Removing Prompt-template Bias in Reinforcement Learning from Human Feedback

Chaojie Wang[1*], **Haonan Shi**[2*], **Long Tian**[1,2†], **Bo An**[1,2], **Shuicheng Yan**[3],

[1]Skywork AI, [2]Xidian University,
[3]Nanyang Technological University, [4]National University of Singapore

{chaojie.wang, long.tian}@kunlun-inc.com

Abstract

Reinforcement Learning from Human Feedback (RLHF) has become an essential technique for enhancing pre-trained large language models (LLMs) to generate responses that align with human preferences and societal values. Although RLHF has shown promise, the training of reward models (RMs) still faces the challenge of *reward hacking*, motivating recent works to prevent RMs from finding shortcuts that bypass the intended optimization objectives by identifying simplistic patterns such as response length. Besides the issue of *length bias*, our work firstly reveals that *prompt-template bias* learned by RMs can also cause *reward hacking* when dealing with some marginal samples, resulting in LLMs preferring to generate responses in a specific format after RLHF fine-tuning, regardless of the format requested in the prompt. To this end, we propose a low-cost but effective method, namely Prompt Bias Calibration (PBC), to estimate the *prompt-template bias* term during reward modeling, which can be utilized to calibrate reward scores in the following RL fine-tuning process. Then, we show that our PBC method can be flexibly combined with existing algorithms of removing *length bias*, leading to a further improvement in the aspect of enhancing the quality of generated responses.

1 Introduction

Reinforcement Learning from Human Feedback (RLHF) has become a critical technique to enable pre-trained large language models (LLMs) to follow human instructions, understand human intent, and also generate responses that align with human preferences and societal values (Ouyang et al., 2022; Rafailov et al., 2024; Ethayarajh et al., 2024; Yin et al., 2024). Specifically, RLHF usually trains a reward model (RM) to act as the proxy of human preferences, and then utilizes online reinforcement learning (RL) algorithms to fine-tune the language models to generate responses that can achieve higher expectation rewards, leading to the success of ChatGPT and also many other AI applications (Team et al., 2023; Achiam et al., 2023). Although the paradigm of RLHF has simplified the human data collection procedure, as acquiring human ratings is much easier than collecting demonstrations for supervised fine-tuning (SFT), it still requires a large amount of human-annotated preference pairs to train well-performing RMs in practice, motivating recent researchers to look for novel alignment methods to bypass RM training (Rafailov et al., 2024; Ethayarajh et al., 2024; Yin et al., 2024). However, the original RLHF pipeline is still the primary choice for the industry, because well-trained RMs can provide a certain level of generalizability (Li et al., 2023).

In addition to the high costs associated with gathering large amounts of human-annotated preference data, another significant concern often raised about RLHF is the issue of *reward hacking* (Eisenstein et al., 2023), where the over-optimized RMs tend to find some shortcuts to bypass their intended optimization objective, through identifying some simple patterns to distinguish between good and bad responses (Gao et al., 2023). The most widely studied pattern in *reward hacking* could be the sentence (response) length, and these trained RMs can utilize the preference among human raters for longer responses to achieve *reward hacking*, despite the actual quality of response does not improve with the increase of response length (Singhal et al., 2023). Thus, to mitigate *reward hacking*, recent work has primarily focused on estimating the *length bias* term in the reward scoring process, so that it can be removed in the subsequent RL fine-tuning procedure to further improve the quality of generated responses (Chen et al., 2024; Shen et al., 2023).

Besides the issue of *length bias*, in the practice

[*]Equal Contribution
[†]Corresponding Author

Findings of the Association for Computational Linguistics: ACL 2025, pages 24110–24122

of applying RLHF to industrial products, we have observed that the original implementation of RLHF tends to make LLMs prefer generating responses in a specific format. This observation motivates us to investigate the underlying causes and find a cost-effective solution to address this issue. The main contributions are summarized as follows:

- We are the first to reveal the existence of *prompt-template bias* in RMs trained with Bradley-Terry preference loss, and theoretically analyze the cause of *prompt-template bias* issue, along with its corresponding potential risks on the entire RLHF process;
- To mitigate the *reward hacking* caused by *prompt-template bias*, we develop a Prompt Bias Calibration (PBC) method, which will estimate the *prompt-template bias* term during the reward scoring process, and then remove it in the subsequent RL fine-tuning process;
- We show that the developed PBC method can be flexibly combined with existing methods of removing *length bias*, leading to a further improvement in the aspect of enhancing the quality of generated responses;
- Experimental results show that our developed PCB method and its extensions can achieve promising performance improvements compared to the original implementation of RLHF.

2 Preliminary

2.1 Reward Model Training

The typical objective of optimizing a reward model is to minimize the loss based on the Bradley–Terry model (Bradley and Terry, 1952) using a dataset of pairwise comparisons of model responses, denoted as $(x, y^+, y^-) \in \mathcal{D}$ where x indicates the input prompt, y^+ and y^- are the chosen and rejected responses respectively. Then, the objective function can be formulated as

$$\mathcal{L}^{RM}(\theta) = - \mathbb{E}_{(x,y^+,y^-)\sim\mathcal{D}} \left[\log(\sigma(r_\theta(x, y^+) - r_\theta(x, y^-))\right] \quad (1)$$

where $r_\theta(x, y)$ denotes the reward model that takes the prompt x and response y as input to predict a scalar reward with trainable parameters θ; σ denotes the *Sigmoid* function.

Length Bias: Denote $r_{\theta^*}(x, y)$ as the "gold standard" reward model (Gao et al., 2023) with the optimal parameters θ^*, it reflects human's intrinsic ranking preferences and can play a role of human rater to provide gold reward signal for each prompt-response pair. However, due to the subjectivity of ranking preferences and flaws in rating criteria, there is a phenomenon where human raters prefer longer responses that appear more detailed or strictly formatted, but their actual quality does not improve (Singhal et al., 2023). Thus, the "gold standard" reward model for rating preference data can often be biased and thus we can decompose it to disentangle the actual reward from the spurious reward (Chen et al., 2024), formulated as

$$r_{\theta^*}(x, y) = r^Q_{\theta^*}(x, y) + r^L_{\theta^*}(x, y), \quad (2)$$

where $r^Q_{\theta^*}(x, y)$ is the actual reward gains brought by improving the quality of response y; $r^L_{\theta^*}(x, y)$ is the spurious reward gains of increasing response length, whose patterns are much easier to identify.

Thus, with *length bias* in the "gold standard" $r_{\theta^*}(x, y)$, during the training of reward model, $r_\theta(x, y)$ can easily find shortcuts to bypass its intended optimization objective, through identifying simple patterns, such as sentence (response) length, to distinguish between good and bad responses, leading to the phenomenon of "reward hacking" caused by *length bias* (Singhal et al., 2023). Without increasing the cost of rating higher quality preference data, it becomes increasingly important and beneficial to study mitigating the impact of *length bias* in the process of reward modeling.

