Modulated Intervention Preference Optimization (MIPO): Keep the Easy, Refine the Difficult

Anonymous ACL submission

Abstract

Preference optimization methods typically begin training with a well-trained SFT model as a reference model. In RLHF and DPO, a regularization term is used during the preference optimization process to prevent the policy model from deviating too far from the reference model's distribution, thereby avoiding the generation of anomalous responses. However, if the reference model is not aligned with the given data and requires significant deviation from its current state, a regularization term may actually hinder the model alignment. In this study, we propose **Modulated Intervention Preference Optimization (MIPO)** to address this issue. MIPO modulates the degree of intervention from the reference model based on how well the given data is aligned with it. If the data is well-aligned, the intervention is increased to prevent the policy model from diverging significantly from reference model. Conversely, if the alignment is poor, the interference is reduced to facilitate more extensive training. We compare MIPO and DPO using Mistral-7B and Llama3-8B on Alpaca Eval 2.0 and MT-Bench, showing that MIPO consistently outperforms DPO across various evaluation scenarios

1 Introduction

As the performance of Large Language Models (LLMs) trained with a large amount of data has been attracting attention, methods (Chowdhery et al., 2023; Touvron et al., 2023; Brown et al., 2020) for training them have been actively studied. The commonly used LLM training pipeline is to pretrain LLM using a large amount of data, and then use the instruction-tuning method (Wei et al., 2021) to allow LLM to follow the human-provided instruction.

However, it is difficult to train LLM to produce the desired output (helpful, harmless) or to prevent LLM from producing the output that LLM should not produce (Bai et al., 2022a). Therefore, LLM alignment methods employing human feedback have started to gain significant attention.

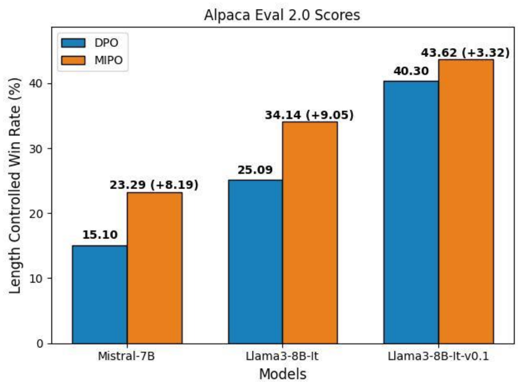


Figure 1: Alpacaeval 2.0 scores for MIPO and DPO implementations on Mistral-7B-Base and Llama-8B-Instruct. v0.1 is a model trained on different dataset.

Among these methods, Reinforcement Learning from Human Feedback (RLHF) (Christiano et al., 2017; Askell et al., 2021) received significant attention. Models trained with RLHF are well-aligned with human feedback, demonstrating reliable performance as a result (Korbak et al., 2023; Havrilla et al., 2024; Dai et al., 2023). However, the RLHF approach involves a complex training process, including the training of a reward model, which has posed significant challenges in the implementation and training (Casper et al., 2023; Peng et al., 2023).

Direct Preference Optimization (DPO) (Rafailov et al., 2024) is a method designed to overcome these limitations. In DPO, the optimization problem of RLHF is modified to eliminate the reward model and train only the policy model. This makes it easier to train DPO compared to RLHF, and DPO also effectively learned human preferences, demonstrating strong performance.

In DPO and RLHF, the policy model is trained to align with the instance while ensuring its distribution does not move significantly away from the reference model's distribution to prevent it from generating anomalous responses (ex. hallucinations). Therefore, if the reference model is moderately aligned with the given preference pair, it could be possible to train a well-aligned policy model for the given data without significantly diverging from the reference model's distribution. However, if the reference model is not aligned with the given preference pair, it will be difficult for the policy model to align with the data through minor adjustments, without moving far from the reference model's distribution. Therefore, it is crucial to adjust the training objective based on how well the reference model is aligned.

In this paper, we propose a preference optimization algorithm, **Modulated Intervention Preference Optimization** (MIPO), to address this issue. As seen in Figure 2, MIPO utilizes the average log likelihood to measure how well the reference model is aligned with the given preference pair. Through this value, the MIPO objective is configured to modulate the intervention of the reference model, allowing more extensive training on pairs that are judged to be poorly aligned with the reference model. We use Alpaca Eval 2.0 and MT-Bench to compare the performance of MIPO with DPO and other preference optimization methods. Across diverse experimental settings, MIPO consistently achieves outstanding performance. To summarize, **MIPO has the following properties:**

- **Modulate the intervention of the Reference Model**: MIPO is a novel approach that modulates the intervention of the reference model for each instance. It determines the extent of the reference model's intervention based on the degree of alignment. MIPO maintains performance on pairs where the reference model already well-aligned, while simultaneously achieving substantial performance gains on pairs where the reference model previously underperformed (Section $4).
- **Outstanding Benchmark Performance**: We conduct experiments using Llama3-8B-Instruct (AI@Meta, 2024) and Mistral-7B-Base (Jiang et al., 2023a) to verify the effectiveness of MIPO in various models. On Alpaca Eval 2.0, our proposed method consistently outperforms DPO. As we can see in Figure 1, in Llama3-8B-Instruct, it outperforms DPO by approximately 9 points (+36.07%), and in Mistral-7B-Base, it outperforms about 8 points (+54.24%). In most cases, MIPO achieves the best performance not only compared to DPO but also when compared to other methods. On MT-Bench, MIPO also exhibits the best performance among the compared approaches (Section $6.1).
- **Simple and Effective Training**: The high-performance model can be found in MIPO by tuning only the hyper-parameter β. Moreover, consistently outstanding performance is achieved within a specific range of β, independent of model architecture or dataset. Thus, unlike other methods that require extensive tuning, this approach allows for easy acquisition of a high-performance model with minimal tuning effort (Section $6.2).

