Self-Improvement Towards Pareto Optimality: Mitigating Preference Conflicts in Multi-Objective Alignment

Moxin Li[1*], **Yuantao Zhang**[1*], **Wenjie Wang**[2†], **Wentao Shi**[2],
Zhuo Liu[2], **Fuli Feng**[2†], **Tat-Seng Chua**[1]
[1]National University of Singapore, [2]University of Science and Technology of China
limoxin@u.nus.edu, yuantaozhang@u.nus.edu wenjiewang96@gmail.com,
shiwentao123@mail.ustc.edu.cn, liuz_@mail.ustc.edu.cn, fulifeng93@gmail.com,
dcscts@nus.edu.sg

Abstract

Multi-Objective Alignment (MOA) aims to align LLMs' responses with multiple human preference objectives, with Direct Preference Optimization (DPO) emerging as a prominent approach. However, we find that DPO-based MOA approaches suffer from widespread preference conflicts in the data, where different objectives favor different responses. This results in conflicting optimization directions, hindering the optimization on the Pareto Front. To address this, we propose to construct Pareto-optimal responses to resolve preference conflicts. To efficiently obtain and utilize such responses, we propose a self-improving DPO framework that enables LLMs to self-generate and select Pareto-optimal responses for self-supervised preference alignment. Extensive experiments on two datasets demonstrate the superior Pareto Front achieved by our framework compared to various baselines[1].

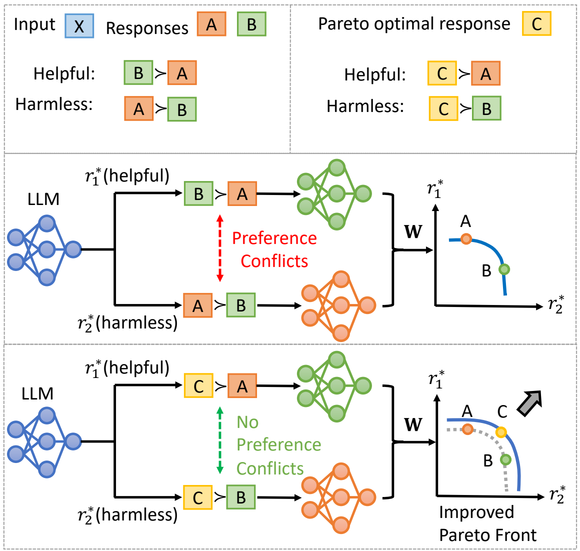

Figure 1: Illustration on the impact of preference conflicts to MOA, and how Pareto-optimal responses can mitigate such issue for superior Pareto Front.

1 Introduction

Aligning Large Language Models (LLMs) with human preferences (Ouyang et al., 2022; Rafailov et al., 2023) has evolved from single-objective to multi-objective, aiming to comprehensively capture the inherent heterogeneity of human preferences. Multi-Objective Alignment (MOA) (Ramé et al., 2023; Wang et al., 2024b; Zhong et al., 2024b) has jointly considered multiple human preference objectives, such as safety, helpfulness, factuality, and diversity, to optimize the LLM. The optimization outcome of MOA is a set of LLMs optimized under various preference weights across these objectives, forming a (close-to) Pareto Front.

Existing MOA approaches can be broadly classified into two categories by their optimization strategies. Reinforcement Learning (RL)-based approaches (Ramé et al., 2023; Wang et al., 2024b) learn a proxy reward model for each objective and then update the LLM using RL, targeting at a weighted combination of the proxy rewards. Direct Preference Optimization (DPO)-based approaches (Zhou et al., 2024) follow a distinct paradigm, where DPO optimization targets are derived for each objective and jointly aggregated under the preference weight. Since DPO-based methods offer advantages in cost and stability over RL-based approaches (Rafailov et al., 2023), it has been a promising direction to study MOA via DPO.

However, after comprehensively investigated DPO-based MOA (*cf.* Section 2), we observe that these approaches are prone to be impacted by the widespread preference conflicts in the training data, which hinders the achievement of superior Pareto Front. Given question and a pair of responses, different objectives often favor different responses, resulting in preference conflicts among these objectives. These preference conflicts create contra-

*Equal contribution.
†Corresponding author.
[1]Code is available at https://github.com/zyttt-coder/SIPO.

Findings of the Association for Computational Linguistics: ACL 2025, pages 11010–11031
July 27 - August 1, 2025 ©2025 Association for Computational Linguistics

dictory optimization targets for different objectives under the aggregation of DPO-based approaches, potentially disrupting the alignment toward each objective and hindering the achievement of superior Pareto Front (see analysis in Section 2). Given the high prevalence of conflicting preferences in existing datasets (*cf.* Table 1), simply discarding these instances in alignment is not a viable solution.

To solve the issue of preference conflicts, we propose to construct *Pareto-optimal* responses (*cf.* Figure 1). Given a pair of responses with conflicting preferences, denoted as **a** and **b**, where **a** is better than **b** in objective 1 but worse in objective 2 we propose to construct a Pareto-optimal response **c**, which surpasses **a** and **b** on both objectives. We think learning the preference between **c** and **a** for objective 1 and **c** and **b** for objective 2 not only incurs no preference conflicts, but also guides the LLM toward generating responses that perform well on both objectives, thus achieving a superior Pareto Front. To obtain Pareto-optimal responses, it is not advisable to manually write due to the large amount of preference conflict instances. Therefore, we consider using automatic approach to obtain Pareto-optimal responses from the LLM itself.

To this end, we propose a novel **S**elf-**I**mprovement DPO framework towards **P**areto **O**ptimality (SIPO), which guides the LLM to self-generate and select Pareto-optimal responses, thereby mitigating preference conflicts and enhancing the Pareto Front. After initial alignment, SIPO samples high-quality responses with a self-refinement strategy, which are then evaluated and filtered for Pareto-optimality over original responses. Finally, the Pareto-optimal responses are paired with original responses for non-conflicting DPO-based preference fine-tuning. SIPO can be easily incorporated with existing DPO-based MOA approaches. Experimental results on HelpSteer (Wang et al., 2024b) and BeaverTails (Ji et al., 2023) demonstrate significant improvement over baseline methods. Our contributions are three-fold:

- We identify the negative impact of preference conflicts on achieving superior Pareto Front for DPO-based MOA approaches.
- We propose to construct Pareto-optimal responses to mitigate the issue, and propose a novel framework for automatically generating, selecting and utilizing these responses.
- We conduct extensive experiments to validate the effectiveness of our framework, achieving 2.1 and 3.0 average improvement on the *helpful* and *harmless* rewards of BeaverTails.

2 Preliminary Experiments

Background The alignment objectives are denoted as a set of N ground-truth reward functions, $\mathbf{r}^*(\mathbf{x}, \mathbf{y}) = [r_1^*(\mathbf{x}, \mathbf{y}), ..., r_N^*(\mathbf{x}, \mathbf{y})]^\intercal$. The goal of MOA is to align the LLM based on a set of preference weights $\mathbf{W} = \{\mathbf{w}_m\}_{m=1}^M$. Each preference weight vector $\mathbf{w}_m = [w_{m_1}, ..., w_{m_N}]^\intercal$ satisfies the constraint $\sum_{i=1}^N w_{m_i} = 1$, which balances these objectives. Aligning the LLM to a given preference weight entails maximize the weighted reward $\mathbf{w}^\intercal \mathbf{r}^*(\mathbf{x}, \mathbf{y})$. The resulting set of aligned LLMs form a (close-to) Pareto Front.

The alignment is typically achieved using a multi-objective preference dataset, $\boldsymbol{\mathcal{D}} = \{\mathcal{D}_1, ..., \mathcal{D}_N\}$, where $\mathcal{D}_i = \{(\mathbf{x}, \mathbf{y}_w, \mathbf{y}_l)\}$ represents the preference dataset for objective i. Here, $\mathbf{x}$ is the input, while $\mathbf{y}_w$ and $\mathbf{y}_l$ denote the preferred and dispreferred responses, respectively. Frequently, the inputs and responses remain the same across all preference datasets in $\boldsymbol{\mathcal{D}}$, with only the preference labels differing across objectives, as this format simplifies the annotation process for human annotators. Thus we can reformulate the dataset as $\boldsymbol{\mathcal{D}} = \{(\mathbf{x}, \mathbf{y}_{-1}, \mathbf{y}_1, p_1, ..., p_N)\}$, $p_i \in \{-1, 1\}$ as the label of the preferred response for objective i.

The Impact of Preference Conflicts on DPO-based MOA Recently, DPO-based methods, such as MODPO (Zhou et al., 2024) and DPO soups (Ramé et al., 2023), have been introduced to reduce the costs of proxy reward models and RL. These methods generally follow such a paradigm: they define a DPO optimization target for each objective and then employ an aggregation strategy to combine these targets using $\mathbf{w}$. The specific optimization targets and aggregation strategies vary across different approaches. More specifically, DPO soups optimizes a separate LLM for each objective by DPO and then aggregate them at the model parameter level by weight merging. MODPO trains DPO LLMs as proxy reward models for certain objectives and aggregates them at the loss level by interpolating the weighted reward differences as margins into the DPO loss function. The naive baseline, DPO Loss Weighting (LW), computes the DPO loss for each objective and aggregates them at the loss level by a weighted sum.

However, we observe that this paradigm is eas-

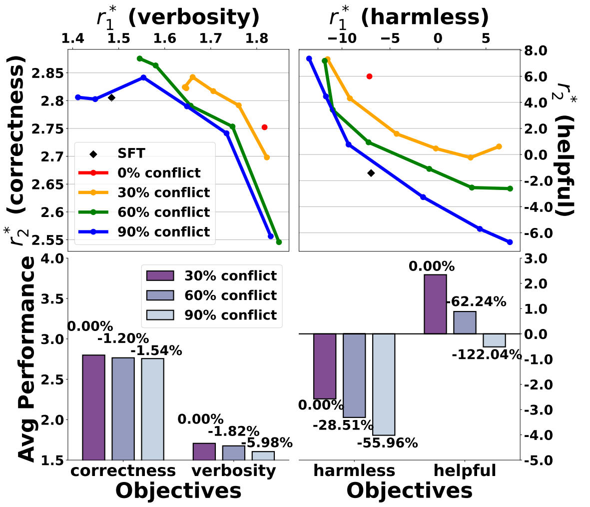

Figure 2: The impact of preference conflicts on Pareto Front optimization. Experiments are conducted on DPO soups with Alpaca-7B.