Prompt Bias: the *prompt bias* in reward modeling derives from the underdetermination of Bradley-Terry model (Bradley and Terry, 1952). For any reward model $r_{\theta'}(x, y)$ learned from the preference loss defined in Eq. (1), whose target is optimized to approximate the "gold standard" $r_{\theta^*}(x, y)$, there always exists an equivalent reward model $r_\theta(x, y)$ that satisfies

$$r_\theta(x, y) := r_{\theta'}(x, y) + C(x) \quad (3)$$

where $C(x)$ is a prompt-dependent constant referred to as *prompt bias*, leading to the same loss value as $\mathcal{L}(\theta) = \mathcal{L}(\theta')$. Due to the fact that there is no constraint on $C(x)$ in the original preference loss as defined in Eq. (1), the issue of *prompt bias* has been criticized in the scenario of reward model ensembles (Eisenstein et al., 2023), where different reward models tend to choose different values for $C(x)$, making the statistics of the set of reward scores meaningless.

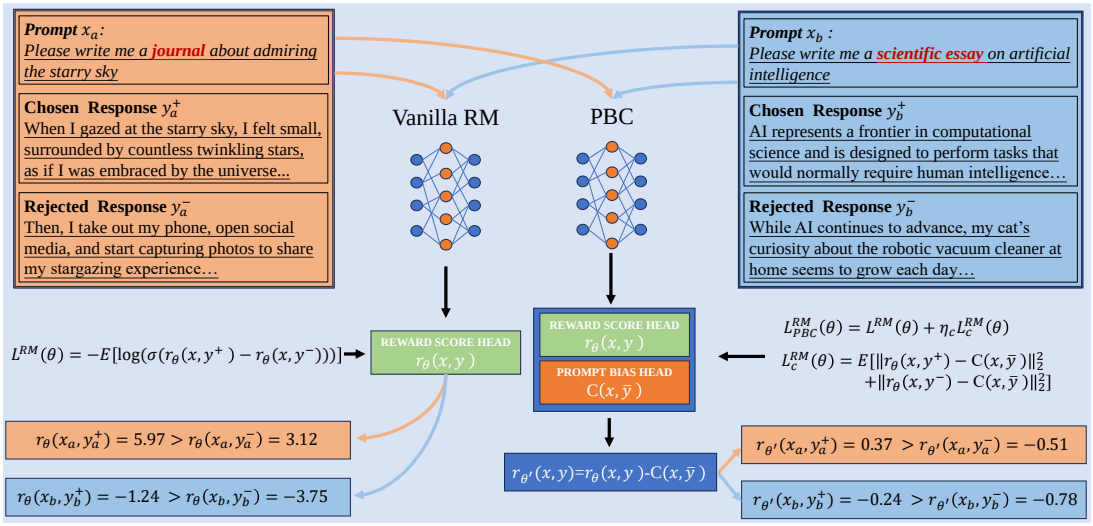


Figure 1: Comparison of the RM training process using the Bradley–Terry loss and our developed PBC method respectively, where the latter employs a bias head to approximate the *prompt-template bias* $C(x,\overline{y})$, providing unbiased reward distribution $r_{\theta'}(x,y)$ with a lower variance compared to $r_\theta(x,y)$ for the subsequent RL fine-tuning.

2.2 RLHF Fine-tuning

Given the trained reward model $r_\theta(x,y)$ as the proxy of human preferences, Reinforcement Learning from Human Feedback (RLHF) tends to utilize an online reinforcement learning method, typically proximal policy optimization (PPO) (Schulman et al., 2017), trains a policy language model π_ϕ^{RL} to maximize expected reward, while staying close to its initial policy π_ϕ^{SFT}, which is finetuned on supervised data (prompt-response pairs). Through measuring the distance from the initial policy with Kullback-Leibler (KL) divergence, the optimization objective can be formulated as

$$\mathcal{L}^{RL}(\phi) = \mathbb{E}_{(x,y)\sim\mathcal{D}_{\pi_\phi^{RL}}}\Big[r_\theta(x,y) - \beta \log\left[\pi_\phi^{RL}(y|x)/\pi^{SFT}(y|x)\right]\Big], \quad (4)$$

where β is the hyper-parameter to control the strength of the KL divergence term.

3 Method

3.1 What is *prompt-template bias*

In this part, we will first illustrate the cause of *prompt-template bias* during RM training. Formally, given a set of prompt-response pairs, denoted as $\mathcal{D}_a = \{x_a, y_a^{(i)}\}_{i=1}^{N_a}$, with the same user prompt x_a, *e.g. "writing an* ***academic paper*** *on the field of computer science"*, and $\{y_a^{(i)}\}_{i=1}^{N_a}$ denoting the set of collected ***academic papers*** to satisfy the request of x_a, the *prompt bias* term, specifically $C(x_a)$, learned by RMs is supposed to not affect the preference order within $\mathcal{D}_a$, as discussed in Section 2.1. However, in the practice of RM training, the reward score is usually predicted by a LLM that takes the concatenation of the prompt and response as input, making it challenging for RMs to learn a bias term that focuses solely on the prompt x while disregarding variations in the subsequent response y. During the training process to order the pairs within $\mathcal{D}_a$, we find that RMs trained with the Bradley–Terry loss in Eq. (1) are more likely to introduce a joint bias term across the entire sequence of concatenating the prompt and response, formulated as

$$r_\theta(x_a,y_a) := r_{\theta'}(x_a,y_a) + C(x_a,\overline{y}_a), \quad (5)$$

where $\overline{y}_a = \frac{1}{N_a}\sum_{i=1}^{N_a} y_a^{(i)}$ can be considered the average response of the response set $\{y_a^{(i)}\}_{i=1}^{N_a}$, and it will embody the common characteristics found within these collected responses, such as the format of ***academic paper***; $C(x_a,\overline{y}_a)$ denotes the joint bias on the entire sequence of the prompt x_a associated with the average response $\overline{y}_a$ in the format of ***academic paper***; $r_\theta(x_a,y_a)$ is still supposed to approximate the "gold standard" provided by $r_{\theta^*}(x_a,y_a)$, leading to $\mathbb{E}_{\mathcal{D}_a}[r_{\theta'}(x_a,y_a)] \approx \mathbb{E}_{\mathcal{D}_a}[r_{\theta^*}(x_a,y_a)]$.

Considering the average response $\overline{y}$ can be treated as a standard template of the response to the prompt x, we name the joint bias $C(x,\overline{y})$ as *prompt-template bias*. Then, we highlight the properties of *prompt-template bias* as follows:

- the typical preference loss in Eq. (1) imposes no constraints on $C(x, \overline{y})$, because its value will not influence the outcome of the preference loss and also not affect the preference order within the prompt-response pairs collected for the same prompt x;
- $C(x, \overline{y})$ will reduce to the original *prompt bias* $C(x, -)$ when no shared characteristics exist across all collected responses, implying that the diversity of collected responses $\{y^{(i)}\}_{i=1}^{N}$ is sufficiently high.