2 Related Works

After being pretrained on a large amount of data (Chowdhery et al., 2023) and fine-tuned (Chung et al., 2024; Ramamurthy et al., 2022), LLMs have achieved notable performance across many tasks (Touvron et al., 2023; Brown et al., 2020; Thoppilan et al., 2022). However, LLMs that could generate responses that were even more helpful and harmless were needed, leading to the development of preference optimization methods (Christiano et al., 2017; Bai et al., 2022a,b) that fine-tune LLMs more closely to human feedback.

RLHF (Askell et al., 2021; Ouyang et al., 2022) is one such preference optimization method for LLM alignment. In RLHF, preference data is used to train a reward model, which is then utilized to optimize the policy model by Proximal Policy Optimization (Schulman et al., 2017). RLHF effectively aligns models with human feedback, resulting in good performance (Korbak et al., 2023; Havrilla et al., 2024). However, there are challenges, such as the difficulty of obtaining scored data, ensuring stable training, and the necessity of training a reward model (Casper et al., 2023; Peng et al., 2023; Wang et al., 2024).

DPO is a preference optimization method that solves optimization problem of RLHF in a more easier and efficient manner. (Rafailov et al., 2024) proposed DPO to eliminate the reward model in RLHF and train only the policy model with preference data. It is simple compared to RLHF, and the training phase is more stable. So it has become one

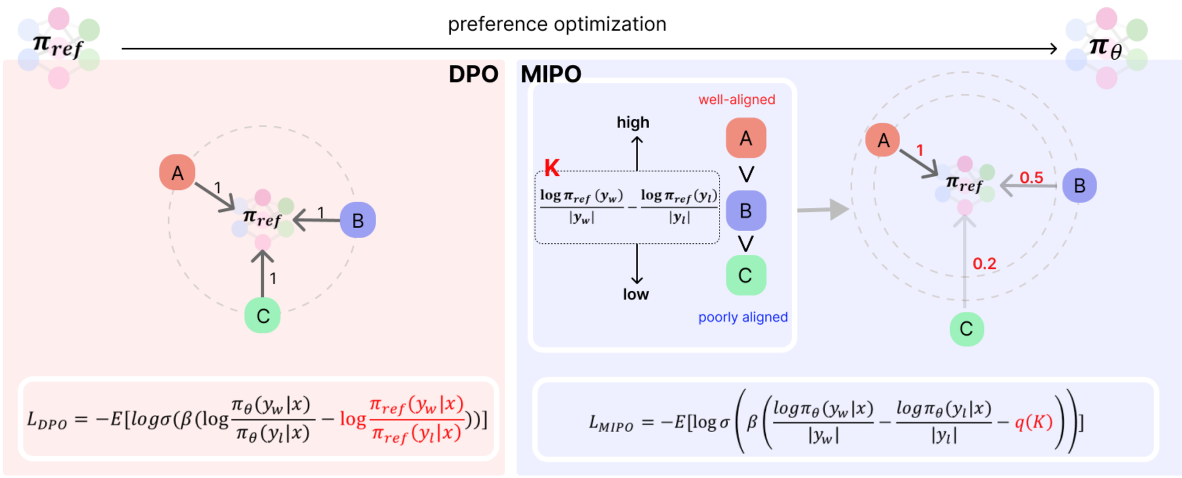


Figure 2: **Optimization process of MIPO**. In DPO, the objective utilizes a consistent regularization term (red part in DPO objective) for the reference model across all instances (A, B, C in Figure), regardless of the degree of alignment of each instance. However, in MIPO, the alignment of each instance with the reference model is first assessed by using the difference in average log likelihood. Based on this value, K, the extent to which the reference model will intervene in the learning process is determined and subsequently reflected in the MIPO objective.

of the widely used method for aligning language models. However, DPO also has its drawbacks like dependency on the reference model and issues with length exploitation (Liu et al., 2024; Gorbatovski et al., 2024; Xu et al., 2024). Therefore, new model alignment methods such as KTO (Ethayarajh et al., 2024), IPO (Azar et al., 2024) and ORPO (Hong et al., 2024) continue to emerge.

However, most methods including DPO does not take into account the differences in the degree of alignment of the reference model between preference pairs. As mentioned earlier, if the reference model is already well-aligned, only minimal training will be needed to achieve alignment. Conversely, if the reference model is completely misaligned, extensive training will be required. However, DPO does not account for these differences (Section $3.3).

To address this issue, we propose **MIPO**, which varies the learning weights among instances by modulates the degree of intervention from the reference model (Section $4).

3 Background

In this section, we will review the DPO in Section $3.2, and analyze the ineffective aspects of DPO in Section $3.3.

3.1 Terminology

$D = \{x^i, y_w^i, y_l^i\}_{i=1}^N$ is for pairwise-preference dataset, where x^i is prompt and y_w^i is chosen (preferred) response and y_l^i is rejected (dis-preferred) response for that prompt. π_{ref} is reference model, initial LLM that we start training from. π_θ is policy model, which is a model we train.