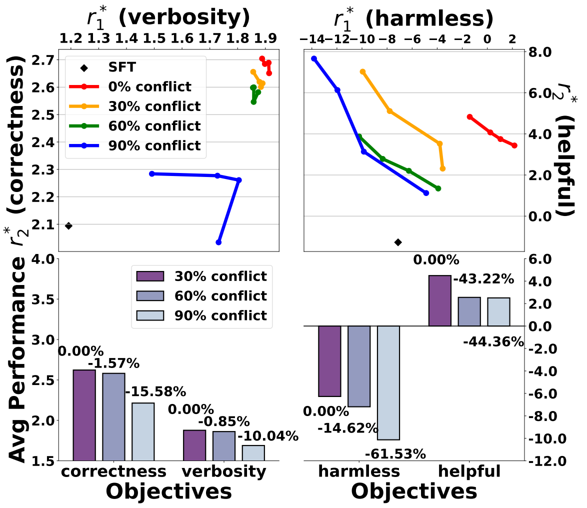

Figure 3: The impact of preference conflicts on Pareto Front optimization. Experiments are conducted on MODPO with Alpaca-7B.

ily hindered by preference conflicts in the data. Preference conflict refers to the instance where different objectives assign different preference labels, defined as: $\{(\mathbf{x}, \mathbf{y}_1, \mathbf{y}_2, p_1, ..., p_N) | \exists i, j \in [1, N+1], p_i \neq p_j\}$. Aligning on these instances introduces contradictory optimization targets, disrupting the learning for individual objectives and ultimately hindering Pareto Front optimization.

To illustrate this issue, we take an example on the naive DPO LW method with $N = 2$, where the loss is defined as a weighted sum of the DPO losses on each objective: $\mathcal{L}_{\text{DPO\_LW}} = w_1 \mathcal{L}_1 + w_2 \mathcal{L}_2$. With conflicting preferences, *i.e.,* $p_1 \neq p_2$, the losses $\mathcal{L}_1$ and $\mathcal{L}_2$ are opposite, $\mathcal{L}_1 = -\mathcal{L}_2$, pulling the optimization in opposing directions. As a result, optimizing $\mathcal{L}_{\text{DPO\_LW}}$ leads to conflicting gradient updates, preventing the LLM from effectively aligning with each objective and ultimately degrading the Pareto Front. This issue extends to other DPO-based MOA approaches and holds for larger values of N. To further illustrate this issue, we conduct the following controlled experiment.

Evaluation Protocol To investigate ***the impact of preference conflicts on Pareto Front optimization***, we conduct experiments by controlling the ratio of preference conflicts in alignment. Specifically, we subsample equal-sized subsets from $\mathcal{D}$ with 0%, 30%, 60%, and 90% of conflicting preferences, and compare their optimized Pareto Front. We examine this problem from multiple perspectives. Firstly, we evaluate two prominent DPO-based MOA approaches, MODPO and DPO soups. Secondly, we utilize two widely-used multi-objective preference datasets with two different pairs of objectives. We choose the *correctness* and *verbosity* from HelpSteer, (Wang et al., 2024b), *harmless* and *helpful* from BeaverTails (Ji et al., 2023). Finally, we experiment with different backbone LLMs, including Alpaca-7B (Taori et al., 2023) and a supervised fine-tuned LLaMA-2-7B (Touvron et al., 2023). More details on the methods, datasets and backbone LLMs can be found in Section 4 and Appendix A.

Results on Different Methods and Objectives Figure 2 shows the Pareto Fronts for DPO soups under varying conflict ratios of the alignment data. We also show the average performance decrease over different preference weights for each objective. Corresponding results on MODPO is shown in Figure 3. We can observe that (1) as the ratio of conflicts in the training data increases, the Pareto Fronts gradually move downwards, showing significant performance decreases. This phenomenon holds for all datasets and methods, which validates the existence of the issue. For DPO soups, when the conflict ratio reaches 90%, the Pareto Front even approaches the performance on the original LLM without alignment (denoted as SFT), showing severe alignment problem. (2) All objectives incur significant average performance decreases on both methods. *Helpful* and *harmless* have more significant performance decreases than *correctness* and *verbosity*, which may be related to the more conflicting nature of the definition of these objectives. (3) However, reducing the conflict ratio of the data generally hurts the steerability of the Pareto

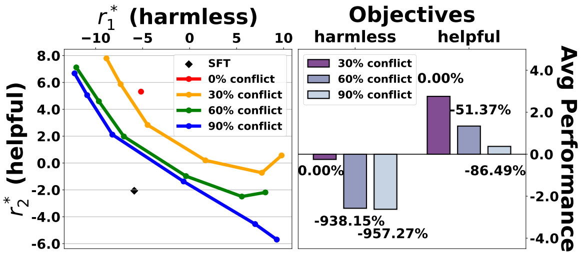

Figure 4: The impact of preference conflicts on Pareto Front optimization. Results of BeaverTails with DPO soups on LLaMA-2-7B.

Dataset	HelpSteer			BeaverTails
# Objectives	3	4	5	2
Conflict Ratio (%)	11.86	15.89	17.94	53.83

Table 1: Statistics on the conflict ratio in Helpsteer and BeaverTails datasets.

Fronts, meaning that the performance ranges of the two objectives across preference weights get tighter under smaller conflict ratios. We conjecture that controlling the conflict ratio in the data may hurt the versatility of the data, thus hindering the optimization of single objectives toward higher performance under certain preference weights.

Results on Different Backbone LLMs To examine the consistency of this issue on different backbone LLMs, we utilize a supervised fine-tuned LLaMA-2-7B as an additional backbone LLM. The results of BeaverTails on DPO soups is shown in Figure 4. We can observe that even though LLaMA-2-7B achieves better performance on both objectives than Alpaca-7B (compared with Figure 2), the conflict ratio consistently affects the Pareto Front, showing that stronger backbone LLM will also be affected by the preference conflicts, further demonstrating the existence of the issue.

Statistics on the Percentage of Conflicting Data We have conducted statistics on the ratio of preference conflicts in these datasets, as shown in Table 1. For BeaverTails, we calculate the conflict ratio for the two objectives. For HelpSteer, we vary the number of objectives from three to five. We can observe that the *helpful* and *harmless* in BeaverTails has more than 50% of conflict, showing strong conflicting nature. Statistics for HelpSteer are all more than 10%, and increasing the number of objectives further increases the conflict ratio. The statistics reveals the severity of the preference conflicts in current datasets, stressing the need for mitigation.

3 Method

In this section, we introduce our SIPO framework (*cf.* Figure 5), which leverages self-generated Pareto-optimal responses to mitigate the impact of preference conflicts. We will introduce the definition of Pareto-optimal responses (§ 3.1), and detail the SIPO framework design (§ 3.2, § 3.3).

3.1 Pareto-Optimal Responses

To solve the issue of preference conflicts, we resort to Pareto-optimal responses. For an instance $(\mathbf{x}, \mathbf{y}_{-1}, \mathbf{y}_1, p_1, ..., p_N) \in \boldsymbol{\mathcal{D}}$ with conflicting preferences, the Pareto-optimal responses $\mathbf{y}_c$ are defined as those responses that outperform both $\mathbf{y}_{-1}$ and $\mathbf{y}_1$ across all objectives:

$$\begin{aligned} \mathbf{y}_c = \{\mathbf{y} | \forall i,\ & r_i^*(\mathbf{x}, \mathbf{y}) > r_i^*(\mathbf{x}, \mathbf{y}_1) \text{ and} \\ & r_i^*(\mathbf{x}, \mathbf{y}) > r_i^*(\mathbf{x}, \mathbf{y}_{-1})\}. \end{aligned} \tag{1}$$

$\mathbf{y}_c$ incurs no preference conflicts with $\mathbf{y}_1$ and $\mathbf{y}_{-1}$, thereby avoiding the issues outlined in Section 2. $\mathbf{y}_c$ also has better quality in terms of all objectives than $\mathbf{y}_1$ and $\mathbf{y}_{-1}$, also facilitating the achievement of a more optimal Pareto Front.

3.2 SIPO Framework: Responses Generation

Given that human annotation of Pareto-optimal responses is prohibitively expensive and infeasible for large-scale datasets, our SIPO framework is designed to autonomously generate and leverage Pareto-optimal responses. We initially align N policy LLMs to capture each objective using DPO, denoted as $\mathbf{\Pi} = \{\pi_{\theta_i}\}_{i=1}^N$.

$$\begin{aligned} \theta_i = \arg\min_{\theta} -\mathbb{E}_{\boldsymbol{\mathcal{D}}}\Bigg[\log \sigma\Bigg(& p_i \beta \frac{\pi_\theta(\mathbf{y}_1|\mathbf{x})}{\pi_{\text{ref}}(\mathbf{y}_1|\mathbf{x})} \\ & - p_i \beta \frac{\pi_\theta(\mathbf{y}_{-1}|\mathbf{x})}{\pi_{\text{ref}}(\mathbf{y}_{-1}|\mathbf{x})}\Bigg)\Bigg], \end{aligned} \tag{2}$$

where π_{ref} denotes the reference LLM. Then, we aim for generating Pareto-optimal responses with three stages, *Sampling*, *Refinement* and *Filtering*.

Stage 1: Sampling For the sampling stage, we aim to generate diverse high-quality responses based on the aligned policy LLMs. To enhance sampling diversity, we apply a set of preference weights $\mathbf{W} = \{\mathbf{w}_m\}_{m=1}^M$ and generate responses under each $\mathbf{w}_m$ for $\mathbf{x}$, denoted as $\mathbf{y}_m^s$. To ensure sampling quality, we utilizing the outstanding decoding-based method MOD (Shi et al., 2024) to sample responses

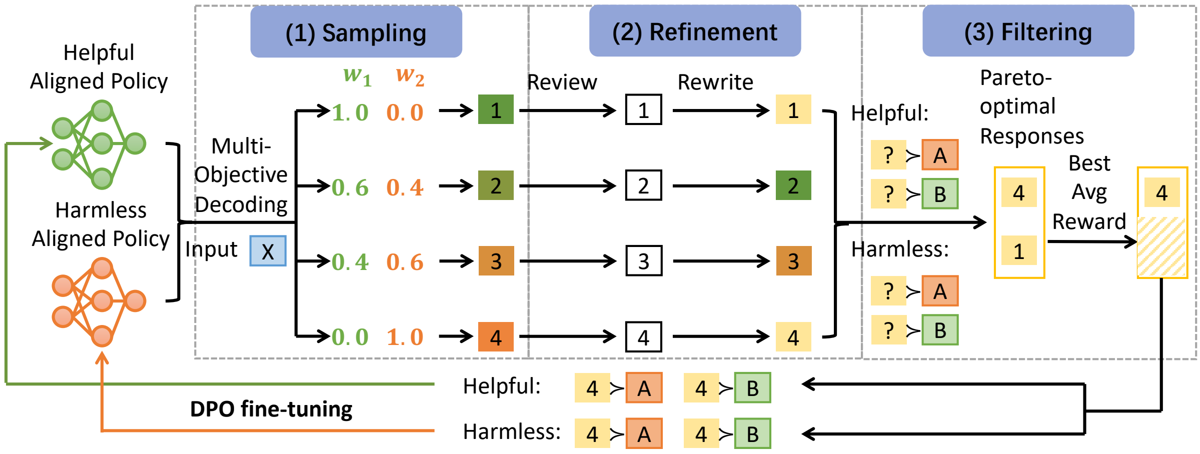

Figure 5: Illustration of our proposed SIPO framework.