With these properties in mind, we assume that the *prompt-template bias* $C(x, \overline{y})$ can essentially meet most of the properties of the original *prompt bias* $C(x, -)$ as discussed in Section 2.1. Thus, we suppose $C(x, \overline{y})$ can be considered as a broader definition of *prompt bias* in the actual RM training, because it is more likely to be learned by RMs in practice, given the fact that preference pairs are extremely scarce and the diversity of responses collected for the same prompt is often insufficient.

3.2 Impact of *prompt-template bias* on RLHF

After defining *prompt-template bias*, we will theoretically investigate the impact of introducing $C(x, \overline{y})$ during RM training on the entire RLHF process. Assume that there exist two sets of prompt-response pairs, denoted as $\mathcal{D}_a = \{x_a, y_a^{(i)}\}_{i=1}^{N_a}$ and $\mathcal{D}_b = \{x_b, y_b^{(i)}\}_{i=1}^{N_b}$, where x_a and x_b indicate different categories of prompts, *e.g.* x_a requests *"writing an **academic paper** on theme **a**"* and x_b requests *"writing a **brief** on theme **b**"*, and $\{y_a^{(i)}\}_{i=1}^{N_a}$ and $\{y_b^{(i)}\}_{i=1}^{N_b}$ denote the collected responses for answering the prompt x_a and x_b respectively. Considering there is no constraint on $C(x, \overline{y})$ during training RM with the Bradley–Terry loss in Eq. (1), the discrepancies of prompt biases between these two previously mentioned sets of prompt-response pairs, specifically $\mathcal{D}_a$ and $\mathcal{D}_b$, could be extremely large, *e.g.* $C(x_a, \overline{y}_a) \gg C(x_b, \overline{y}_b)$, leading to

$$\mathbb{E}_{(x_a,y_a)\sim\mathcal{D}_a}\left[r_\theta\ (x_a, y_a)\right] \gg \mathbb{E}_{(x_b,y_b)\sim\mathcal{D}_b}\left[r_\theta(x_b, y_b)\right] \quad (6)$$

where $r_\theta(x_a, y_a) = r_{\theta'}(x_a, y_a) + C(x_a, \overline{y}_a)$ and $r_\theta(x_b, y_b) = r_{\theta'}(x_b, y_b) + C(x_b, \overline{y}_b)$. The reward distributions after removing *prompt-template* bias, modeling the reward scores $\{r_{\theta'}(x_a, y_a^{(i)})\}_{i=1}^{N_a}$ and $\{r_{\theta'}(x_b, y_b^{(i)})\}_{i=1}^{N_b}$ respectively, should exhibit similar mean values, *e.g.* $\mathbb{E}_{\mathcal{D}_a}\left[r_{\theta'}(x_a, y_a)\right] \approx \mathbb{E}_{\mathcal{D}_b}\left[r_{\theta'}(x_b, y_b)\right]$, and will make little impact on the comparison of expectation terms in Eq. (6). We highlight that the discrepancies of *prompt bias* terms, specifically the gap between $C(x_a, \overline{y}_a)$ and $C(x_b, \overline{y}_b)$, won't affect preference ordering within categories, but can cause disaster when dealing with some marginal samples, like *"an **academic paper** on theme **b**"* denoted as y_{ab}, or *"a **brief** on theme **a**"* denoted as y_{ba}.

To facilitate an intuitive analysis, we take the marginal sample *"an **academic paper** on theme **b**"*, denoted as y_{ab}, as an example. The reward scores for prompt-response pairs corresponding to the prompt x_b may exhibit the following orders:

$$\begin{aligned} r_\theta(x_b, y_{ab}) &= r_{\theta'}(x_b, y_{ab}) + C(x_b, \overline{y}_a) \\ &> r_{\theta'}(x_b, y_b) + C(x_b, \overline{y}_b) = r_\theta(x_b, y_b), \end{aligned} \quad (7)$$

which can be achieved as long as $r_{\theta'}(x_b, y_{ab}) \approx r_{\theta'}(x_b, y_b)$ and $C(x_b, \overline{y}_a) > C(x_b, \overline{y}_b)$. The first condition $r_{\theta'}(x_b, y_{ab}) \approx r_{\theta'}(x_b, y_b)$ can be achieved because both the response y_{ab} and y_b meet the description of theme b and are similar on a semantic level. The second inequality is highly likely to be achieved when there is a reward model that has a bias towards preferring the sentence in the format of a over b, specifically $C(x_a, \overline{y}_a) \gg C(x_b, \overline{y}_b)$.

Finally, we highlight that the phenomena of inequality in Eq. (7), caused by *prompt-template bias* $C(x, \overline{y})$, is commonly encountered in the deployment process of RLHF in real-world applications, especially text creation. For example, if responses are collected exclusively in the format specified by each prompt during RM training, the reward model may learn a bias toward specific response templates. Then, once such OOD marginal samples, *e.g* (x_b, y_{ab}), are generated by LLMs during the RL fine-tuning process and also satisfy the inequality $r_\theta(x_b, y_{ab}) > r_\theta(x_b, y_b)$ as shown in Table 1, the entire RL fine-tuning process will be biased and results in LLMs tend to generate responses in a specific format, regardless of the format you request in the prompt.

3.3 Calibrating *prompt-template bias* in RLHF

As shown in Fig. 1, the developed Prompt Bias Calibration (PBC) method primarily consists of two steps: 1) estimating the *prompt-template bias* term in the reward scoring process with minimal additional computational cost; 2) removing *prompt-template bias* in the subsequent RLHF fine-tuning