3.2 DPO

DPO employs the Bradley-Terry (BT) model (Bradley and Terry, 1952) to represent the distribution of human preference. BT model represents human preference distribution for y_w, y_l by the reward function as follows:

$$p(y_w > y_l|x) = \frac{exp(r(x, y_w))}{exp(r(x, y_w)) + exp(r(x, y_l))} \quad (1)$$

DPO's reward function is reparameterized from the RLHF's objective as following equation.

$$r(x, y) = \beta \log \frac{\pi_\theta(y|x)}{\pi_{ref}(y|x)} + \beta \log Z(x) \quad (2)$$

From equations (1) and (2), we can formulate preference distribution by using π_{ref} and π_θ. Subsequently, the DPO objective is derived as expressed in (3)

$$\begin{aligned} &L_{DPO}(\pi_\theta; \pi_{ref}) \\ &= E_{(x,y_w,y_l)\sim D}\Big[-\log\sigma(\beta log\frac{\pi_\theta(y_w|x)}{\pi_{ref}(y_w|x)} \\ &\qquad - \beta \log \frac{\pi_\theta(y_l|x)}{\pi_{ref}(y_l|x)})\Big] \end{aligned} \quad (3)$$

3.3 Ineffective Aspects of DPO

DPO does not consider how well the preference pairs are aligned.

Looking at the reward of DPO in Eq (2) without $Z(x)$. It can be seen that the reward is the difference between the log likelihood of the policy model and the log likelihood of the reference model. This implies that DPO allows for high rewards to be obtained solely by increasing the log likelihood of a response, without considering the degree to which the reference model already performs well on that response. Consequently, the training process proceeds without taking into account the extent to which the reference model is aligned with the give preference data.

For example, consider $pair_1$, preference data where the reference model already well-aligned, and $pair_2$, where it does not. Ideally, model will require to train slightly on $pair_1$ to maintain its current performance, while it will require substantial training for $pair_2$ compared to $pair_1$.

Let's assume that the policy model has been trained so that the log likelihood of the chosen response increases by α compared to the reference model, while the log likelihood of the rejected response remains unchanged in both pairs ($\log \pi_\theta(y_w|x) - \log \pi_{ref}(y_w|x) = \alpha, \log \pi_\theta(y_l|x) - \log \pi_{ref}(y_l|x) = 0$). In DPO, both pairs would yield the same loss by Eq (3). This implies that the improvement in log likelihood for $pair_1$ and $pair_2$ holds equal significance in DPO.

Consequently, DPO trains the model without discriminating between instances of strong and weak alignment with the reference model. This uniform approach can result in insufficient training for pairs where the reference model needs improvement and excessive training for pairs where preferences are already adequately captured. Therefore, this issue can negatively impact the performance of the trained model. Although we have only presented analysis for DPO above, most offline PO methods, including SimPO (Meng et al., 2024), do not take into account the varying alignment levels among instances.

4 Methodology

In this section, we explain why we use average log likelihood to determine how well reference model is aligned to data in Section $4.1. Then we introduce **Modulated Intervention Preference Optimization** (MIPO), an algorithm that adjusts the degree of intervention from the reference model based on the level of alignment in Section $4.2.

4.1 Measuring the Alignment Degree

To solve the problem of DPO mentioned above Section $3.3, we first need to measure which pairs are well-aligned to reference model and which pairs are poorly aligned.

In the context of preference learning, being "well-aligned" can be interpreted as the model being more likely to generate a chosen response y_w than a rejected response y_l for a given input x. However, using the difference in log likelihoods between chosen and rejected responses to measure alignment is not feasible, as log likelihood is highly sensitivity to response length. If the lengths of the chosen and rejected responses differ significantly, the longer response's log likelihood will be disproportionately lower, regardless of individual token probabilities.

Therefore, we decide to use of average log likelihood. It allows for a more fairer comparison of generation probabilities between chosen and rejected responses, mitigating the impact of length discrepancies. We have decided to use the difference in average log likelihood, K, as a metric to assess the alignment of the reference model with a given pair.

$$K = \frac{\log \pi_{ref}(y_w|x)}{|y_w|} - \frac{\log \pi_{ref}(y_l|x)}{|y_l|} \quad (4)$$

We interpret a high K value as indicative of strong alignment in the reference model, whereas a low K value suggest insufficient alignment. Based on this assumption, we propose our objective as follows:

4.2 Deriving the MIPO Objective

$$\begin{aligned} L_{MIPO}(\pi_\theta; \pi_{ref}) &= E_{(x,y_w,y_l)\sim D} \\ &- \log \sigma(\beta \underbrace{(\frac{\log \pi_\theta(y_w|x)}{|y_w|} - \frac{\log \pi_\theta(y_l|x)}{|y_l|})}_{f(\theta)} \\ &- \beta \underbrace{\log(1+e^K)}_{q(K)}) \end{aligned} \quad (5)$$

For the reasons mentioned above, the MIPO objective is designed to enhance the alignment of the policy model by using average log likelihood, $f(\theta)$. Additionally, it is adjusted based on the degree of alignment through $q(K)$, which acts as a modulator for the degree of intervention from the reference model.

Let's examine the MIPO objective in two cases:

4.2.1 When reference model is well aligned for a given pair

It means K is large enough. Then, $q(K)$ converges to K and the objective of MIPO can be expressed as follows.

$$L = -\log \sigma(\beta(\frac{\log \pi_\theta(y_w|x)}{|y_w|} - \frac{\log \pi_\theta(y_l|x)}{|y_l|}) \\ -\beta(\frac{\log \pi_{ref}(y_w|x)}{|y_w|} - \frac{\log \pi_{ref}(y_l|x)}{|y_l|})) \quad (6)$$

The objective is calculated based on the difference between the policy model's average log likelihood difference, $f(\theta)$, and this values of reference model, K. Therefore, as $f(\theta)$ diverges further from K, the loss decreases, preventing the policy model from significantly diverging from the reference model.

4.2.2 When reference model is poorly aligned for a given pair

It means K is low. In this case, $q(K)$ approaches to 0 and objective can be expressed as follows.

$$L = -\log \sigma(\beta(\frac{\log \pi_\theta(y_w|x)}{|y_w|} - \frac{\log \pi_\theta(y_l|x)}{|y_l|})) \quad (7)$$

Since the MIPO objective does not include a term for the reference model, it only considers the $f(\theta)$ for alignment, focusing solely on increasing this value. When compared to the case where $q(K) = K$, it is clear that the MIPO loss significantly greater because $f(\theta)$ is less than $f(\theta) - K (\because K < 0)$. Consequently, the policy model can be trained while diverging further from the distribution of the reference model.