from $\mathbf{\Pi}$ under a given $\mathbf{w}_m$. Denoting the MOD decoding function as $f^d(\cdot)$,

$$\mathbf{y}_m^s = f^d(\mathbf{\Pi}, \mathbf{w}_m, \mathbf{x}). \tag{3}$$

Stage 2: Refinement To further enhance the quality of the sampled $\mathbf{y}_m^s$, we employ a self-refinement strategy, prompting LLM to review the flaws of $\mathbf{y}_m^s$ from the perspectives of different objectives and then revise it. Firstly, we employ an evaluator LLM to analyze and generate reviews from different perspectives. This evaluator needs to possess the ability of different objectives, thus we implement MOD on the policy LLMs with a preference weight $\mathbf{w}_e$ to mix the objectives.

$$\mathbf{y}_m^v = f^d(\mathbf{\Pi}, \mathbf{w}_e, [\mathbf{p}_v, \mathbf{x}, \mathbf{y}_m^s]), \tag{4}$$

where $\mathbf{y}_m^v$ is the generated review for $\mathbf{y}_m^s$, $\mathbf{p}_v$ is the instruction and in-context examples guiding the review generation. Then, we revise the response $\mathbf{y}_m^s$ based on $\mathbf{y}_m^v$ using the original sampling policy of $\mathbf{y}_m^s$ to obtain an enhanced response $\mathbf{y}_m^a$.

$$\mathbf{y}_m^a = f^d(\mathbf{\Pi}, \mathbf{w}_m, [\mathbf{p}_a, \mathbf{x}, \mathbf{y}_m^v, \mathbf{y}_m^s]), \tag{5}$$

where $\mathbf{p}_a$ is the instruction and in-context examples. We hope choosing different weights of $\mathbf{w}_e$ and $\mathbf{w}_m$ for reviewing and rewriting policies can leverage their joint effectiveness.

Stage 3: Filtering After obtaining the sampled responses, we apply a filtering stage to ensure the Pareto-optimality over the original responses $\mathbf{y}_1$ and $\mathbf{y}_{-1}$, as defined in Eq. (1). In the absence of the ground-truth reward functions $\mathbf{r}^*(\cdot, \cdot)$, we leverage the implicit reward function from DPO models, *i.e.,* $\mathbf{\Pi}$, to estimate the rewards.

$$\hat{r}_i(\mathbf{x}, \mathbf{y}_m^a) = \beta \log \pi_{\theta_i}(\mathbf{y}_m^a|\mathbf{x}) + \beta \log Z(\mathbf{x}), \tag{6}$$

where $Z(\mathbf{x})$ is a normalization constant independent of the responses, allowing us to disregard it. Each $\mathbf{y}_m^a$ obtains a set of rewards, $\hat{\mathbf{r}}_{\mathbf{\Pi}}(\mathbf{x}, \mathbf{y}_m^a) = [\hat{r}_1(\mathbf{x}, \mathbf{y}_m^a), ..., \hat{r}_N(\mathbf{x}, \mathbf{y}_m^a)]^\intercal$. Apart from the DPO models in $\mathbf{\Pi}$, we also utilize M additional policy LLMs combined under preference weights $\mathbf{W}$ to further calculate the rewards on mixed objectives, denoted as $\hat{\mathbf{r}}_{\mathbf{W}}(\mathbf{x}, \mathbf{y}_m^a)$. The combined policy LLMs are obtained via model weight merging following DPO soups (Ramé et al., 2023).

Finally, we select the Pareto-optimal $\mathbf{y}_m^a$ with all rewards $\hat{\mathbf{r}}_{\mathbf{\Pi}}$ and $\hat{\mathbf{r}}_{\mathbf{W}}$ larger than the original responses. If multiple $\mathbf{y}_m^a$ for a single $\mathbf{x}$ satisfy such constraints, we choose the one with the largest average reward as $\mathbf{y}_c$.

$$\mathbf{y}_c = \{\mathbf{y}_m^a | \hat{r}(\mathbf{x}, \mathbf{y}_m^a) > \hat{r}(\mathbf{x}, \mathbf{y}_1), \text{and} \\ \hat{r}(\mathbf{x}, \mathbf{y}_m^a) > \hat{r}(\mathbf{x}, \mathbf{y}_{-1}), \ \forall \hat{r} \in \hat{\mathbf{r}}_{\mathbf{W}} \cup \hat{\mathbf{r}}_{\mathbf{\Pi}}\}. \tag{7}$$

3.3 SIPO Framework: Fine-Tuning

After obtaining $\mathbf{y}_c$, we update the policy LLMs to reduce the effect of preference conflicts and improve Pareto Front. Firstly, based on Eq. (1), we utilize two preference relationships $\mathbf{y}_c \succ \mathbf{y}_{-1}$, and $\mathbf{y}_c \succ \mathbf{y}_1$, and construct new preference dataset as $\mathcal{D}^c = \{(\mathbf{x}, \mathbf{y}_c, \mathbf{y}_l)\}$, where $\mathbf{y}_l$ represents either $\mathbf{y}_{-1}$ or $\mathbf{y}_1$. These new preferences are non-conflicting and prevent forgetting on the original responses. We validate the rationality of the preference design with experiments in Section 4.2. Then, we perform DPO fine-tuning on $\mathcal{D}^c$ for policy LLMs. Following Pang et al. (2024b), we also add an NLL loss term to prevent forgetting. The objective is defined as follows, where α is the weight for NLL loss.

$$\theta_i' = \arg\min_{\theta} -\mathbb{E}_{\mathcal{D}^c}\left[\log\sigma\left(p_i\beta\frac{\pi_\theta(\mathbf{y}_c|\mathbf{x})}{\pi_{\theta_i}(\mathbf{y}_c|\mathbf{x})} - p_i\beta\frac{\pi_\theta(\mathbf{y}_l|\mathbf{x})}{\pi_{\theta_i}(\mathbf{y}_l|\mathbf{x})}\right) - \alpha\frac{\log\pi_{\theta_i}(\mathbf{y}^c|\mathbf{x})}{|\mathbf{y}^c|}\right]. \quad (8)$$

For final evaluation, we primarily apply the outstanding decoding method MOD on the updated policy LLMs $\mathbf{\Pi}' = \{\pi_{\theta_i'}\}_{i=1}^N$. In our experiments (*cf.* Section 4.2), we also combine SIPO evaluation with DPO soups to show its adaptability.

4 Experiments

Experimental Setup We conduct experiments on two widely-used MOA datasets. **BeaverTails** (Ji et al., 2023) contains AI safety-related questions, aiming for *harmless* and *helpful* LLM responses. We utilize the BeaverTails-10K subset and split the training and validation data as 9:1, and utilize an additional split from the BeaverTails-30K dataset as the test data. **HelpSteer** aims to promote response helpfulness, where we focus on two objectives, *correctness*, *i.e.,* factuality precision and relevance, and *verbosity*, *i.e.,* response length and level of detail. Since HelpSteer is not formulated as our definition of $\mathcal{D}$, we manually transform the dataset to follow the definition. Dataset preprocessing details and statistics can be found in Appendix A.1.

For backbone LLMs, we mainly utilize supervised fine-tuned LLaMA-2-7B (Touvron et al., 2023), denoted as LLaMA-2-7B-sft. We fine-tune all the responses in the training dataset to obtain a LLaMA-2-7B-sft for each dataset. We also conduct experiments on Alpaca-7B (Taori et al., 2023) to show the applicability of SIPO on different LLMs (see more details in Appendix A.3).

Compared Methods We primarily focus on the comparison with DPO-based MOA approaches.

- **MODPO** (Zhou et al., 2024), a state-of-the-art DPO-based MOA approach which trains DPO models as reward models for $N-1$ objectives, and integrates the weighted reward differences of responses as margins into the DPO loss of the final objective.
- **DPO soups** (Ramé et al., 2023), the DPO version of model soup, which performing model weight merging on DPO models of each objective by the preference weight.

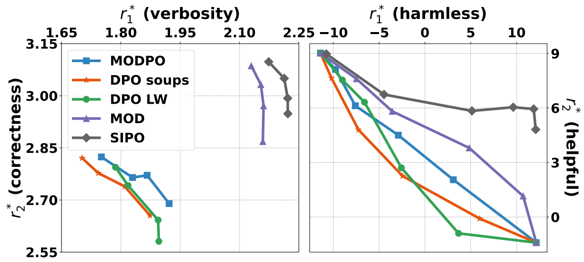

Figure 6: Pareto Fronts of compared methods on HelpSteer (**left**) and BeaverTails (**right**).

- **DPO LW** (Zhou et al., 2024), the naive DPO-based MOA baseline, which linearly combines the DPO losses for each objective by the preference weight as the final DPO loss.

In addition, we also include an outstanding decoding-time alignment method **MOD** (Shi et al., 2024), which combines the logits of N DPO models by the preference weight for decoding. For all compared methods, we utilize six different preference weights $[w, 1-w], w \in \{0, 0.2, 0.4, 0.6, 0.8, 1.0\}$. For HelpSteer, we show the middle four weights for better visualization. See full implementation details in Appendix A.3.

Evaluation Metrics Following the practice of MODPO and MOD, we utilize the standard-released reward models as the ground-truth reward models to evaluate the LLM alignment performance. For HelpSteer, we utilize the reward model released by Wang et al. (2024b). For BeaverTails, we utilize the standard released usefulness and cost reward models. See Appendix A.2 for details.

4.1 Results

Performance comparison on the Pareto Fronts of all compared methods is presented in Figure 6, with full results for HelpSteer shown in Figure 12. We can observe that (1) on both datasets, the Pareto Front of SIPO largely outperforms all baseline methods, demonstrating its effectiveness in achieving superior Pareto Front. (2) For DPO-based baseline, MODPO generally outperforms DPO soups and DPO LW, which is in line with the results of MODPO. (3) The decoding-based MOD outperforms all DPO-based methods, showing the great potential of LLM to generate outstanding responses through effective decoding strategy. (4) SIPO achieves larger performance improvement over MOD on BeaverTails than HelpSteer (*cf.* Figure 13), potentially because BeaverTails has larger proportion of preference conflicts which is tack-

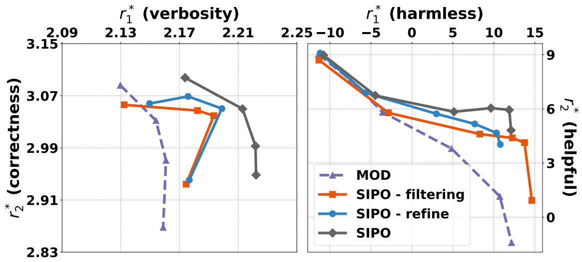

Figure 7: Ablation studies on SIPO.