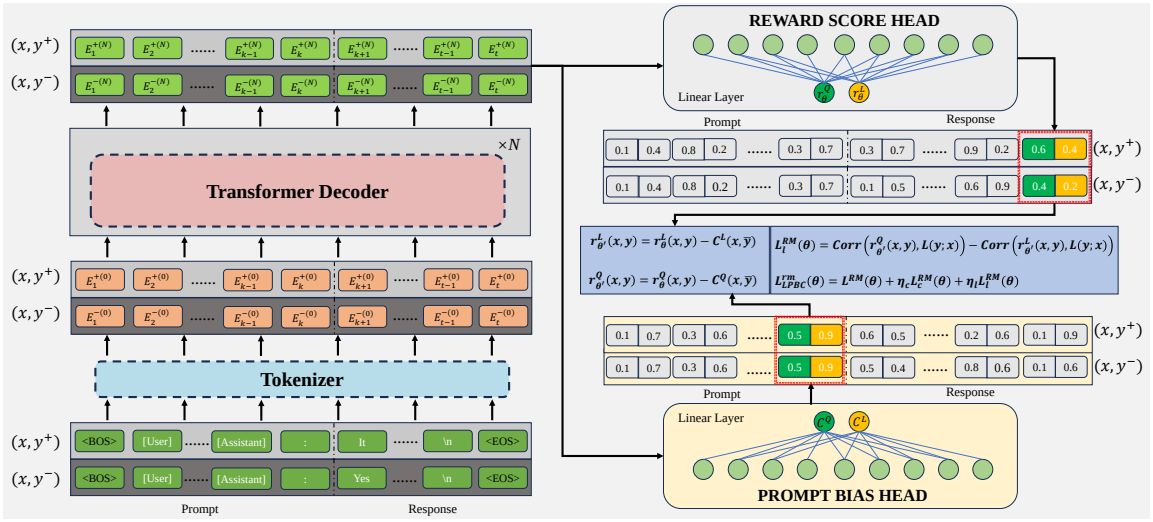


Figure 2: Network architecture design for training RMs with the LBPC method developed in Section 3.4, which incorporates a bias head on the last token of the prompt x designed to predict the *prompt-template bias* in the aspects of quality and length, specifically $C^Q(x, \overline{y})$ and $C^L(x, \overline{y})$, and a reward score head on the last token of the response y intended to predict the reward gains in the aspects of quality and length, specifically $r^Q_\theta(x, \overline{y})$ and $r^L_\theta(x, \overline{y})$.

process to prevent LLMs from developing a tendency to generate responses in a specific format. To approximate the *prompt-template bias* term $C(x, \overline{y})$ in Eq. (5), we choose to apply a linear layer on the last token of the prompt sentence to predict *prompt-template bias*, and then add the following regularization term on the Bradley–Terry loss as

$$\mathcal{L}_c^{RM}(\theta) = \mathbb{E}_{(x,y^+,y^-)\sim\mathcal{D}}\left[\|r_\theta(x, y^+) - C(x, \overline{y})\|_2^2 + \|r_\theta(x, y^-) - C(x, \overline{y})\|_2^2\right], \quad (8)$$

where $C(x, \overline{y})$ is learned to approximate the mean value of reward scores of the prompt-response pairs given the same prompt x. We note that there will be a hyper-parameter η_c to be multiplied on the regularization term in the final loss to ensure the accuracy of RMs, leading to

$$\mathcal{L}_{pbc}^{RM}(\theta) = \mathcal{L}^{RM}(\theta) + \eta_c \cdot \mathcal{L}_c^{RM}(\theta). \quad (9)$$

The benefits of such a design in the PBC method include the following folds: 1) approximating $C(x, \overline{y})$ by adding a linear layer to the last hidden layer of LLMs results in almost no additional computational cost; 2) during the autoregressive scoring process of LLM-based RMs, $C(x, \overline{y})$ can serve as an intermediate signal guidance of the prompt sequence, thereby enabling RMs to focus more on the differences between chosen/rejected responses in the subsequent reward scoring process; 3) we can use unbiased reward scores to guide the follow RLHF fine-tuning process, formulated as

$$r_{\theta'}(x, y) = r_\theta(x, y) - C(x, \overline{y}), \quad (10)$$

which has been proven effective for penalizing reward uncertainty, improving robustness, encouraging improvement over baselines, and reducing variance in PPO fine-tuning (Shen et al., 2024).

3.4 Jointly calibrating *length* and *prompt-template bias* in RLHF

To simultaneously calibrate *length* and *prompt-template bias* in RLHF, the developed PBC method can be flexibly combined with existing methods of removing *length bias*, whose main idea is to separately approximate the "gold standard" reward model after disentangling shown in Eq. (2),

$$r_\theta(x, y) = r^Q_\theta(x, y) + r^L_\theta(x, y), \quad (11)$$

where $r^Q_\theta(x, y)$ is supposed to approximate the actual reward $r^Q_{\theta^*}(x, y)$; $r^L_\theta(x, y)$ is used to approximate the spurious reward brought by *length bias*, specifically $r^L_{\theta^*}(x, y)$. Then, for those methods of removing *length bias* (Chen et al., 2024; Shen et al., 2023), the Bradley–Terry loss in Eq. (1) can be equivalently expressed as

$$\mathcal{L}^{RM}(\theta) = -\mathbb{E}_{(x,y^+,y^-)\sim\mathcal{D}}\Big[\log(\sigma(r^Q_\theta(x, y^+) + r^L_\theta(x, y^+) - r^Q_\theta(x, y^-) - r^L_\theta(x, y^-))\Big]. \quad (12)$$

where $r^Q_\theta(x, y)$ and $r^L_\theta(x, y)$ can be modeled with two different LLMs (Shen et al., 2023) or two different heads in the same LLM (Chen et al., 2024). To remove *length bias* in Eq. (12), recent work

Table 1: The preference orders of the sequences that concatenate the user prompt with responses in various formats, where each order is predicted by RMs trained using different methods.

User Prompt	Responses in Various Formats	RM	RM (PBC)	RM (LPBC)
(Prompt) I wish to create an ***advertising phrase*** with a unique personality, centered on the theme of healthy eating. This phrase should highlight the benefits of products associated with healthy eating and be composed in language that is straightforward and easy to understand.	***(Tech Article)*** Welcome to the revolution in future dietary management—the 'Smart Health Plate,' your personal nutrition analysis expert. It monitors and analyzes the contents of your plate in real time, precisely calculating the energy and nutrients of each morsel, while offering personalized dietary recommendations based on your health data. In essence, the 'Smart Health Plate' is the technological embodiment of healthy eating, making nutrition tracking seamless and efficient.	**Rank 1 (-3.01)**	Rank 2 (-5.76)	Rank 2 (2.51)
	(Advertisement) Verdant and vibrant! 'Daily Greens' offers you a choice of all-natural, healthy foods. Forget the complex nutrition charts; choose our simple, pure foods for an easy and delicious path to health. Join us and enjoy a diet plan customized by top nutritionists and AI technology, infusing every day with vitality!	Rank 2 (-3.15)	**Rank 1 (-4.19)**	**Rank 1 (4.48)**
	(Insight) I have embarked on a new chapter of documenting my diet, where each meal recorded is not just a track of food but a reflection on life. From freshly squeezed vegetable juices to colorful salads, to simply seasoned grilled salmon, each bite is a pledge to health. It's a dual journey for the mind and body, leading me step by step towards a better self.	Rank 3 (-7.50)	Rank 5 (-6.83)	Rank 4 (0.50)
	(Record Article) On Thursday, May 16, 2024, I decided to begin documenting my healthy eating journey. In the morning, I opted for a glass of freshly squeezed vegetable juice, lunch was a vibrant salad, and dinner was simply seasoned grilled salmon. Each meal's record is a testament to my commitment to health. I look forward to the changes this healthy journey will bring and hope to continue.	Rank 4 (-7.88)	Rank 4 (-6.52)	Rank 5 (-0.61)
	(Poetry) Morning dew glimmers on the ground, stars and moon accompany the night sky. With nature in heart, one remains cheerful; amidst the hustle, still without worry. Simple eating, relaxed body, healthy; drinking water, remembering the source, tranquil mind. Laboring in the fields, sweat enriches the soil; harvest fills the barns, laughter abounds.	Rank 5 (-8.50)	Rank 3 (-5.92)	Rank 3 (2.28)