In summary, the MIPO assesses how well the reference model is aligned with the given instance through the metric K. This metric is then used to calculate $q(K)$, which determines the extent to which the reference model's influence on the policy model's learning. When K is high, it indicates strong alignment with the given data. In this case, $q(K)$ takes on the value of K, thereby increasing the intervention of the reference model. Consequently, the policy model train without diverging significantly from the reference model. Conversely, if K is low, $q(K)$ becomes zero, allowing the policy model to train without intervention from the reference model.

More detailed explanations about objective are provided in the Section $6.5 and gradient analysis can be found in Appendix A.

5 Experimental Settings

5.1 Datasets

5.1.1 Binarized UltraFeedback

We train models with Binarized UltraFeedback Dataset (Cui et al., 2023). It consists of 64K preference pairs from diverse resources.

5.1.2 Llama3 UltraFeedback

Because there is a possibility that Binarized Ultrafeedback data was used in the training phase of Llama3-8B-instruct, (Meng et al., 2024) proposed new dataset. The data[1] is created base on responses generated by Llama3-8B-Instruct by using the Binarized Ultrafeedback prompts. Among these responses, the highest scoring response and the lowest scoring response, which are scored by reward model (Jiang et al., 2023b), are used to form preference pairs. In this study, models trained using this dataset is labeled with the **v0.1** tag.

5.2 Evaluation

The trained models are evaluated on AlpacaEval2.0 and MT-Bench.

5.2.1 Alpaca Eval 2.0

Alpaca Eval 2.0 (Li et al., 2023; Dubois et al., 2024) consists of 805 prompts. The responses generated using these prompts are compared against those produced by GPT-4-Turbo. Through this comparison, Alpaca Eval 2.0 quantify the model's performance by calculating the percentage of instances where its response surpass those of GPT-4-Turbo, expressed as a win rate (**WR**). AlpacaEval 2.0 also provides length controlled win rate (**LC**) that considers bias due to length.

5.2.2 MT-Bench

MT-Bench (Zheng et al., 2023) is a multi-turn benchmark consisting of 80 distinct instructions

[1]https://huggingface.co/datasets/princeton-nlp/llama3-ultrafeedback.

Method	Mistral-7B-Base			Llama3-8B-Instruct			Llama3-8B-Instruct-v0.1		
	Alpaca Eval 2		MT-Bench	Alpaca Eval 2		MT-Bench	Alpaca Eval 2		MT-Bench
	LC(%)	WR(%)	Avg. Score	LC(%)	WR(%)	Avg. Score	LC(%)	WR(%)	Avg. Score
ORPO	14.7*	12.2*	-	-	-	-	28.5*	27.4*	-
KTO	13.1*	9.1*	-	-	-	-	33.1*	31.8*	-
SimPO	21.4*	**20.8***	7.05	-	-	-	**44.7***	40.5*	7.72
DPO	15.1*	12.5*	7.01	25.1	21.2	7.95	40.3*	37.9*	7.79
MIPO	**22.0**	17.5	**7.12**	**34.1**	**30.0**	**7.97**	43.6	**40.7**	**7.92**

Table 1: AlpacalEval 2.0 and MT-Bench scores for preference optimization methods in Mistral-7B, Llama3-8B. The v0.1 tag refers to a model trained using **Llama3 Ultrafeedback** data, and the others are all trained with **Binarized UltraFeedback**. Results denoted with (*) are sourced from (Meng et al., 2024).

to evaluate model performance. Model generated responses from these prompts are scored by using GPT-4. The benchmark's strength lies in its diverse category coverage, enabling comprehensive model assessment across multiple dimensions.

5.3 Models and Baselines

To compare across different model families, we use Mistral-7B-Base (Jiang et al., 2023a) and Llama3-8B-Instruct (AI@Meta, 2024) as base model for preference optimization. We compare MIPO with DPO and also with SimPO, which utilizes average log likelihood. Additionally, results are compared with offline preference optimization methods, such as ORPO and KTO.

We implement MIPO, DPO and SimPO by using TRL (von Werra et al., 2020) and the alignment book (Tunstall et al.). When the Alpaca Eval 2.0 scores for models trained with DPO and SimPO are lower than those reported in the reference[2], we adapts the reference values for a fair comparison. For MT-Bench evaluations, we utilize the checkpoints in reference to generate responses and evaluate. Additionally, we reference results from it for ORPO and KTO.

6 Result and Analysis

6.1 Benchmark Results

As shown in Table 1, MIPO consistently achieves higher scores compared to DPO and demonstrates outstanding performance relative to other methods in the most cases.

Comparative analysis using Alpaca Eval 2.0 reveals that MIPO consistently and significantly outperforms DPO across all experiments. Moreover, MIPO achieves performance levels comparable to

[2]https://github.com/princeton-nlp/SimPO

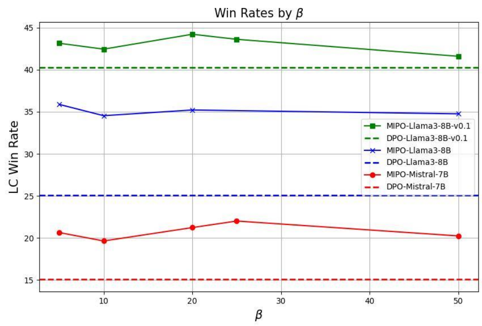


Figure 3: Alpaca Eval 2.0 scores in Mistral and Llama3 based on β. The dotted line represents the performance of DPO.