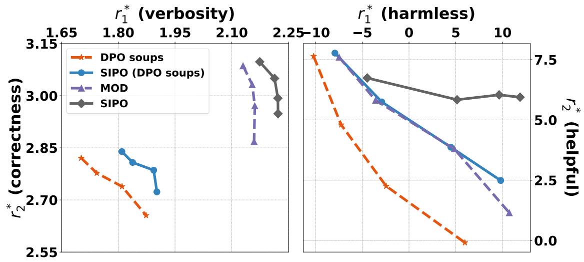

Figure 9: Combination of SIPO with different DPO-based MOA baselines.

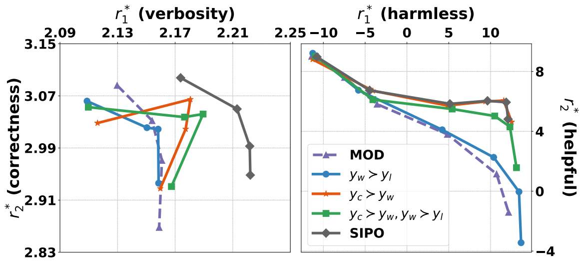

Figure 8: Performance comparison on alternative preference design in re-alignment.

led by SIPO. (5) Particularly, the improvement on BeaverTails is mostly on the *helpful* side, while for HelpSteer both sides improve. This corresponds to the improvement between MOD and DPO-based approaches since the Pareto-optimal response sampling is based on MOD. This might also be related to the stronger conflicts between *helpful* and *harmless*, making simultaneous improvement difficult.

4.2 In-depth Analysis

Ablation Studies To validate the effectiveness of each component within our framework, we conduct the following ablation studies: removing the refinement stage, denoted as *SIPO - refine*, and removing the filtering stage by randomly subsampling the refined responses to the same size, denoted as *SIPO - filter*. As shown in Figure 7, we can observe that (1) removing each component in our framework largely decreases the Pareto Front, validating their effectiveness. (2) Removing the filtering stage causes larger performance decrease on both datasets than removing the refinement stage. On HelpSteer, the performance of *SIPO - filter* even gets lower than MOD under some preference weights, highlighting the necessity of ensuring response quality to meet the Pareto-optimal criteria. (3) The refinement stage has larger improvement on HelpSteer than BeaverTails, potentially related to the larger improvement by MOD on HelpSteer. See Appendix E for a case study on SIPO components.

Rationality of Preference Design We validate the rationality of our preference design as in Section 3. We consider the following four alternative preferences, denoting the preferred and dispreferred responses for a certain objective as $\mathbf{y}_w$ and $\mathbf{y}_l$. (1) $\mathbf{y}_c \succ \mathbf{y}_w$, only learning the preference with the preferred response, (2) $\mathbf{y}_c \succ \mathbf{y}_w, \mathbf{y}_w \succ \mathbf{y}_l$, learning a sequential of preferences, (3) $\mathbf{y}_w \succ \mathbf{y}_l$, ablating the preference of $\mathbf{y}_c$, The results are shown in Figure 8. We can observe that (1) $\mathbf{y}_w \succ \mathbf{y}_l$ achieves the worst performance due to fine-tuning on the conflicting preference, even lower than MOD in HelpSteer. (2) $\mathbf{y}_c \succ \mathbf{y}_w, \mathbf{y}_w \succ \mathbf{y}_l$ is lower than SIPO, potentially related to fine-tuning on the conflicting preference $\mathbf{y}_w \succ \mathbf{y}_l$. (3) $\mathbf{y}_c \succ \mathbf{y}_w$ achieves comparable performance on BeaverTails, while lower than SIPO on HelpSteer. We think this is because fine-tuning on $\mathbf{y}_c \succ \mathbf{y}_w$ sometimes may lead to forgetting on the original $\mathbf{y}_w \succ \mathbf{y}_l$ preference. Therefore, we incorporate this preference as a non-conflicting $\mathbf{y}_c \succ \mathbf{y}_l$ in SIPO to avoid forgetting.

Combination with Other DPO-based Approaches To demonstrate the effectiveness of combining SIPO with other DPO-based approaches, we combine SIPO with DPO soups. The results are shown in Figure 9. We can observe that SIPO largely improves the performance of DPO soups on both datasets, showing its applicability and strong effectiveness on different approaches.

Studies on Resolving Preference Conflicts We examine two research questions on preference conflicts. Firstly, how well does the sampled $\mathbf{y}_c$ resolve preference conflicts? We compare the average reward of preferred and dispreferred responses with $\mathbf{y}_c$. Table 2 shows that $\mathbf{y}_c$ has significantly better rewards for both objectives on HelpSteer, demonstrating its Pareto-optimality. On BeaverTails, $\mathbf{y}_c$ enhances *harmless* but slightly decreases *helpful* by 0.4, while SIPO still improving the Pareto Front in the *helpful* dimension. However, the improvements

y	BeaverTails r_1^*(helpful)	r_2^*(harmless)	y	HelpSteer r_1^*(corr)	r_2^*(verb)
helpful	**3.8**	-19.9	corr	<u>2.9</u>	1.6
harmless	-1.0	<u>-11.5</u>	verb	2.7	<u>1.8</u>
SIPO	<u>3.4</u>	**-0.4**	SIPO	**3.1**	**2.3**
RI	-11.75%	96.83%	RI	6.03%	27.44%

Table 2: Average response reward comparison between SIPO and the original responses. Bold font and underline indicate the best and second-best results. RI denotes the relative improvement to the second-best results.

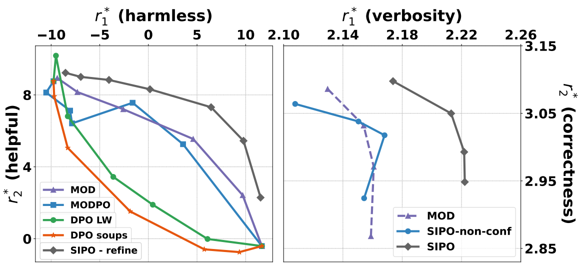

Figure 10: Performance comparison on Alpaca-7B (**Left**). Experiments of adapting SIPO on non-conflicting preferences (**Right**).

are imbalanced between objectives, indicating the need for further optimization.

Second, how effective is SIPO on non-conflicting preferences? We replace conflicting preferences with non-conflicting ones and analyze the results (*cf.* Figure 10). SIPO does not enhance performance in this scenario and sometimes performs worse than the non-training MOD, confirming its suitability for resolving conflicts.

Extension to Multi-Round Self-Improvement Since the self-improvement framework can be conducted for multiple rounds, we perform an additional round of SIPO on BeaverTails to observe its continued improvement. From Figure 11, we observe a general performance improvement in the second-round model compared to the first-round model. In addition, the average response rewards continue to show a significant increase over the first round. Notably, the helpfulness reward exhibits a substantial relative improvement of 85.29% (3.4 $\rightarrow$ 6.3), in contrast to -11.75% in the first round (*cf.* Table 2), highlighting the continued potential for improvement through additional rounds.

Generalization to Different Backbone LLMs To assess SIPO's effectiveness across LLMs, we use Alpaca-7B as the backbone. Due to its context length limitation, we apply *SIPO - refine*, the closest variation of SIPO. As shown in the left of Figure 10, *SIPO - refine* consistently outperforms all baselines on BeaverTails, demonstrating its effectiveness across different LLMs. Refer to Appendix B for results on LLM of smaller size.

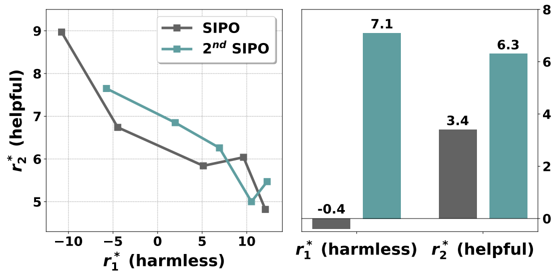

Figure 11: Performance (**left**) and average response rewards (**right**) of second round SIPO on BeaverTails.

5 Related Work

Learning from Human Feedback Learning from human feedback is essential for aligning LLMs with human values, enhancing safety, helpfulness, and factual accuracy (Ji et al., 2023; Wang et al., 2024d; Lin et al., 2024; Cui et al., 2024; Liu et al., 2025; Xu et al., 2025). A key approach is RLHF (Ouyang et al., 2022; Stiennon et al., 2020; Bai et al., 2022; Touvron et al., 2023), where a reward model learns human preferences, and RL methods like PPO (Schulman et al., 2017) update the LLM accordingly. To improve the efficiency and stability of RL-based methods, DPO-based methods (Rafailov et al., 2023; Ethayarajh et al., 2024; Wang et al., 2024a; Azar et al., 2024) bypass reward modeling by directly learning from preference data. More recently, AI-generated feedback has been explored to reduce human labeling efforts (Lee et al., 2024; Yu et al., 2024).

Multi-Objective Alignment of LLM Human preferences are inherently heterogeneous and better modeled as multi-dimensional rather than a single-dimensional preferences. RL-based methods (Ramé et al., 2023; Jang et al., 2023; Zhong et al., 2024a; Wang et al., 2024b,c) learns proxy reward model for each objective and update LLMs via RL, often aggregating preferences at the parameter level to reduce computational cost. DPO-based approaches, such as MODPO (Zhou et al., 2024), aim to reduce reliance on multiple proxy rewards and RL optimization while maintaining alignment efficiency, as detailed in Section 2. Besides, decoding-based methods offer alternative MOA strategies, such as logit manipulation (Shi et al., 2024; Liu et al., 2024a; Xu et al., 2024; Chen et al., 2025) and prompt-based techniques (Fu et al., 2024). Additionally, some studies explore other constraints among objectives (Liu et al., 2024b; Zhang et al., 2024) or conditional generation (Guo et al., 2024; Yang et al., 2024; Ren et al., 2024), which remain orthogonal to our setting.

Self-Improvement LLMs can self-improve (Huang et al., 2022; Wang et al., 2023), reducing the reliance on external data or feedback (Deng et al., 2024; Gou et al., 2024) through self-data generation and self-feedback (Huang et al., 2022; Wang et al., 2023; Pang et al., 2024a; Yuan et al., 2024). This technique has also been integrated with DPO (Pang et al., 2024c; Xu et al., 2023; Xiong et al., 2024). For application on MOA, our approach shares similarities with Wang et al. (2024b) but differs in that we focus on sampling Pareto-optimal responses.

6 Conclusion

This paper addressed the negative impact of preference conflicts on achieving a superior Pareto Front in DPO-based MOA. Through extensive analysis and experiments, we revealed the impact of preference conflicts on Pareto Front optimization. To mitigate this issue, we proposed SIPO, a framework that automatically generates and leverages Pareto-optimal responses to resolve preference conflicts, which outperformed baseline methods in achieving a superior Pareto Front. In the future, we plan to extend our experiments to more objectives and additional DPO-based methods. We will also explore how to improve the efficiency of obtaining Pareto-optimal responses to reduce the cost.