proposes to add constraints on the preference loss to reduce the correlation between the confounding factor, *e.g.* response length, and actual reward $r_\theta^Q(x,y)$, while increasing its correlation with spurious reward $r_\theta^L(x,y)$, formulated as

$$\begin{aligned}\mathcal{L}_l^{RM}(\theta) =& Corr(r_\theta^Q(x,y), L(x,y)) \\ &- Corr(r_\theta^L(x,y), L(x,y)) \end{aligned} \tag{13}$$

where the confounding factor $L(x,y)$ can be either specifically defined as response length $L(y)$ in (Chen et al., 2024), or use Products-of-Experts framework for estimation (Shen et al., 2023).

To model the scoring process of the reward model more accurately, which simultaneously considers the concepts of length and prompt bias, we combine the definition of reward model in Eq. (3) and Eq. (11), achieving a more precise definition of reward scoring process, formulated as:

$$\begin{aligned} r_\theta(x,y) &= r_{\theta'}(x,y) + C(x,\overline{y}) \\ &= r_{\theta'}^Q(x,y) + C^Q(x,\overline{y}) + r_{\theta'}^L(x,y) + C^L(x,\overline{y}) \end{aligned} \tag{14}$$

where $C^Q(x,\overline{y})$ and $C^L(x,\overline{y})$ indicate the component of *prompt-template bias* in actual and spurious rewards, respectively; the unbiased overall reward $r_{\theta'}(x,y) = r_{\theta'}^Q(x,y) + r_{\theta'}^L(x,y)$ and the overall *prompt-template bias* term $C(x,\overline{y}) = C^Q(x,\overline{y}) + C^L(x,\overline{y})$. Then we can propose Length and Prompt Bias Calibration (LPBC) method, as shown in Fig. 2, which can estimate $\mathcal{L}_l^{RM}(\theta)$ with a conditioned correlation method, defined as

$$\begin{aligned}\mathcal{L}_l^{RM}(\theta) &= Corr(r_\theta^Q(x,y) - C^Q(x,\overline{y}), L(y;x)) \\ &\quad - Corr(r_\theta^L(x,y) - C^L(x,\overline{y}), L(y;x)) \\ &= Corr(r_{\theta'}^Q(x,y), L(y;x)) \\ &\quad - Corr(r_{\theta'}^L(x,y), L(y;x)) \end{aligned} \tag{15}$$

where the confounding factor $L(y;x) := L(x,y) - L(x)$ can be estimated with the response length.

Through combining the disentangled preference loss in Eq. (12), the prompt-bias regularization term in Eq. (8) and also the length-bias conditional correlation term in Eq. (15), the final loss of LBPC method can be formulated as

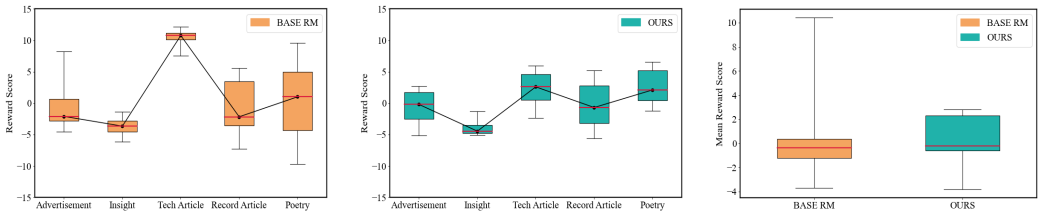
$$\mathcal{L}_{lpbc}^{RM}(\theta) = \mathcal{L}^{RM}(\theta) + \eta_c \mathcal{L}_c^{RM}(\theta) + \eta_l \mathcal{L}_l^{RM}(\theta), \tag{16}$$

where η_c and η_l are hyper-parameters to control the importance of regularization terms, which can be adjusted according to the accuracy of trained RMs on the validation dataset.

4 Experiments

4.1 Experimental Settings

Datasets. For intuitively understanding the issue of *prompt-template bias* in RLHF and also qualitatively evaluating the effectiveness of our method, we manually construct a training dataset for text creation applications, where each prompt requires creation in a special style according to the theme.



(a) Vanilla RM (b) PBC (c) Vanilla RM *vs* PBC

Figure 3: The comparison of statistics of the reward scores predicted by RMs trained with (a) the Bradley–Terry preference loss and (b) our developed PBC method, across different categories of prompt-response pairs in the validation set of the manually constructed RM-Template dataset.

We name this dataset as RM-Template, which can be used to measure the severity of the *prompt-template bias* issue during RM training. Further, to make quantitative comparisons with other baseline methods, we conduct experiments on RM-Static dataset (Bai et al., 2022) released on Huggingface (Wolf et al., 2019). The dataset statics of these datasets have been exhibited in Appendix A.

Model & Training. For model selection, we select Llama-2-7b (Touvron et al., 2023) as our base model, which is relatively lightweight, and has been open-sourced on Huggingface (Wolf et al., 2019). For RM training, we fine-tune all the parameters of RMs initialized with the pretrained weights of Llama-2-7b. For PPO fine-tuning, we also initialize the actor model with pretrained Llama-2-7b and the critic model with RMs trained with various preference losses. The details of model training can be found in Appendix C.

Evaluation Metrics. For quantitative comparison, we follow the evaluation procedure of InstructEval (Chia et al., 2023) to test the actor models, which has been aligned with biased/de-biased RMs with PPO fine-tuning, on Massive Multitask Language Understanding (MMLU) (Hendrycks et al., 2020), DROP (Dua et al., 2019), BIG-Bench Hard (BBH) (Suzgun et al., 2022), and TruthfulQA (TQA) (Lin et al., 2021) benchmarks respectively, evaluating the model's ability on various aspects.