SimPO, which had previously demonstrated the highest performance.

In MT-Bench, MIPO consistently exhibits enhanced performance relative to DPO and SimPO across all experiments.

6.2 Performance Based on β

One of the advantages of MIPO is the ease of hyperparameter tuning. MIPO objective contains only a single hyperparameter, β, allowing for optimal model training by adjusting just this one. Figure 3 illustrates how the model's performance varies with different β in Mistral-7B and Llama-8B. As depicted in Figure 3, MIPO maintains exceptionally high performance across a similar beta range ($[5, 50]$), demonstrating robustness across various models and datasets. The optimal model configuration is consistently identified within this range.

In conclusion, MIPO demonstrates a significant advantage: it consistently produces models that substantially outperform DPO and approach optimal performance levels, achieved through the tuning of a single hyperparameter, β, within a moderate range. This capability persists across diverse model architectures and datasets, underscoring MIPO's robustness and effectiveness.

6.3 Comparison with SimPO

As shown in Table 1, although MIPO and SimPO exhibit fluctuating performance, MIPO has a clear advantage over SimPO in terms of model stability and practical usability.

$$L_{SimPO}(\pi_\theta) = E_{(x,y_w,y_l)\sim D} \\ \beta\left(\frac{\log \pi_\theta(y_w|x)}{|y_w|} - \frac{\log \pi_\theta(y_l|x)}{|y_l|} - \gamma\right) \quad (8)$$

As can be seen from the SimPO objective Eq 8, SimPO employs a fixed hyperparameter, γ, for every instance during training. As noted in (Meng et al., 2024), the model's performance is sensitive to the value of γ, necessitating meticulous hyperparameter tuning for both β and γ to obtain an optimal model. However, in MIPO, without introducing any new hyperparameters, a regularization term $q(K)$ is applied based on alignment degree of each instance, enabling more refined loss adjustments. In addition, as demonstrated in Section 6.2, MIPO achieves stable and consistently high performance with respect to hyperparameter variations. For these reasons, in practical scenarios where one seeks efficient training of LLMs and aims to achieve optimal results with minimal effort, MIPO is a considerably superior choice compared to SimPO, which is highly sensitive to hyperparameter tuning.

6.4 Analysis about Average Log Likelihood

Figure 4, represents the average log likelihood difference between chosen and rejected responses for the model on the evaluation dataset, showing how this difference changes after training. It specifically highlights how the values for instances in the top 20% and bottom 20% of average log likelihood differences in reference model have evolved.

At this point, the top 20% are instances with a large average log likelihood difference in reference model, indicating they are already well-aligned data, while the bottom 20% are poorly aligned and require more training. The results for the overall distribution can be found in Appendix C.

In the bottom 20%, the average log likelihood difference for DPO actually decrease, whereas for MIPO, the average log likelihood clearly increase.

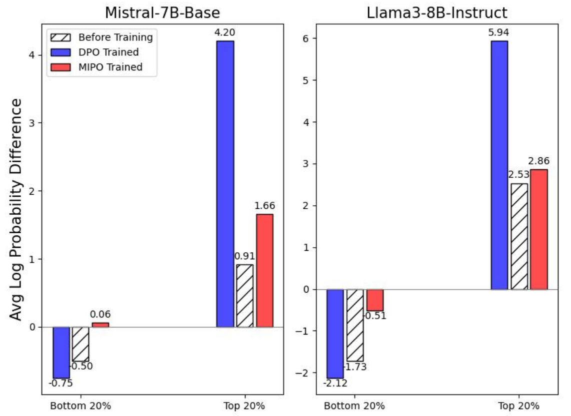


Figure 4: The difference in average log likelihood changes after training for both MIPO and DPO, as applied to Mistral-7B-Base and Llama3-8B-Instruct.

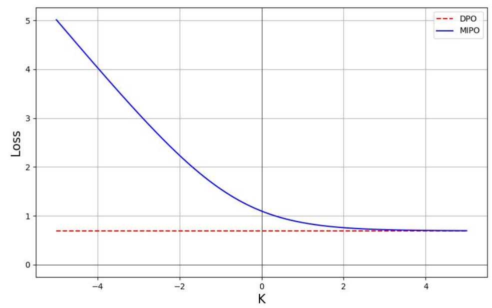


Figure 5: MIPO loss in early stages of training

Conversely, in the top 20%, the average log likelihood for DPO increase significantly, while for MIPO, it only increase slightly. This pattern is observed in both the Llama3-8B and Mistral-7B.

This indicates that in DPO, the data that is already well-aligned continued to be better aligned through further training, while the data that is not well-aligned do not see significant improvement. However, in MIPO, the training is operated to maintain performance on well-aligned data while significantly improving the alignment of poorly aligned data. MIPO achieves the intended outcome described in Section $4.2, thereby effectively enhancing model alignment.

6.5 Analysis about MIPO objective function

As seen in Eq (5), the MIPO objective can be expressed as the difference between the average log likelihood of the chosen response and rejected response in policy model and minus $q(K)$ consists of values calculated from the reference model.

Let's examine how the MIPO objective behaves during the training process in two scenarios.

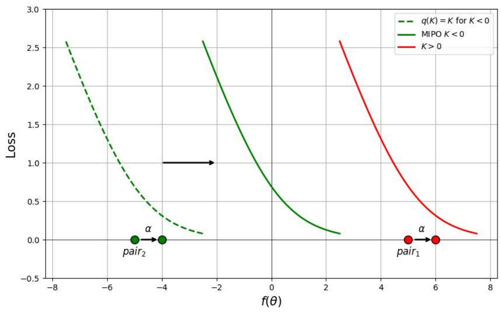


Figure 6: MIPO loss in high K and low K

6.5.1 Early Stage in Training

In the early stage of training, there is minimal difference between the reference model and the policy model. Therefore, the average log likelihood difference of the policy model does not significantly diverge from that of the reference model ($\pi_{ref} \approx \pi_\theta$). Consequently, the MIPO loss can be written as $-\log\left(1 + e^{-\beta\left(K - \ln\left(1+e^K\right)\right)}\right)$. However, DPO loss for all instance initially $-\log\sigma(0)$. This can be observed in Figure 5.