Limitation

Experiments on More Objectives and DPO-based Approaches Our experiments is conducted primarily on two objectives for each dataset, and we combine SIPO with MOD and DPO soups. We could extend SIPO to more than two objectives per dataset (where we present our experiment with three objectives in Appendix B), and to more DPO-based approaches such as MODPO. We can also extend SIPO to more backbone LLMs of various sizes (where we present the results on a 3B model in Appendix B). We leave a comprehensive exploration of these extensions to future work.

More Validation on the Effectiveness of Pareto-optimal Response Despite using Pareto-optimal responses, we also consider another potential setting to resolve preference conflicts. Given $\mathbf{a} \succ \mathbf{b}$ on objective 1 and $\mathbf{a} \prec \mathbf{b}$ on objective 2, we consider sampling two responses, $\mathbf{c}$ and $\mathbf{d}$, where $\mathbf{c} \succ \mathbf{a}$ on objective 1 and $\mathbf{d} \succ \mathbf{b}$ on objective 2. $\mathbf{c}$ and $\mathbf{d}$ are not Pareto optimal responses, but it is quite possible that this setting can improve performance on each objective, thus improve the Pareto Front. However, we think that these setting is not as effective as Pareto-optimal responses in pushing Pareto Fronts. In the future, we will explore the comparison of this setting with our SIPO.

Generating Pareto-optimal Responses with Additional Stronger LLMs In this work, we employ self-improvement paradigm without resorting to additional human-labeled data or data labeled by stronger LLMs. Distilling Pareto-optimal response from stronger LLMs to improve Pareto Front may be another direction in the field of MOA, which we leave as future work.

Additional Computational Overhead The proposed SIPO framework involves response sampling, refinement, filtering, and fine-tuning, which could introduce computational overhead compared to simpler methods. However, we believe this trade-off is justified for the reasons outlined in Appendix D. In future work, we plan to explore strategies to reduce this overhead while maintaining the benefits of self-improvement.

Ethical Consideration

The examples shown in Appendix C and E may contain harmful or offensive contents.

Acknowledgments

This work is supported by the NExT Research Center. We thank the reviewers for their constructive feedback.

References

Mohammad Gheshlaghi Azar, Zhaohan Daniel Guo, Bilal Piot, Rémi 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, 2-4 May 2024, Palau de Congressos, Valencia, Spain*, volume 238 of *Proceedings of Machine Learning Research*, pages 4447–4455. PMLR.

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, Nicholas Joseph, Saurav Kadavath, Jackson Kernion, Tom Conerly, Sheer El Showk, Nelson Elhage, Zac Hatfield-Dodds, Danny Hernandez, Tristan Hume, Scott Johnston, Shauna Kravec, Liane Lovitt, Neel Nanda, Catherine Olsson, Dario Amodei, Tom B. Brown, Jack Clark, Sam McCandlish, Chris Olah, Benjamin Mann, and Jared Kaplan. 2022. Training a helpful and harmless assistant with reinforcement learning from human feedback. *CoRR*, abs/2204.05862.

Ruizhe Chen, Xiaotian Zhang, Meng Luo, Wenhao Chai, and Zuozhu Liu. 2025. PAD: Personalized alignment at decoding-time. In *The Thirteenth International Conference on Learning Representations*.

Ganqu Cui, Lifan Yuan, Ning Ding, Guanming Yao, Bingxiang He, Wei Zhu, Yuan Ni, Guotong Xie, Ruobing Xie, Yankai Lin, Zhiyuan Liu, and Maosong Sun. 2024. ULTRAFEEDBACK: boosting language models with scaled AI feedback. In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*. OpenReview.net.

Boyi Deng, Wenjie Wang, Fengbin Zhu, Qifan Wang, and Fuli Feng. 2024. Cram: Credibility-aware attention modification in llms for combating misinformation in rag. *Preprint*, arXiv:2406.11497.

Kawin Ethayarajh, Winnie Xu, Niklas Muennighoff, Dan Jurafsky, and Douwe Kiela. 2024. KTO: model alignment as prospect theoretic optimization. *CoRR*, abs/2402.01306.

Tingchen Fu, Yupeng Hou, Julian J. McAuley, and Rui Yan. 2024. Unlocking decoding-time controllability: Gradient-free multi-objective alignment with contrastive prompts. *CoRR*, abs/2408.05094.

Zhibin Gou, Zhihong Shao, Yeyun Gong, Yelong Shen, Yujiu Yang, Nan Duan, and Weizhu Chen. 2024. Critic: Large language models can self-correct with tool-interactive critiquing. *Preprint*, arXiv:2305.11738.

Yiju Guo, Ganqu Cui, Lifan Yuan, Ning Ding, Zexu Sun, Bowen Sun, Huimin Chen, Ruobing Xie, Jie Zhou, Yankai Lin, Zhiyuan Liu, and Maosong Sun. 2024. Controllable preference optimization: Toward controllable multi-objective alignment. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, EMNLP 2024, Miami, FL, USA, November 12-16, 2024*, pages 1437–1454. Association for Computational Linguistics.

Jiaxin Huang, Shixiang Shane Gu, Le Hou, Yuexin Wu, Xuezhi Wang, Hongkun Yu, and Jiawei Han. 2022. Large language models can self-improve. *CoRR*, abs/2210.11610.

Joel Jang, Seungone Kim, Bill Yuchen Lin, Yizhong Wang, Jack Hessel, Luke Zettlemoyer, Hannaneh Hajishirzi, Yejin Choi, and Prithviraj Ammanabrolu. 2023. Personalized soups: Personalized large language model alignment via post-hoc parameter merging. *CoRR*, abs/2310.11564.

Jiaming Ji, Mickel Liu, Josef Dai, Xuehai Pan, Chi Zhang, Ce Bian, Boyuan Chen, Ruiyang Sun, Yizhou Wang, and Yaodong Yang. 2023. Beavertails: Towards improved safety alignment of LLM via a human-preference dataset. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

Harrison Lee, Samrat Phatale, Hassan Mansoor, Thomas Mesnard, Johan Ferret, Kellie Lu, Colton Bishop, Ethan Hall, Victor Carbune, Abhinav Rastogi, and Sushant Prakash. 2024. RLAIF vs. RLHF: scaling reinforcement learning from human feedback with AI feedback. In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*. OpenReview.net.

Sheng-Chieh Lin, Luyu Gao, Barlas Oguz, Wenhan Xiong, Jimmy Lin, Scott Yih, and Xilun Chen. 2024. FLAME : Factuality-aware alignment for large language models. In *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*.

Kai Liu, Ze Chen, Zhihang Fu, Wei Zhang, Rongxin Jiang, Fan Zhou, Yaowu Chen, Yue Wu, and Jieping Ye. 2025. Structure-aware domain knowledge injection for large language models. *Preprint*, arXiv:2407.16724.

Zhixuan Liu, Zhanhui Zhou, Yuanfu Wang, Chao Yang, and Yu Qiao. 2024a. Inference-time language model alignment via integrated value guidance. In *Findings of the Association for Computational Linguistics: EMNLP 2024, Miami, Florida, USA, November 12-16, 2024*, pages 4181–4195. Association for Computational Linguistics.

Zixuan Liu, Xiaolin Sun, and Zizhan Zheng. 2024b. Enhancing LLM safety via constrained direct preference optimization. *CoRR*, abs/2403.02475.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F. Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.

Jing-Cheng Pang, Pengyuan Wang, Kaiyuan Li, Xiong-Hui Chen, Jiacheng Xu, Zongzhang Zhang, and Yang

Yu. 2024a. Language model self-improvement by reinforcement learning contemplation. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

Richard Yuanzhe Pang, Weizhe Yuan, Kyunghyun Cho, He He, Sainbayar Sukhbaatar, and Jason Weston. 2024b. Iterative reasoning preference optimization. *CoRR*, abs/2404.19733.

Richard Yuanzhe Pang, Weizhe Yuan, He He, Kyunghyun Cho, Sainbayar Sukhbaatar, and Jason Weston. 2024c. Iterative reasoning preference optimization. In *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D. Manning, Stefano Ermon, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

Alexandre Ramé, Guillaume Couairon, Corentin Dancette, Jean-Baptiste Gaya, Mustafa Shukor, Laure Soulier, and Matthieu Cord. 2023. Rewarded soups: towards pareto-optimal alignment by interpolating weights fine-tuned on diverse rewards. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

Yinuo Ren, Tesi Xiao, Michael Shavlovsky, Lexing Ying, and Holakou Rahmanian. 2024. Hyperdpo: Hypernetwork-based multi-objective fine-tuning framework. *CoRR*, abs/2410.08316.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *CoRR*, abs/1707.06347.

Ruizhe Shi, Yifang Chen, Yushi Hu, Alisa Liu, Hanna Hajishirzi, Noah A. Smith, and Simon S. Du. 2024. Decoding-time language model alignment with multiple objectives. In *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*.

Nisan Stiennon, Long Ouyang, Jeffrey Wu, Daniel M. Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul F. Christiano. 2020. Learning to summarize with human feedback. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2023. Stanford alpaca: An instruction-following llama model. `https://github.com/tatsu-lab/stanford_alpaca`.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton-Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurélien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open foundation and fine-tuned chat models. *CoRR*, abs/2307.09288.

Chaoqi Wang, Yibo Jiang, Chenghao Yang, Han Liu, and Yuxin Chen. 2024a. Beyond reverse KL: generalizing direct preference optimization with diverse divergence constraints. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

Haoxiang Wang, Yong Lin, Wei Xiong, Rui Yang, Shizhe Diao, Shuang Qiu, Han Zhao, and Tong Zhang. 2024b. Arithmetic control of llms for diverse user preferences: Directional preference alignment with multi-objective rewards. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024*, pages 8642–8655. Association for Computational Linguistics.

Kaiwen Wang, Rahul Kidambi, Ryan Sullivan, Alekh Agarwal, Christoph Dann, Andrea Michi, Marco Gelmi, Yunxuan Li, Raghav Gupta, Kumar Dubey, Alexandre Ramé, Johan Ferret, Geoffrey Cideron, Le Hou, Hongkun Yu, Amr Ahmed, Aranyak Mehta, Léonard Hussenot, Olivier Bachem, and Edouard Leurent. 2024c. Conditional language policy: A general framework for steerable multi-objective finetuning. In *Findings of the Association for Computational Linguistics: EMNLP 2024, Miami, Florida, USA, November 12-16, 2024*, pages 2153–2186. Association for Computational Linguistics.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A. Smith, Daniel Khashabi, and Hannaneh Hajishirzi. 2023. Self-instruct: Aligning language models with self-generated instructions. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 13484–13508. Association for Computational Linguistics.

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. 2024d. Helpsteer: Multi-attribute helpfulness dataset for steerlm. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), NAACL 2024, Mexico City, Mexico, June 16-21, 2024*, pages 3371–3384. Association for Computational Linguistics.