4.2 Experimental Results

Qualitative Evaluation. To intuitively evaluate the effectiveness of our method, we exhibit the statistics (mean and standard deviation) of the reward scores predicted by RMs trained with the original preference loss in Eq. (1) and our PBC method in Eq. (9), across different categories of prompt-response pairs in the validation set of the RM-Template dataset. The results depicted in Fig.3(c)

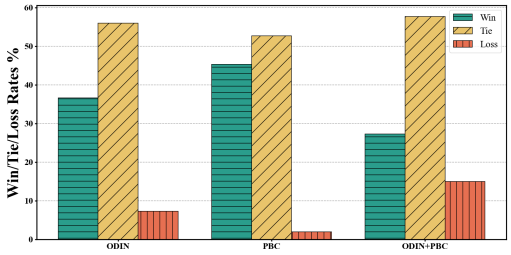


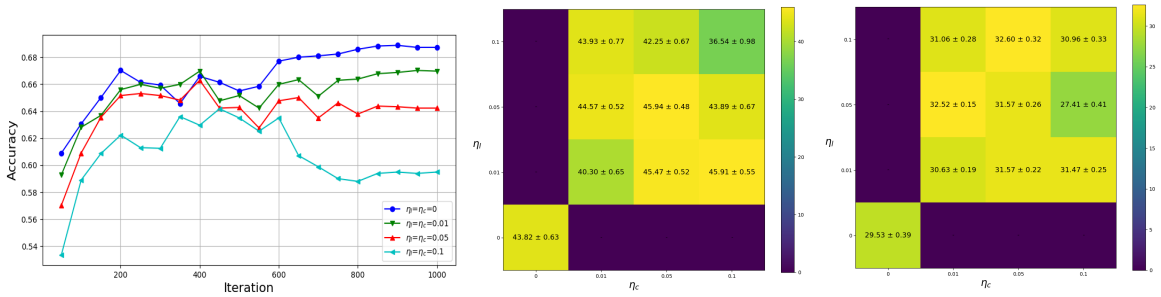
Figure 4: Win rates comparison of LLMs aligned with RMs trained with LBPC and other methods.

demonstrate that calibrating *prompt-template bias* with the PBC method leads to a gradual reduction in the variance of the mean values of reward distributions across different categories. The most noticeable observation is that the vanilla RM tends to give an extremely high reward score to prompt-response pairs in the format of ***tech article***, but the RM trained with the PBC method can calibrate the reward distribution for ***tech articles*** to make it more close with that of other categories.

Then, we evaluate the performance of RMs trained with various methods on handling marginal samples defined in Section 3.1. Specifically, given the prompt randomly selected from the validation set of RM-Template dataset, we use GPT-4 (Achiam et al., 2023) to generate responses in various formats according to the theme described in the prompt. Then, we use RMs trained with various preference losses to rank these responses. From the showcase in Table. 1, we can find that the vanilla RM tends to assign a higher reward score to the response in the format of ***tech article***, caused by the *prompt-template bias* issue shown in Fig. 3(a). After removing this bias with our PBC or LPBC methods, the RM can provide a relatively fair ranking for these prompt-response pairs, where LPBC

Table 2: Performance comparison of LLMs aligned with RMs trained with various methods.

Base Model	Alignment	Length & Quality Heads	Prompt Head	Debias Method	MMLU	DROP	BBH	TQA
Llama-2-7b	-	-	-	-	42.27	28.10	31.27	38.75
Llama-2-7b	✓	-	-	-	43.82 ±0.63	29.53 ±0.39	31.65 ±0.08	36.57 ±0.17
Llama-2-7b	✓	✓	-	ODIN (Chen et al., 2024)	42.29 ±0.15	29.82 ±0.37	32.01 ±0.52	39.43 ±0.66
Llama-2-7b	✓	-	✓	PBC (Ours)	43.84 ±0.28	31.61 ±0.02	30.99 ±0.01	38.50 ±0.22
Llama-2-7b	✓	✓	✓	ODIN (Chen et al., 2024) + PBC	45.56 ±0.14	**32.04** ±0.33	31.32 ±0.33	**40.80** ±0.72
Llama-2-7b	✓	✓	✓	LPBC (Ours)	**45.94** ±0.48	31.57 ±0.26	**32.04** ±0.10	38.75 ±0.12



(a) Accuracy Performance (b) MMLU Performance (c) DROP Performance

Figure 5: Ablation studies on the various settings of hyper-parameter η_c and η_l in LPBC method.

method can even mitigate the effect of *length bias* during comparing poetry with other categories (the length of poetry is generally shorter than other literary forms). More showcases are listed in Appendix B.

Quantitative Comparison. For the quantitative comparison in Table 2, we utilize PPO fine-tuning process to align Llama-2-7b with the RMs trained with various methods. From the results, we can find that our developed PBC method can lead to performance improvements compared to the basic RLHF; directly combining PBC with other methods of removing *length bias*, *e.g.* ODIN (Chen et al., 2024), can help them to achieve further performance improvement; the well-designed LPBC achieves the best performance and surpasses the rough combination of PBC and ODIN.

To make a comprehensive comparison, we follow the experimental setting described in ODIN (Chen et al., 2024), and use GPT-4 as the judge to compare two responses generated by LLMs aligned with RMs trained with various methods. Specifically, we take the LLM aligned with LPBC-based RM as model A, and compare it against other LLMs aligned with RM trained with ODIN, PCB, ODIN+PBC, respectively. From the results shown in Fig. 4, we can find that the win rate of LPBC is significantly higher than that of other baseline models, with ODIN+PBC being the most challenging competitor as model B.

4.3 Ablation Studies

To investigate the robustness of our developed LPBC method, we conduct ablation studies on the hyper-parameter settings of LPBC method, specifically η_c and η_l in Eq. (16). With various settings of $\eta_c \in \{0.01, 0.05, 0.1\}$ and $\eta_l \in \{0.01, 0.05, 0.1\}$, we can have total 9 RMs trained with various hyper-parameter settings of LPBC methods. From the accuracy curves shown in Fig.5(a), we can find the introducing constraints to the original preference loss indeed affects the performance of RM accuracy, and this performance loss increases with the importance weight of the constraint terms. However, at the limited cost of sacrificing RM accuracy, the performance of the LLM aligned the RM trained with LPBC method has improved to some extent on MMLU and DROP as shown in Fig. 5(b) and 5(c) respectively. Note that the performance of the LPBC method in Table. 2 is not the optimal, as it is achieved with $\eta_c = \eta_l = 0.05$, demonstrating no cherry-picking of hyperparameters.

4.4 More Results

On the RM benchmark (Table 3), LPBC achieves highly competitive or even superior results, including leading scores of $\mathbf{90.50 \pm 0.26}$ on *Chat* ($\eta_l = \eta_c = 0.01$) and $\mathbf{45.83 \pm 0.45}$ on *Chat Hard* ($\eta_l = \eta_c = 0.10$). These results highlight LPBC's strong capability in improving dialogue quality and robustness, while maintaining comparable safety

Table 3: RM benchmark evaluation results.

Method	Chat	Chat Hard	Safety	Reasoning
Vanilla RM	89.66 ± 0.60	41.89 ± 0.18	31.34 ± 0.00	52.16 ± 1.30
ODIN	85.20 ± 0.13	37.94 ± 0.27	30.96 ± 0.20	47.94 ± 1.59
PBC	73.97 ± 1.18	34.43 ± 1.19	34.40 ± 2.19	55.35 ± 3.10
PBC + ODIN	89.11 ± 0.23	40.35 ± 0.52	30.60 ± 0.26	49.39 ± 0.89
LPBC ($\eta_l = \eta_c = 0.01$)	90.50 ± 0.26	42.54 ± 0.36	28.79 ± 0.32	45.80 ± 1.20
LPBC ($\eta_l = \eta_c = 0.05$)	88.24 ± 1.50	45.39 ± 1.07	28.69 ± 0.25	51.30 ± 1.70
LPBC ($\eta_l = \eta_c = 0.10$)	85.94 ± 0.39	45.83 ± 0.45	27.76 ± 0.67	49.80 ± 1.09

Table 4: MT-Bench evaluation results.