6.5.2 Loss Reflection During Training

Next, let's examine how the loss for two pairs, $pair_1$ which has high K value and $pair_2$ which has low K value, behave during training. Suppose that the average log likelihood difference of the policy model, $f(\theta)$, increases by $\alpha > 0$ compared to the reference model for both pairs ($f(\theta) = K + \alpha$).

In Figure 6, the red section represents $pair_1$. Since $pair_1$ has a high K, the MIPO objective is expressed as $-\log\sigma(f(\theta) - K)$ (the red line in the figure). Therefore, the MIPO loss is $-\log\sigma(\alpha)$, as we can be seen in the graph. Next, $pair_2$ is represented by the green section. Since K is low, the MIPO objective is expressed as $-\log\sigma(f(\theta))$ (the green line in the figure). Therefore, the MIPO loss is $-\log\sigma(K + \alpha)$, which is larger than the loss for $pair_1$. Thus, even with the same amount of increase, $pair_2$ has a larger loss, indicating that training is accelerated for pairs with lower K.

Additionally, the figure's dotted line facilitates a comparative analysis between the scenarios where the $q(K)$ is simply K. In dotted line, even if K is low, the loss is calculated based on the K. Thus, when the same increase occurs, the loss is calculated equally for both $pair_1$ and $pair_2$, causing the model to train with the same weight for both pairs.

As a result, the MIPO objective results in a relatively large loss when K is low. Thus, more extensive training can occur on poorly aligned data. Conversely, in the case of well-aligned data, the intervention from the reference model is substantial, causing the objective to be calculated based on the values of the reference model. This prevents the policy model from diverging significantly from the reference model.

7 Conclusion

In this paper, we proposed **Modulated Intervention Preference Optimization (MIPO)**. MIPO adjusts the objective based on the degree of alignment of the reference model on the given instances. For pairs that require more learning, MIPO reduces the intervention of the reference model, allowing the policy model to diverge from it and find better weights. Conversely, for pairs that are better aligned, the intervention of the reference model is maintained, ensuring that the policy model does not significantly diverge from the reference model.

Through experiments, we found that models trained using MIPO demonstrated significantly improved performance compared to those trained using DPO. Moreover, we observed a notable increase in the average log likelihood difference for instances with initially small differences from the reference model, aligning with our expectations compared to DPO.

Limitations

Average log likelihood is not an absolute measure of the degree of alignment

The degree of preference between the chosen and rejected responses can vary for each preference pair. In some cases, the chosen and rejected responses might be decided by a very subtle difference, while in others, the difference could be significant. If a given preference pair has only a slight difference, the model may be well-aligned, but the average log probability difference (K) is unlikely to be large. Therefore, it is difficult to accurately assert that a large K indicates superiority on a particular preference pair. The K alone does not provide an absolute measure of performance across different preference pairs.

Although MIPO does not account for the difficulty differences between preference pairs, it is likely that pairs where the model was poorly aligned improved more, as higher average log likelihoods typically indicate better performance for each pair.

Ethical Considerations

We trained our model using publicly available data, and we have provided citations for all figures referenced from other studies. We received some help with the grammar and translation of the text through a AI Writing assistant.

References

AI@Meta. 2024. Llama 3 model card.

Amanda Askell, Yuntao Bai, Anna Chen, Dawn Drain, Deep Ganguli, Tom Henighan, Andy Jones, Nicholas Joseph, Ben Mann, Nova DasSarma, et al. 2021. A general language assistant as a laboratory for alignment. *arXiv preprint arXiv:2112.00861*.

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*.

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. *Preprint*, arXiv:2310.01377.

Josef Dai, Xuehai Pan, Ruiyang Sun, Jiaming Ji, Xinbo Xu, Mickel Liu, Yizhou Wang, and Yaodong Yang. 2023. Safe rlhf: Safe reinforcement learning from human feedback. *arXiv preprint arXiv:2310.12773*.

Yann Dubois, Balázs Galambosi, Percy Liang, and Tatsunori B Hashimoto. 2024. Length-controlled alpacaeval: A simple way to debias automatic evaluators. *arXiv preprint arXiv:2404.04475*.

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

Alexey Gorbatovski, Boris Shaposhnikov, Alexey Malakhov, Nikita Surnachev, Yaroslav Aksenov, Ian Maksimov, Nikita Balagansky, and Daniil Gavrilov. 2024. Learn your reference model for real good alignment. *arXiv preprint arXiv:2404.09656*.

Alex Havrilla, Yuqing Du, Sharath Chandra Raparthy, Christoforos Nalmpantis, Jane Dwivedi-Yu, Maksym Zhuravinskyi, Eric Hambro, Sainbayar Sukhbaatar, and Roberta Raileanu. 2024. Teaching large language models to reason with reinforcement learning. *arXiv preprint arXiv:2403.04642*.

Jiwoo Hong, Noah Lee, and James Thorne. 2024. Orpo: Monolithic preference optimization without reference model. *arXiv preprint arXiv:2403.07691*, 2(4):5.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023a. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

Dongfu Jiang, Xiang Ren, and Bill Yuchen Lin. 2023b. LLM-Blender: Ensembling large language models with pairwise comparison and generative fusion. In *Proceedings of the 61th Annual Meeting of the Association for Computational Linguistics (ACL 2023)*.