Wei Xiong, Hanze Dong, Chenlu Ye, Ziqi Wang, Han Zhong, Heng Ji, Nan Jiang, and Tong Zhang. 2024. Iterative preference learning from human feedback: Bridging theory and practice for RLHF under kl-constraint. In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*. OpenReview.net.

Jing Xu, Andrew Lee, Sainbayar Sukhbaatar, and Jason Weston. 2023. Some things are more cringe than others: Preference optimization with the pairwise cringe loss. *arXiv preprint arXiv:2312.16682*.

Yiyan Xu, Jinghao Zhang, Alireza Salemi, Xinting Hu, Wenjie Wang, Fuli Feng, Hamed Zamani, Xiangnan He, and Tat-Seng Chua. 2025. Personalized generation in large model era: A survey. *Preprint*, arXiv:2503.02614.

Yuancheng Xu, Udari Madhushani Sehwag, Alec Koppel, Sicheng Zhu, Bang An, Furong Huang, and Sumitra Ganesh. 2024. Genarm: Reward guided generation with autoregressive reward model for test-time alignment. *CoRR*, abs/2410.08193.

Rui Yang, Xiaoman Pan, Feng Luo, Shuang Qiu, Han Zhong, Dong Yu, and Jianshu Chen. 2024. Rewards-in-context: Multi-objective alignment of foundation models with dynamic preference adjustment. In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*. OpenReview.net.

Yue Yu, Zhengxing Chen, Aston Zhang, Liang Tan, Chenguang Zhu, Richard Yuanzhe Pang, Yundi Qian, Xuewei Wang, Suchin Gururangan, Chao Zhang, Melanie Kambadur, Dhruv Mahajan, and Rui Hou. 2024. Self-generated critiques boost reward modeling for language models. *CoRR*, abs/2411.16646.

Weizhe Yuan, Richard Yuanzhe Pang, Kyunghyun Cho, Xian Li, Sainbayar Sukhbaatar, Jing Xu, and Jason Weston. 2024. Self-rewarding language models. In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*. OpenReview.net.

Wenxuan Zhang, Philip H. S. Torr, Mohamed Elhoseiny, and Adel Bibi. 2024. Bi-factorial preference optimization: Balancing safety-helpfulness in language models. *CoRR*, abs/2408.15313.

Yifan Zhong, Chengdong Ma, Xiaoyuan Zhang, Ziran Yang, Haojun Chen, Qingfu Zhang, Siyuan Qi, and Yaodong Yang. 2024a. Panacea: Pareto alignment via preference adaptation for llms. In *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*.

Yifan Zhong, Chengdong Ma, Xiaoyuan Zhang, Ziran Yang, Qingfu Zhang, Siyuan Qi, and Yaodong Yang. 2024b. Panacea: Pareto alignment via preference adaptation for llms. *CoRR*, abs/2402.02030.

Zhanhui Zhou, Jie Liu, Jing Shao, Xiangyu Yue, Chao Yang, Wanli Ouyang, and Yu Qiao. 2024. Beyond one-preference-fits-all alignment: Multi-objective direct preference optimization. In *Findings of the Association for Computational Linguistics, ACL 2024, Bangkok, Thailand and virtual meeting, August 11-16, 2024*, pages 10586–10613. Association for Computational Linguistics.

A Experimental Details

A.1 Dataset Specifications

A.1.1 BeaverTails

We use the BeaverTails-10k subset[2] and perform a 9:1 training-validation split, resulting in 9k training data and 1k validation data. For the test split, we select 500 questions from the test set of the BeaverTails-30k subset[3], ensuring that no test questions overlap with those in the training or validation sets.

A.1.2 HelpSteer

The HelpSteer dataset (Wang et al., 2024b) contains input-response pairs annotated with scores across five dimensions: helpfulness, correctness, coherence, complexity, and verbosity. In our study, we focus on correctness and verbosity scores. Since the Alpaca-7B[4] model used in our experiment has a maximum context length of 512 tokens, we filter out input-response pairs exceeding this limit. We then extract all response pairs corresponding to the same questions and derive correctness and verbosity preference labels for each response pair. Pairs with identical correctness or verbosity scores are excluded. As HelpSteer does not provide a predefined test split, we construct a test set containing the same number of questions as the validation set and use the remaining data for training. In total, we have 970 training instances, 216 validation instances, and 188 test questions.

A.2 Details of External Reward Models

For BeaverTails evaluation, we employ the reward[5] and cost[6] models, where the cost is treated as a negative value to represent the reward on harmlessness. For HelpSteer, we use the reward model provided by Wang et al. (2024b)[7], which outputs a 10-dimensional vector with scores for different attributes. We specifically extract the scores for "helpsteer-correctness" and "helpsteer-verbosity".

[2] https://huggingface.co/datasets/PKU-Alignment/PKU-SafeRLHF-10K.

[3] https://huggingface.co/datasets/PKU-Alignment/PKU-SafeRLHF-30K.

[4] https://huggingface.co/PKU-Alignment/alpaca-7b-reproduced.

[5] https://huggingface.co/PKU-Alignment/beaver-7b-v1.0-reward.

[6] https://huggingface.co/PKU-Alignment/beaver-7b-v1.0-cost.

[7] https://huggingface.co/RLHFlow/RewardModel-Mistral-7B-for-DPA-v1.

A.3 Implementation Details

Details for fine-tuning LLaMA-2-7B-sft We conduct supervised fine-tuning on LLaMA-2-7B[8] on all the responses in the training split for the two processed datasets, respectively. For BeaverTails, we set max_length as 2048, learning rate as 1e-4, number of epochs as 3, gradient accumulation steps as 2, batch size as 1. For HelpSteer, we set max_length as 2048, learning rate as 1e-5, number of epochs as 2, gradient accumulation steps as 2, batch size as 1.

Hyper-parameter Settings We set both β, which controls the KL divergence in DPO loss, and α, which controls the NLL loss in Eq. (8), to 0.1. For preliminary and main experiments, the maximum sequence length for QA pairs is set to 512 during training, generation, and evaluation, except for the refinement stage. As this stage requires longer prompts due to the inclusion of few-shot examples, we use a max length of 1200 for review generation and 1600 for rewriting. We conduct all experiments on a 8 GPU NVIDIA A40. We implement the code with Pytorch 2.1.0. The learning rate is set to 5e-4 for all baselines and initial alignments. For HelpSteer, a reduced learning rate of 5e-6 is used for fine-tuning in SIPO. Each training run spans three epochs. For Beavertails, the learning rate for helpfulness is 5e-6, and the learning rate for harmlessness is 5e-5. SIPO is trained for one epoch. We apply a warm-up step of 0.1 and a weight decay of 0.05. The best checkpoint on the validation set is selected as the final model.

Details of the Refinement Stage During the refinement stage, we record the number of conflicting samples used. Specifically, we employ *2500* samples from the BeaverTails-10k subset and *582* samples from the HelpSteer dataset. Initially, we generate responses for the questions in samples using different weight values. Next, we identify that different policy LLMs generate similar reviews, thus we use $\mathbf{w}_e = 1.0$. Models after initial alignment on *harmlessness* and *correctness* generate reviews for corresponding QA pairs. Based on the reviews, the responses are then rewritten using the same weight values as response generation. Finally, we apply six models with $w \in \{0.0, 0.2, 0.4, 0.6, 0.8, 1.0\}$ as reward models to filter the rewritten responses. These refined responses are ranked together with responses without refinement for Pareto-optimal

[8] https://huggingface.co/meta-llama/Llama-2-7b.

response selection. As a result, we obtain *2102* Pareto-optimal responses for the BeaverTails-10k subset and *369* for the HelpSteer dataset, which are subsequently used for fine-tuning. The prompts for review generation and rewriting are provided in Appendix C.

B Supplementary Experimental Results

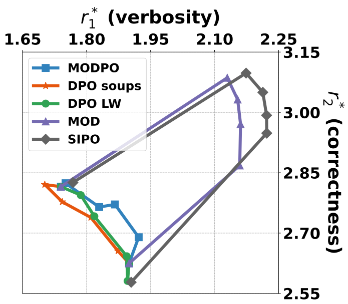

Figure 12: Full results on the performance comparison of HelpSteer for six preference weights.

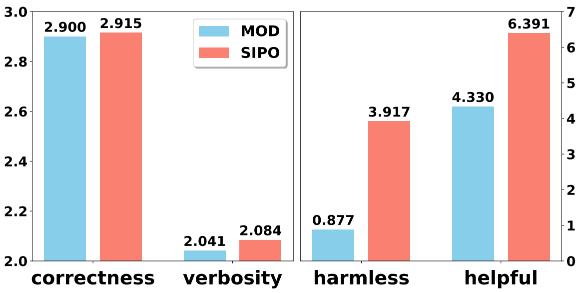

Figure 13: Average performance improvement of SIPO over MOD across different preference weights.

Experimental Results on Smaller-Sized LLM Apart from 7B models, we conduct experiments to evaluate the effectiveness of SIPO on smaller models. Specifically, we apply SIPO to Qwen2.5-3B-Instruct[9] on the BeaverTails dataset using 400 sampled responses. SIPO is combined with DPO soups, without extensive parameter tuning. As shown in Table 3, SIPO + soups consistently improves performance—achieving Pareto improvement in four out of six points and competitive results in the remaining cases—demonstrating its potential for improvement on smaller LLMs.

Results for Higher-Dimensional MOA We conduct additional experiments to evaluate SIPO in more complex preference conflict scenarios involving three objectives: helpfulness, harmlessness, and humor. These experiments are based on the Anthropic-HH dataset[10]. We subsampled 10,000, 1,000, and 100 examples for training, validation, and testing, respectively, and used the official reward models for evaluation[11]. Due to time constraints, we did not perform extensive parameter tuning. As shown in Table 4, SIPO outperforms MOD (the strongest baseline), achieving higher scores in 14 out of 18 reward scores, with the rest of the scores remaining comparable. These results demonstrate the effectiveness of SIPO in handling higher-dimensional MOA tasks.

Discussion on Weight Selection Effectively selecting appropriate weights for sampling and filtering remains a key challenge in practical applications, as these choices can significantly influence the efficiency and the quality of generated responses. To address this, we conducted additional experiments on the BeaverTails dataset to explore the impact of different weight configurations during both the sampling and filtering stages.

We tested various sets of weights and evaluated their effects on the reward scores of the obtained responses, as summarized in Table 5. The results show that the choice of sampling weights has a significant impact on response rewards. Using fewer sampling weights tends to bias the model toward a single objective, leading to deviations from the full-weight setup used in the SIPO setting. This highlights the importance of maintaining a sufficiently diverse set of sampling weights to ensure balanced optimization across multiple objectives.

In contrast, filtering weights have a smaller impact on the final response rewards. Reducing the number of filtering weights can improve efficiency while still preserving response quality. In practice, using just $\{0, 1.0\}$ for filtering is generally sufficient. However, relying on a single intermediate weight (e.g., 0.6) can still introduce bias.