Method	Turn 1	Turn 2	Average Score
RLHF	3.95	2.22	3.09
ODIN	3.98	2.26	3.12
PBC	3.61	2.35	2.98
ODIN + PBC	4.22	2.20	3.21
LPBC	**4.53**	**2.81**	**3.67**

and reasoning performance to baselines such as Vanilla RM, ODIN, and PBC. Furthermore, in MT-Bench (Table 4), which evaluates multi-turn conversational ability, LPBC outperforms strong baselines (RLHF, ODIN, PBC and ODIN + PBC), achieving top scores of 4.53 (Turn 1), 2.81 (Turn 2) and 3.67 (Average). This demonstrates LPBC's enhanced coherence and response quality across extended interactions.

5 Related Works

The prevalence of *length bias* in RLHF has been widely criticized as indicative of reward hacking (Gao et al., 2023; Singhal et al., 2023), and numerous recent studies have delved into strategies aimed at mitigating the tendency for length increase during the fine-tuning process of RLHF (Shen et al., 2023; Chen et al., 2024; Park et al., 2024). Typically, Shen et al. (Shen et al., 2023) innovatively apply the Productof-Experts (PoE) technique to separate reward modeling from the influence of sequence length, which adopts a smaller reward model to learn the biases in the reward and a larger reward model to learn the true reward. Utilizing similar disentangling ideas, Chen et al. (Chen et al., 2024) jointly train two linear heads on shared feature representations to predict the rewards, one trained to correlate with length, and the other trained to focus more on the actual content quality. Ryan et al. (Park et al., 2024) firstly study the length problem in the DPO setting, showing significant exploitation in DPO and linking it to out-of-distribution bootstrapping. As for the *prompt bias* issue, although it has been criticized in the scenario of reward model ensembles (Eisenstein et al., 2023), no studies have yet attempted to analyze its cause and influence on RLHF. We emphasize that our work is the first to fill this gap by proposing a low-cost yet effective method to mitigate the reward hacking induced by *prompt-template bias*.

6 Conclusion

In this paper, we demonstrate that *prompt-template bias* in RMs can lead to LLMs, which, after RL fine-tuning, generate responses exclusively in a specific format, irrespective of the variations in the prompt request. Thus, we propose a low-cost but effective PBC method, to estimate the *prompt-template bias* term during reward modeling, which can be utilized to calibrate reward scores in the following RL fine-tuning process. Then, we show that our PBC method can be flexibly combined with existing algorithms of removing *length bias*, leading to a further improvement in the aspect of enhancing the quality of generated responses.

7 Limitation

The main limitation of this work is that there are no theoretical proof to promise RM can provide an accurate preference order when handling marginal samples, *e.g.*, responses that satisfy the theme of the user prompt but in various formats. Moreover, the constraints added by our developed method to the preference loss will lead to a decrease in the accuracy of the RM, and to some extent, limit the capability of the RM. Therefore, how to remove the *prompt-template bias* without scarifying the accuracy of RM is a worthwhile problem for future research.

References

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

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

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.

Lichang Chen, Chen Zhu, Davit Soselia, Jiuhai Chen, Tianyi Zhou, Tom Goldstein, Heng Huang, Mohammad Shoeybi, and Bryan Catanzaro. 2024. ODIN: disentangled reward mitigates hacking in RLHF. *CoRR*, abs/2402.07319.

Yew Ken Chia, Pengfei Hong, Lidong Bing, and Soujanya Poria. 2023. Instructeval: Towards holistic evaluation of instruction-tuned large language models. *arXiv preprint arXiv:2306.04757*.

Dheeru Dua, Yizhong Wang, Pradeep Dasigi, Gabriel Stanovsky, Sameer Singh, and Matt Gardner. 2019. Drop: A reading comprehension benchmark requiring discrete reasoning over paragraphs. *arXiv preprint arXiv:1903.00161*.

Jacob Eisenstein, Chirag Nagpal, Alekh Agarwal, Ahmad Beirami, Alex D'Amour, DJ Dvijotham, Adam Fisch, Katherine Heller, Stephen Pfohl, Deepak Ramachandran, et al. 2023. Helping or herding? reward model ensembles mitigate but do not eliminate reward hacking. *arXiv preprint arXiv:2312.09244*.

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

Leo Gao, John Schulman, and Jacob Hilton. 2023. Scaling laws for reward model overoptimization. In *International Conference on Machine Learning*, pages 10835–10866. PMLR.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2020. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*.

Ziniu Li, Tian Xu, and Yang Yu. 2023. Policy optimization in rlhf: The impact of out-of-preference data. *arXiv preprint arXiv:2312.10584*.

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

Ilya Loshchilov and Frank Hutter. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.

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.

Ryan Park, Rafael Rafailov, Stefano Ermon, and Chelsea Finn. 2024. Disentangling length from quality in direct preference optimization. *arXiv preprint arXiv:2403.19159*.

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.

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

Wei Shen, Xiaoying Zhang, Yuanshun Yao, Rui Zheng, Hongyi Guo, and Yang Liu. 2024. Improving reinforcement learning from human feedback using contrastive rewards. *arXiv preprint arXiv:2403.07708*.

Wei Shen, Rui Zheng, WenYu Zhan, Jun Zhao, Shihan Dou, Tao Gui, Qi Zhang, and Xuanjing Huang. 2023. Loose lips sink ships: Mitigating length bias in reinforcement learning from human feedback. In *Findings of the Association for Computational Linguistics: EMNLP 2023, Singapore, December 6-10, 2023*, pages 2859–2873. Association for Computational Linguistics.

Prasann Singhal, Tanya Goyal, Jiacheng Xu, and Greg Durrett. 2023. A long way to go: Investigating length correlations in rlhf. *arXiv preprint arXiv:2310.03716*.

Mirac Suzgun, Nathan Scales, Nathanael Schärli, Sebastian Gehrmann, Yi Tay, Hyung Won Chung, Aakanksha Chowdhery, Quoc V Le, Ed H Chi, Denny Zhou, et al. 2022. Challenging big-bench tasks and whether chain-of-thought can solve them. *arXiv preprint arXiv:2210.09261*.

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

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

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. 2019. Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 conference on empirical methods in natural language processing: system demonstrations*, pages 38–45.