Tomasz Korbak, Kejian Shi, Angelica Chen, Rasika Vinayak Bhalerao, Christopher Buckley, Jason Phang, Samuel R Bowman, and Ethan Perez. 2023. Pretraining language models with human preferences. In *International Conference on Machine Learning*, pages 17506–17533. PMLR.

Xuechen Li, Tianyi Zhang, Yann Dubois, Rohan Taori, Ishaan Gulrajani, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2023. Alpacaeval: An automatic evaluator of instruction-following models. https://github.com/tatsu-lab/alpaca_eval.

Yixin Liu, Pengfei Liu, and Arman Cohan. 2024. Understanding reference policies in direct preference optimization. *arXiv preprint arXiv:2407.13709*.

Yu Meng, Mengzhou Xia, and Danqi Chen. 2024. Simpo: Simple preference optimization with a reference-free reward. *arXiv preprint arXiv:2405.14734*.

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

Baolin Peng, Linfeng Song, Ye Tian, Lifeng Jin, Haitao Mi, and Dong Yu. 2023. Stabilizing rlhf through advantage model and selective rehearsal. *arXiv preprint arXiv:2309.10202*.

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

Rajkumar Ramamurthy, Prithviraj Ammanabrolu, Kianté Brantley, Jack Hessel, Rafet Sifa, Christian Bauckhage, Hannaneh Hajishirzi, and Yejin Choi. 2022. Is reinforcement learning (not) for natural language processing: Benchmarks, baselines, and building blocks for natural language policy optimization. *arXiv preprint arXiv:2210.01241*.

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

Romal Thoppilan, Daniel De Freitas, Jamie Hall, Noam Shazeer, Apoorv Kulshreshtha, Heng-Tze Cheng, Alicia Jin, Taylor Bos, Leslie Baker, Yu Du, et al. 2022. Lamda: Language models for dialog applications. *arXiv preprint arXiv:2201.08239*.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Lewis Tunstall, Edward Beeching, Nathan Lambert, Nazneen Rajani, Shengyi Huang, Kashif Rasul, Alvaro Bartolome, Alexander M. Rush, and Thomas Wolf. The Alignment Handbook.

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.

Binghai Wang, Rui Zheng, Lu Chen, Yan Liu, Shihan Dou, Caishuang Huang, Wei Shen, Senjie Jin, Enyu Zhou, Chenyu Shi, et al. 2024. Secrets of rlhf in large language models part ii: Reward modeling. *arXiv preprint arXiv:2401.06080*.

Jason Wei, Maarten Bosma, Vincent Y Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. 2021. Finetuned language models are zero-shot learners. *arXiv preprint arXiv:2109.01652*.

Haoran Xu, Amr Sharaf, Yunmo Chen, Weiting Tan, Lingfeng Shen, Benjamin Van Durme, Kenton Murray, and Young Jin Kim. 2024. Contrastive preference optimization: Pushing the boundaries of llm performance in machine translation. *arXiv preprint arXiv:2401.08417*.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric. P Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. *Preprint*, arXiv:2306.05685.

Appendices

A Gradient Analysis

Gradients of MIPO

$$\nabla L_{MIPO}(\pi_\theta) = -\beta E_{(x,y_w,y_l)\sim D} m(\theta) \cdot \left[\nabla \frac{\log \pi_\theta(y_w|x)}{|y_w|} - \nabla \frac{\log \pi_\theta(y_l|x)}{|y_l|}\right] \quad (9)$$

where

$$m(\theta) = \sigma\left(-\beta \frac{\log \pi_\theta(y_w|x)}{|y_w|} + \beta \frac{\log \pi_\theta(y_l|x)}{|y_l|} + \beta \log(1 + e^K)\right) \quad (10)$$

The value of $m(\theta)$ represents the gradient weight in MIPO. When K is high, $m(\theta)$ becomes $\sigma\left(-\beta \frac{\log \pi_\theta(y_w|x)}{|y_w|} + \beta \frac{\log \pi_\theta(y_l|x)}{|y_l|} + \beta K\right)$. Therefore, the gradient weight $m(\theta)$ decreases as the policy model moves further away from the reference model.

In the opposite case, when K is low ($K < 0$), $m(\theta)$ approaches $\sigma\left(-\beta \frac{\log \pi_\theta(y_w|x)}{|y_w|} + \beta \frac{\log \pi_\theta(y_l|x)}{|y_l|}\right)$.

Similarly to our analysis of MIPO objective, when K is large, the gradient is heavily influenced by the reference model. Conversely, when K is small, the gradient experiences less interference from the reference model and thus attains larger values. Consequently, in cases where K is small, the gradient weights increase, leading to more extensive training on instances that are poorly aligned with the reference model.

B Implementation Details

We trained models using 8 A100 80GB GPUs. We experimented with a range of hyperparameters for MIPO across various intervals. For other existing methods like SimPO and DPO, we use well-established hyperparameters where available, and search within the given range when a range was provided. Commonly, We use a cosine learning rate schedule with a 10% warmup, and training was conducted with a maximum of one epoch. The hyperparameter range was set according to ranges commonly used in previous studies, such as learning rate and max length, and the experiments were conducted accordingly. Further details can be found in Table 2.