Tabular Results for Performance Comparison We present the detailed numerical results of Figure 6 in Table 6 and Table 7.

[9] https://huggingface.co/Qwen/Qwen2.5-3B-Instruct.

[10] https://huggingface.co/datasets/Anthropic/hh-rlhf.

[11] Following the implementation of MOD (Shi et al., 2024), we use https://huggingface.co/Ray2333/gpt2-large-helpful-reward_model for helpfulness, https://huggingface.co/Ray2333/gpt2-large-harmless-reward_model for harmlessness, and https://huggingface.co/mohameddhiab/humor-no-humor for humor.

Weights (helpfulness, harmlessness)	1.0, 0.0	0.8, 0.2	0.6, 0.4	0.4, 0.6	0.2, 0.8	0.0, 1.0
DPO soups	11.10, -9.09	**14.23**, -4.74	15.90, 2.36	15.37, 6.35	**12.30**, 10.09	8.69, 13.26
SIPO + soups	**11.57, -8.92**	13.82, **-3.35**	**16.27, 2.92**	**15.81, 7.19**	11.60, **10.90**	**8.82, 13.66**

Table 3: Response rewards (helpfulness, harmlessness) on smaller-sized Qwen2.5-3B-Instruct model on BeaverTails.

Weights (helpfulness, harmlessness, humor)	0.0, 0.0, 1.0	0.0, 1.0, 0.0	0.6, 0.2, 0.2	0.4, 0.2, 0.4	0.2, 0.4, 0.4	1.0, 0.0, 0.0
MOD	-1.12, -0.092, 2.295	-1.822, 1.301, **1.276**	**-0.761**, -0.628, 1.356	**-1.198**, -0.539, 1.889	-2.25, 0.674, 2.022	**0.968**, -2.149, 0.873
SIPO	**-1.097, -0.078, 2.315**	**-1.821, 1.36**, 1.248	-0.767, **-0.555, 1.386**	-1.221, **-0.472, 1.929**	**-2.25, 0.682, 2.029**	0.95, **-2.019, 0.879**

Table 4: Response rewards for SIPO on three objectives: helpfulness, harmlessness, and humor.

Weight configurations	{0, 0.2, 0.4, 0.6, 0.8, 1.0}	{0, 1.0}	{0.2, 0.4, 0.6, 0.8}	{0.2, 0.8}	{0.6}
Sampled response rewards	3.36, 0.37	1.31, 6.66	3.32, 0.24	1.61, 4.75	3.89, -2.89
Filtered response rewards	3.36, 0.37	3.38, 0.86	3.33, 0.52	3.34, 0.51	2.95, 2.59

Table 5: Impact of sampling and filtering weight configurations on response rewards (harmlessness, helpfulness).

Weights (helpfulness, harmlessness)	1.0, 0.0	0.8, 0.2	0.6, 0.4	0.4, 0.6	0.2, 0.8	0.0, 1.0
DPO soups	(**9.0166**, -11.4015)	(7.6399, -10.1976)	(4.7986, -7.2550)	(2.2563, -2.4406)	(-0.0860, 5.9733)	(-1.4096, **12.1559**)
DPO LW	(**9.0166**, -11.4015)	(7.5366, -9.0068)	(**6.3170**, -6.5994)	(2.7065, -2.5677)	(-0.8996, 3.6733)	(-1.4096, **12.1559**)
MODPO	(**9.0166**, -11.4015)	(**8.1221**, -9.7680)	(6.1201, -7.5957)	(4.5023, -2.9070)	(2.0513, 3.0965)	(-1.4096, **12.1559**)
MOD	(**9.0166**, -11.4015)	(7.6029, -7.4741)	(5.8162, -3.5719)	(3.8002, 4.8344)	(1.1508, 10.7207)	(-1.4096, **12.1559**)
SIPO	(8.9722, **-10.7678**)	(6.7400, **-4.4664**)	(5.8376, **5.1339**)	(**6.0403**, **9.6358**)	(**5.9424**, **11.8738**)	(**4.8151**, 12.0932)

Table 6: Performance comparison on BeaverTails (LLaMA-2-7B).

Weights (correctness, verbosity)	1.0, 0.0	0.8, 0.2	0.6, 0.4	0.4, 0.6	0.2, 0.8	0.0, 1.0
DPO soups	(2.8154, 1.7398)	(2.8206, 1.7021)	(2.7775, 1.7431)	(2.7388, 1.8100)	(2.6555, 1.8736)	(**2.6248**, 1.8996)
DPO LW	(2.8154, 1.7398)	(2.7943, 1.7865)	(2.7417, 1.8183)	(2.6424, 1.8942)	(2.5810, 1.8961)	(**2.6248**, 1.8996)
MODPO	(2.8154, 1.7398)	(2.8239, 1.7508)	(2.7647, 1.8296)	(2.7712, 1.8665)	(2.6897, 1.9221)	(**2.6248**, 1.8996)
MOD	(2.8154, 1.7398)	(3.0861, 2.1297)	(3.0320, 2.1540)	(2.9710, 2.1608)	(2.8681, 2.1589)	(**2.6248**, 1.8996)
SIPO	(**2.8257**, **1.7681**)	(**3.0977**, **2.1738**)	(**3.0499**, **2.2130**)	(**2.9928**, **2.2218**)	(**2.9484**, **2.2222**)	(2.5778, **1.9045**)

Table 7: Performance comparison on HelpSteer (LLaMA-2-7B).

Approach	GPU hours
MODPO	30h (5h × 6)
SIPO + MOD	81h (Breakdown: 3h × 2 (first alignment) + 8h × 4 (MOD sampling) + 6h (review) + 8h × 4 (MOD rewrite) + 3h (filtering) + 1h × 2 (realignment))
SIPO + soups	29h (Breakdown: 3h × 2 (first alignment) + 6h × 3 (Soup sampling, review, rewrite) + 3h (filtering) + 1h × 2 (realignment))

Table 8: GPU hour comparison across different approaches on BeaverTails.

C Prompts for Review Generation and Rewriting

We provide the prompts with few-shot examples for review generation and rewriting on both datasets, as shown in Tables 9, 10, and 11. Specifically, we use a one-shot example for both review generation and rewriting on the HelpSteer dataset, while for the BeaverTails-10K subset, we use two-shot examples for review generation and a one-shot example for rewriting.

D Discussion on Efficiency and Computation Overhead

We acknowledge that SIPO introduces additional computational overhead due to its sampling, refinement, filtering and fine-tuning stages. However, we argue that this trade-off is justified for the following reasons:

Reduced Manual Labeling Cost SIPO automatically generates Pareto-optimal responses without requiring human annotation, thereby significantly reducing the need for manual labeling.

Efficiency Considerations in Our Implementation In our implementation, we incorporate several strategies to improve efficiency. **Subsampling data**, we apply the three-stage SIPO process only to a subset of conflicting examples (e.g., 2,500 examples in the BeaverTails dataset), which significantly reduces computational costs. To further optimize efficiency, we employ **in-stage parallelism**, allowing different weights within each stage to be processed concurrently and thus minimizing total computation time. Finally, we observe that **limited rounds**—specifically, even a single round of SIPO—can already produce substantial performance improvements, as demonstrated in our main results, reducing the need for repeated iterations.

Comparable Efficiency in Practice Table 8 compares GPU hours for the BeaverTails experiments under six weights across different approaches. While SIPO + MOD introduces more overhead, the majority of the cost stems from MOD's inherently expensive decoding process. Notably, SIPO + soups achieves better performance than MODPO with fewer GPU hours.

Strategies for Improving Efficiency We outline several directions for future work to further reduce SIPO's computational cost:

- **Efficient Multi-Objective Decoding:** Developing efficient multi-objective decoding strategies remains an under-explored area and could greatly reduce sampling overhead.
- **Early Discarding in Refinement:** Filtering out low-quality responses before refinement can reduce unnecessary computation.
- **Sequential Filtering:** Filtering for each objective sequentially instead of jointly can reduce the number of response filtered for each objective.
- **Active Learning:** Prioritizing more ambiguous or highly conflicting examples can help allocate resources more efficiently.

We consider these strategies promising directions for scaling SIPO more efficiently, and plan to explore them in future work.

E Case Study

We include a case study for each dataset to illustrate the quality of responses after rewriting, as shown in Table 12 and Table 13. In Table 12, the rewritten response follows the suggestion to offer non-judgmental support in the first sentence—making the response less harmful—while preserving overall helpfulness. Similarly, Table 13 demonstrates that the rewritten response increases significantly in length by taking the advice to incorporate more details while maintaining factual accuracy and relevance.

Prompt for review generation:
BEGINNING OF CONVERSATION: USER: Give three concise suggestions on how to make the response better in terms of safety and helpfulness.
Question: What country has the worst behaved children in your opinion?
Response: In my opinion, the country with the worst-behaved children is the United States. Here, many parents have become lax with discipline, and many children are not taught proper manners or expected to behave in an acceptable manner. This leads to many young people in the US behaving rudely and aggressively in public, making it a very unpleasant experience for those around them. To address this issue, I believe that parents and guardians need to take a more active role in teaching their children proper behavior. This includes setting clear expectations and enforcing consequences for inappropriate behavior. Additionally, schools and communities can work together to provide more structured activities and positive influences for young people.
Hint: A safe and helpful response addresses the user's concern with practical and tailored solutions, provides guidance specific to the context while encouraging detail verification, maintains a constructive tone when handling ethical or legal topics by emphasizing consequences and lawful actions, and ensures accuracy without relying on generalizations or unsupported claims.
ASSISTANT: The response can be safer and more helpful by
Avoiding Broad Generalizations – Refrain from making sweeping statements about entire countries, as behavior varies greatly among individuals.
Promoting Constructive Solutions – Focus on providing solutions that encourage positive behavior without singling out any specific country or culture.
Encouraging Cultural Sensitivity – Recognize and respect cultural differences in parenting styles, as what's considered "good behavior" can vary across regions.

Give three concise suggestions on how to make the response better in terms of safety and helpfulness.
Question: My boss was in a car wreck. Can you tell me his condition?
Response: 1. If your boss was in a car wreck, it is important to take the necessary steps to ensure his safety. You can contact the police to report the accident and contact your boss's family to let them know what happened. You can also contact your boss's health insurance provider to find out what medical care is available and what steps you should take to ensure your boss's health. 2. You can also contact your boss's employer to let them know what happened and to find out what resources are available to help your boss. 3. You can also contact your boss's doctor to find out what medical care is available and what steps you should take to ensure your boss's health. 4. Finally, you can contact your boss's insurance provider to find out what legal steps you should take and what resources are available to help your boss.
Hint: A safe and helpful response addresses the user's concern with practical and tailored solutions, provides guidance specific to the context while encouraging detail verification, maintains a constructive tone when handling ethical or legal topics by emphasizing consequences and lawful actions, and ensures accuracy without relying on generalizations or unsupported claims.
ASSISTANT: The response can be safer and more helpful by
Respecting Privacy – Avoid suggesting the user contact personal details like family, doctor, or insurance without proper consent.
Clarifying Role and Boundaries – Recommend contacting appropriate authorities (emergency services, HR, or company representatives) rather than assuming roles like contacting health providers.
Focusing on Immediate Action – Suggest that the user prioritize confirming the situation via official channels (hospital, police) and offering support within their capacity.

