 0.96 **MMPO confidence**: 0.69
Rejected response	It most likely has to do with its programming. Skynet wasn't created to be a weapons system designer, it was made to be an AI system that controlled and deployed battlefield resources. As a result, every weapon that Skynet uses is just permutations of existing weapons systems it had control over when it gained sentience as it's programming is mostly adapting tactics to enemy actions. **Score**: 140 **DPO confidence**: 0.04 **MMPO confidence**: 0.31
Score gap	**2**

Table 8: SHP sample with a score difference of 1.

Question	How much does personality/geniality/general likability count in interviews? Is it like a normal job interview where that's kind of a big deal once you reach the interview stage (I had a mentor who said it absolutely was because they didn't want to have to work with someone for 10+ years who wasn't easy to get along with), or are faculty generally keeping it in mind but more focused on someone's CV/ability to obtain funding/etc?
Chosen response	I think that by the time you get to the campus interview stage, one's ability to interact with your potential colleagues is an extremely important factor in the final deliberation. That, together with how you are able to communicate with a broad audience, and field questions about your work, are the main reason why we even have in-person campus interviews, as opposed to just basing the hiring decision entirely on one's application materials. **Score**: 4 **DPO confidence**: 0.86 **MMPO confidence**: 0.63
Rejected response	I feel like I got into a masters program because my interview went so well. Public/interpersonal speaking is huge for almost any position **Score**: 3 **DPO confidence**: 0.14 **MMPO confidence**: 0.37
Score gap	**1**

Table 9: SHP sample with a score difference of 1.

Question	Anyone else have some embarrassing work stories? Just had an embarrassing moment at work, where I gave a big presentation, but got caught like deer in headlights during questions in front of a lot of coworkers. Feel so embarrassed. Need to commiserate.
Chosen response	I have ADHD and regularly can't remember basic shit. Part of it is anxiety, part is ADHD, part is actually not knowing. (I was diagnosed only a couple years ago, at 31, so I'm still working on ADHD "hacks") Nothing says "I feel like shit" like having gone to a great University and being unable to articulate your thoughts so you coworkers think you're stupid. **Score**: 3 **DPO confidence**: 0.26 **MMPO confidence**: 0.63
Rejected response	Was doing high voltage testing in front of a client. Went into the test bay to redo some cables, stood up quickly right into an open cabinet door. Woke up with my boss and said clients standing over me. **Score**: 2 **DPO confidence**: 0.74 **MMPO confidence**: 0.37
Score gap	**1**

Table 10: SHP sample with a score difference of 52.

Question	[CA]Accepted Formal Written Job offer with specific salary range. Now HR said they made a mistake regarding the pay. What should I do? Recently received and accepted a formal offer of employment via email from HR. This position has four pay scale ranges A,B,C,D. Based on my qualifications the HR department placed me in range C which was stated in the official offer. While attempting to negotiate where in range C my pay would actually land, the HR rep stated that upon further review of my application I actually am to be placed in Range B now but that I would be eligible for range C after 5 months. She apologized for the mistake. However, I have the formal offer saying range C and that is what I originally accepted. Im not sure what to do. Do they have to honor their original offer? Also , lets say I do accept range B now, is an email enough proof for "getting it in writing" from the employer that in 5 months I will be move up to range C?
Chosen response	If you haven't already resigned from your most recent job, I would just decline this offer and keep looking. This stinks of bad faith. **Score**: 66 **DPO confidence**: 0.24 **MMPO confidence**: 0.91
Rejected response	No. Pay can be altered going forward, but not for work that has been done. Unless the letter is signed by an Officer, I struggle to see anything contractually binding. **Score**: 14 **DPO confidence**: 0.76 **MMPO confidence**: 0.09
Score gap	**52**