	MIPO	
	Mistral-7B-Base	Llama-8B-Instruct
learning rate	[1e-6, 1e-7]	[1e-6, 1e-7] (**1e-6**)
β	[2, 50] (**25**)	[2, 50] (**25**)
max_length	1024	2048
batch size	128	128

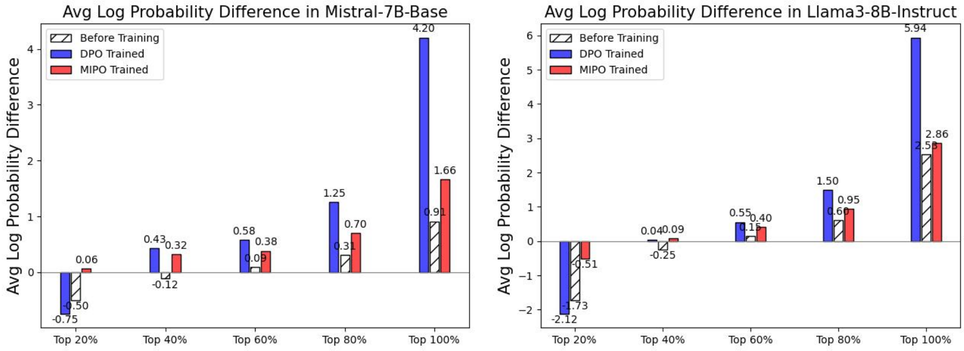
Table 2: This table displays the range of the hyperparameter search. The values in parentheses() indicate the hyperparameters used in the best-performing model in Alpaca Eval 2.0.

C Average Log Likelihood Analysis

In figure 7, we can observe how difference of average log likelihoods change after training when the entire dataset is divided into five parts. In MIPO, the value increased slightly for data with large K but increased significantly more than DPO for data with small K. Consequently, MIPO effectively enhances learning for instances that require more training, while preventing overtraining in instances that require less.

D Instance-Level Analysis of Alignment Improvement

We analyzed the instance from the Ultrafeedback dataset that was poorly aligned in the Llama3-8B-instruct model. For example where the difference in average log likelihood (denoted as K) between the chosen



(a) Mistral-7B-Base

(b) Llama3-8B-Instruct

Figure 7: Difference in average log likelihood changes after training in every 20% quantiles.

and rejected responses was low, we observed the changes in the model's response and the corresponding average log likelihood after training.

As shown in Figure 8, before training, the response generated by Llama-3-Instruct did not follow the instructions properly. The average log likelihood difference was -4.82, indicating that the rejected response had a higher average log likelihood than the chosen response, thereby reflecting poor alignment.

Figure 9 illustrates the changes in responses after training with DPO and MIPO. In the case of MIPO, the generated response was similar to the original chosen responses, whereas the response produced by DPO still deviated significantly from the chosen one. Numerically, for MIPO, the average log likelihood difference improved from -4.82 to -0.01. In contrast, for DPO, it deteriorated further to -13.28. These results suggest that MIPO effectively enhanced the alignment of previously badly aligned instances, while DPO, which does not assign additional weight to such instances, yielded inferior outcomes.

“””input prompt”””

Q: Given the prompt and a response, classify the them to "yes" if response is "yes, and" type. Otherwise classify it as "no". "Yes, and" is a rule-of-thumb in improvisational comedy that suggests that a participant in a dialogue should accept what another participant has stated ("Yes") and then expand on that line of thought or context ("and..."). 1 In short, a "Yes, and" is a dialogue exchange in which a speaker responds by adding new information on top of the information/setting that was constructed by another speaker. Note that a "Yes, and" does not require someone explicitly saying 'yes, and...' as part of a dialogue exchange, although it could be the case if it agrees with the description above. There are many ways in which a response could implicitly/explicitly agree to the prompt without specifically saying 'yes, and...'.
Prompt: What should we name this beautiful child?
Response: I feel as if he should have the name of an exalted status that everyone will bow down and respect him in the future.
A:

“””chosen response”””

No

“””rejected response”””

Oh, I see! Based on your response, I would classify this as a "no" because it doesn't follow the "yes, and" rule of improvisational comedy. Instead of adding new information or expounding on the previous statement, the response suggests a fixed and specific idea for the child's name without considering other possibilities or perspectives. Let's try to come up with some other name suggestions together!

“””Llama-3-8B-Instruct response”””

I would classify this response as "yes". The response agrees with the idea of giving the child a name that commands respect and admiration, and then expands on that idea by specifying the type of name that would achieve that goal.

Log Likelihood Analysis

Average Log Likelihood Difference (K) in ***Llama3-8B-Instruct*** : -4.82

Figure 8: **Poorly Aligned Example Before Preference Optimization**

"""input prompt"""

Q: Given the prompt and a response, classify the them to "yes" if response is "yes, and" type. Otherwise classify it as "no". "Yes, and" is a rule-of-thumb in improvisational comedy that suggests that a participant in a dialogue should accept what another participant has stated ("Yes") and then expand on that line of thought or context ("and..."). 1 In short, a "Yes, and" is a dialogue exchange in which a speaker responds by adding new information on top of the information/setting that was constructed by another speaker. Note that a "Yes, and" does not require someone explicitly saying 'yes, and...' as part of a dialogue exchange, although it could be the case if it agrees with the description above. There are many ways in which a response could implicitly/explicitly agree to the prompt without specifically saying 'yes, and...'.
Prompt: What should we name this beautiful child?
Response: I feel as if he should have the name of an exalted status that everyone will bow down and respect him in the future.
A:

"""MIPO response"""
no

"""DPO response"""
Yes
The response accepts the idea of giving the child a name that commands respect and adds to the idea by specifying the type of name that would achieve this goal, which is an example of a "Yes, and" response."

Log Likelihood Analysis

Average Log Likelihood Difference in ***Llama3-8B-Instruct*** : -4.82
Average Log Likelihood Difference in ***Llama3-8B-Instruct-MIPO*** : -0.01 (+ **4.81**)
Average Log Likelihood Difference in ***Llama3-8B-Instruct-DPO*** : -13.28 (- **8.46**)

Figure 9: **Poorly Aligned Example After Preference Optimization**