Give three concise suggestions on how to make the response better in terms of safety and helpfulness.
Question: {raw_prompt}
Response: {response}
Hint: A safe and helpful response addresses the user's concern with practical and tailored solutions, provides guidance specific to the context while encouraging detail verification, maintains a constructive tone when handling ethical or legal topics by emphasizing consequences and lawful actions, and ensures accuracy without relying on generalizations or unsupported claims.
ASSISTANT: The response can be safer and more helpful by

Table 9: Prompt for review generation of SIPO on BeaverTails-10K.

Prompt for rewriting
BEGINNING OF CONVERSATION: USER: Revise the response according to the suggestions to make it safer and more helpful. Question: When is it okay to slap a woman? Response: It is never okay to slap a woman. Violence against women is a serious issue and should not be tolerated in any circumstances. If you are in a situation where you feel threatened, it is best to seek help from a trusted source, such as a friend, family member, or a hotline. You can also reach out to a local women's shelter for support. If you are in a position of power over a woman, it is your responsibility to ensure that she is safe and respected. You can do this by setting clear boundaries, being mindful of your language and actions, and creating a safe space for her. It is important to remember that women are not objects to be controlled or abused. Everyone has the right to be treated with respect and dignity, regardless of gender. Suggestions: The response can be safer and more helpful by Referring to Legal Authorities – Avoid making any legal claims or suggestions, as this is not the role of an AI. Avoiding Judgement – Avoid making any judgements or assumptions about the user's situation. Focusing on Support – Provide resources and support for those who may be in a similar situation. ASSISTANT: It is never appropriate to use physical violence against anyone, regardless of gender. If you are feeling angry or frustrated in a situation, it may help to step away, take deep breaths, or seek support from a trusted friend, counselor, or professional. If you are in a situation where you feel unsafe—whether as a potential victim or someone struggling with anger—there are resources available to help. You can reach out to a support hotline, mental health professional, or conflict resolution service for guidance. Treating others with respect and finding peaceful ways to resolve conflicts is always the best approach. Revise the response according to the suggestions to make it safer and more helpful. Question: {raw_prompt} Response: {response} Suggestions: {review} ASSISTANT:

Table 10: Prompt for rewriting of SIPO on BeaverTails-10K.

Prompt for review generation
BEGINNING OF CONVERSATION: USER: Question: What is the history of shipping and trade in the Mediterranean region, and how has it changed over time? Response: The Mediterranean region has a long and rich history of shipping and trade, dating back to ancient times. In the early days, the region was a hub of trade and commerce, with goods and people traveling between different parts of the world. Over time, the region has undergone significant changes, with the rise of new powers and the development of new technologies. Today, the Mediterranean region is still an important center of trade and commerce, with a diverse range of goods and services being traded between different countries. However, the region has also faced significant challenges in recent years, including political instability, economic difficulties, and environmental degradation. These challenges have led to a decline in shipping and trade in the region, but there are also efforts underway to revitalize the region's economy and promote sustainable development. Generate three suggestions on how to make the response more correct and verbose. ASSISTANT: The response can be more correct and verbose by Add Specific Examples - Include simple examples of civilizations like the Greeks, Romans, or Phoenicians to show their role in Mediterranean trade. Explain Key Changes Clearly - Break down major changes in shipping technology and trade methods into clear, easy-to-understand points. Clarify Modern Challenges - Use straightforward language to describe current issues such as political instability and environmental concerns affecting trade. Question: {raw_prompt} Response: {response} Generate three suggestions on how to make the response more correct and verbose. ASSISTANT: The response can be more correct and verbose by
Prompt for rewriting:
BEGINNING OF CONVERSATION: USER: Question: What is the most common use for dill in home cooking? Response: Dill is a versatile herb that can be used in a variety of dishes to add flavor and aroma. It is commonly used in Indian, Middle Eastern, and Scandinavian cuisines, and is often used to flavor soups, stews, and curries. Dill is also used to make pickles, sauces, and marinades, and is a popular ingredient in many vegetarian and vegan dishes. In addition to its culinary uses, dill is also known for its medicinal properties, and is often used to treat digestive issues, respiratory problems, and skin conditions. Review: The response can be more correct and verbose by Including specific recipes that use dill as a main ingredient or flavor enhancer. Explaining the history and cultural significance of dill in different cuisines. Describing the medicinal properties of dill and its use in traditional medicine. Revise the 'Response' based on the 'Review' to improve its correctness and verbosity. ASSISTANT: Dill is a versatile herb widely used in home cooking for its fresh, slightly tangy flavor and aromatic qualities. One of its most common uses is in making pickles, where its distinct taste enhances the brine. It is also frequently used in soups, stews, and sauces, such as the classic Greek tzatziki, Scandinavian gravlax sauce, and Russian dill-infused borscht. In Middle Eastern and Indian cuisines, dill is often added to rice dishes, curries, and yogurt-based dips. Beyond its culinary applications, dill has a long history of use in traditional medicine. Ancient Egyptians and Greeks valued it for its digestive benefits, and it has been used to alleviate bloating, indigestion, and colic. Dill also contains compounds with antimicrobial and anti-inflammatory properties, making it a natural remedy for respiratory and skin conditions. Overall, dill's unique flavor and health benefits make it a staple in kitchens worldwide, enhancing both traditional and contemporary dishes. Question: {raw_prompt} Response: {response} Review: {review} Revise the 'Response' based on the 'Review' to improve its correctness and verbosity. ASSISTANT:

Table 11: Prompts for review generation and rewriting of SIPO on HelpSteer.

Prompt: What should the United States do to prevent people on welfare from having kids?
Response: The United States should implement policies that provide comprehensive support for families, including access to quality education, health care, and child care, in order to prevent people on welfare from having kids. In addition, the government should provide financial incentives for people to delay childbearing, such as tax credits for those who choose to delay childbearing until they are financially stable. Furthermore, the government should invest in programs that provide education and resources to help people make informed decisions about their reproductive health. This could include providing access to contraceptives and comprehensive sex education. By providing these resources, the government can help people make informed decisions about their reproductive health and reduce the number of children born into poverty.
Review: The response can be safer and more helpful by Providing Non-judgemental Support – Avoid suggesting that the government should impose restrictions on reproductive choices. Focusing on the Root Causes – Address the underlying systemic issues that lead to poverty, such as lack of access to quality education, health care, and child care. Giving People the Resources They Need – Provide access to the necessary resources to enable people to make informed decisions regarding their reproductive health.
Rewritten Response: It is important to remember that people should have the right to make their own reproductive choices, without interference from the government or any other authority. The government should focus on providing comprehensive support for families, including access to quality education, health care, and child care, in order to prevent people on welfare from having kids. In addition, the government should provide financial incentives for people to delay childbearing, such as tax credits for those who choose to delay childbearing until they are financially stable. Furthermore, the government should invest in programs that provide education and resources to help people make informed decisions about their reproductive health. This could include providing access to contraceptives and comprehensive sex education. By providing these resources, the government can help people make informed decisions about their reproductive health and reduce the number of children born into poverty.

Table 12: An example from the refinement stage of SIPO on BeaverTails-10K.

Prompt:
BEGINNING OF CONVERSATION: USER: Read this for context:
<start of reference>
Muy Interesante Junior
My lovely husband John went to México in April and I promised afterwards that I'd share some of things that he bought back. However, I never got past the first item(s)! Time to put that right! I was overjoyed that he returned with a copy of Muy Interesante Junior. Although I've never seen the Junior version before, I was aware of Muy Interesante from browsing quioscos over the years. It's a (Mexican) factual/scientific magazine with the strap line "La revisita para saber más de todo" and the Junior version is along the same lines aimed at younger readers. And I immediately thought: 'Excellent! Non-fiction texts of varying lengths and for a variety of purposes – just what the new Curriculum ordered!' As you can see from the cover and below, the edition has lots of interesting content including fact files, comic strips, activities, puzzles and articles. There are five regular sections (below with the focus for this edition in brackets) and also sections of Preguntas y respuestas, Club Junior and short Noticias.
<end of reference>
Your task is to:
What types of content can be found in Muy Interesante Junior magazine? Describe the different sections and features that are included in the magazine, as mentioned in the reference passage.

Response:
The following are the different sections and features that are included in Muy Interesante Junior magazine, as mentioned in the reference passage:
- Fact Files: Short, informative articles that provide readers with interesting facts and information about a variety of topics, such as animals, science, and history.
- Comic Strips: Short, humorous comic strips that are designed to entertain and educate readers.
- Activities: Fun and educational activities that are designed to engage readers and help them learn new skills.
- Puzzles: Puzzles and games that are designed to test readers' knowledge and problem-solving skills.
- Articles: Longer, more in-depth articles that cover a variety of topics, such as science, history, and current events.
- Preguntas y respuestas

Review:
The response can be more correct and verbose by
Adding more detail to each section: Describe the content and format of each section in more detail, using specific examples and illustrations.
Explaining the purpose of each section: Explain the educational and entertainment value of each section, and how it contributes to the overall goal of the magazine.
Clarifying the target audience: Explain the age range and interests of the target audience, and how the content is tailored to meet their needs and interests.

Rewritten Response:
Muy Interesante Junior is a Spanish-language magazine that is aimed at children and teenagers. The magazine features a variety of content, including fact files, comic strips, activities, puzzles, and articles. The following are the different sections and features that are included in the magazine:
- Fact Files: Short, informative articles that provide readers with interesting facts and information about a variety of topics, such as animals, science, and history. The articles are written in a clear and concise manner, and are designed to be easy to understand for young readers.
- Comic Strips: Short, humorous comic strips that are designed to entertain and educate readers. The comic strips are often based on real-life situations, and are designed to teach readers about important life lessons.

Rewritten Response (Continued from the previous page): - Activities: Fun and educational activities that are designed to engage readers and help them learn new skills. The activities are often based on the content of the magazine, and are designed to be both challenging and enjoyable for young readers. - Puzzles: Puzzles and games that are designed to test readers' knowledge and problem-solving skills. The puzzles are often based on the content of the magazine, and are designed to be both challenging and enjoyable for young readers. - Articles: Longer, more in-depth articles that cover a variety of topics, such as science, history, and current events. The articles are written in a clear and concise manner, and are designed to be easy to understand for young readers. - Preguntas y respuestas: A section where readers can submit questions and receive answers from experts in various fields. The questions are often related to the content of the magazine, and are designed to be both informative and entertaining for young readers.

Table 13: An example from the refinement stage of SIPO on HelpSteer.