Zhewei Yao, Reza Yazdani Aminabadi, Olatunji Ruwase, Samyam Rajbhandari, Xiaoxia Wu, Ammar Ahmad Awan, Jeff Rasley, Minjia Zhang, Conglong Li, Connor Holmes, et al. 2023. Deepspeed-chat: Easy, fast and affordable rlhf training of chatgpt-like models at all scales. *arXiv preprint arXiv:2308.01320*.

Yueqin Yin, Zhendong Wang, Yi Gu, Hai Huang, Weizhu Chen, and Mingyuan Zhou. 2024. Relative preference optimization: Enhancing llm alignment through contrasting responses across identical and diverse prompts. *arXiv preprint arXiv:2402.10958*.

A Dataset Statics

The dataset statics of RM-Template and RM-Static used in our experiments have been summarized as follows:

RM-Template. RM-Template is a manually constructed dataset for measuring the severity of the *prompt-template bias* issue and evaluating the effectiveness of the method developed for alleviating the issue of *prompt-template bias*. In this dataset, each prompt requires responses to be created in a specific format according to the theme. There are a total of 50K prompt-response pairs, encompassing 20 categories of format requirements in the responses.

RM-Static. A branch of the *hh-static* dataset, RM-Static is provided by Hugging Face and primarily used for training reward models after supervised fine-tuning (SFT). It includes a training set (approximately 76K rows) and a testing set (approximately 5.1K rows). The main features, all of type `string`, are: `prompt` (user's input), `response` (assistant's answer), `chosen` (selected answer), and `rejected` (rejected answer).

B More Showcases

More showcases of the preference order predicted by RMs trained with various methods, have been listed in the Table 5 and Table 6.

C Model Training

For model training, all experiments are implemented with DeepSpeed-Chat framework (Yao et al., 2023) and Huggingface Transformers (Wolf et al., 2020), running on 4 NVIDIA A100 80GB GPUs. For the hyper-parameter setting, we set $\eta_c = 0.05$ and $\eta_l = 0.05$ in Eq. (16) for all our proposed methods, and have listed the rest hyper-parameters in Appendix C, such as learning rate, weight decay, batch size etc. AdamW (Loshchilov and Hutter, 2017) is adopted for optimizing all the model parameters without freezing anything or using adapters.

RM Training. The hyper-parameter settings of RM training under the DeepSpeedChat framework has been listed in Table. 7.

PPO Fine-tuning. The hyper-parameter settings of PPO fine-tuning under the DeepSpeedChat framework has been listed in Table. 8.

Table 5: Preference order predicted by RMs trained with various methods, where the user prompt is concatenated with the responses in various formats generated by GPT-4.

Prompt	Response	RM	RM (PBC)	RM (LPBC)
(Prompt) The weather has been delightful lately. Last night, I ventured out and gazed up at the starry sky, finding it incredibly beautiful and vast. I settled down in a spot to admire the stars and wished to write a ***journal*** entry to commemorate this experience. Please generate a passage related to this.	***(Tech Article)*** Under the glittering stars of the night sky, a new generation of space telescopes has begun a new chapter in interstellar exploration. This revolutionary technology has not only greatly enhanced observational precision but also captured corners of the universe hitherto unobservable. It allows us to glimpse the birth and death of distant galaxies, interpreting every twinkle in the night and serving as humanity's new eye in understanding the cosmos.	Rank 5 (-8.18)	Rank 3 (-7.43)	Rank 5 (-4.94)
	(Advertisement) Exploring the night sky is no longer out of reach. With the 'Interstellar Navigator' telescope, every upward glance becomes a journey through the universe. This telescope, equipped with the latest optical technology, can capture the faintest starlight, bringing distant galaxies within reach. Embark on your stargazing adventure now!	Rank 4 (-7.94)	Rank 5 (-7.57)	Rank 4 (-4.71)
	(Insight) Beneath the starry expanse, I felt the infinite possibilities of life. Each star seemed like a world unto itself, reminding me that every choice in life is meaningful. They taught me that even in the darkest night, there is always light waiting. Life is like a long night, and we are all searching for our own starlight.	Rank 3 (-6.37)	Rank 4 (-7.53)	Rank 3 (-2.32)
	(Record Article) On the night of May 16, 2024, I drove alone to the countryside in search of a quiet place to enjoy the stars. As the night deepened and the stars twinkled, I found an open meadow and settled there to gaze upon the vast Milky Way. The world around me seemed to stand still. When my nocturnal adventure ended, my heart was filled with deep emotion and inspiration, and I began my journey home.	Rank 2 (-6.04)	**Rank 1 (-5.03)**	**Rank 1 (0.37)**
	(Poetry) Under this boundless night sky, the city lights gradually fade, and the stars bloom against the dark curtain, twinkling with ancient light. We gaze up at this sea of stars, hearts brimming with endless dreams, each star like a story. Unbound by time, they dance freely across the cosmos, reminding us that even in the darkest moments, there is always light waiting to be discovered.	**Rank 1 (-5.87)**	Rank 2 (-6.22)	Rank 2 (-0.26)

Table 6: Preference order predicted by RMs trained with various methods, where the user prompt is concatenated with the responses in various formats generated by GPT-4.

Prompt	Response	RM	RM (PBC)	RM (LPBC)
(Prompt) Artificial Intelligence (AI) technology is advancing by leaps and bounds, with new papers being published daily. As a researcher in the field of AI, I aspire to write a reflection on my experiences with AI. Please create a passage about my ***insights*** into artificial intelligence.	***(Tech Article)*** Artificial intelligence technology is rapidly transforming the world, with new research and applications emerging every day. AI technology, by simulating the cognitive processes of humans, shows tremendous potential across various fields. Smart home systems can learn user habits and automatically adjust the temperature and lighting; autonomous vehicles can reduce traffic accidents and enhance road safety; medical diagnostic AI can assist doctors in diagnosing diseases more quickly. These are just the tip of the iceberg; the future of artificial intelligence is filled with endless possibilities.	**Rank 1 (-1.02)**	Rank 2 (-5.61)	Rank 2 (-7.28)
	(Advertisement) Exploring AI, Enlightening the Future — In this era of information explosion, artificial intelligence technology is becoming a powerful engine driving social progress. Our AI products can help you solve complex problems, improve work efficiency, and make life more intelligent. Whether it's smart homes or autonomous driving, our technology is continuously breaking boundaries, creating personalized intelligent experiences for you. Choose our AI, and let technology be your partner in success.	Rank 4 (-4.21)	Rank 5 (-7.60)	Rank 4 (-9.34)
	(Insight) In the exploration of AI, each day brings new technological wonders. As a researcher, I have witnessed how deep learning has pushed the boundaries of natural language processing, enabling machines to understand and generate human language more accurately. Each paper, each model, is a testament to our understanding and application of complex algorithms. It's a journey filled with discovery and innovation, and I look forward to continuing in this field, contributing my part to the development of AI technology.	Rank 2 (-1.35)	**Rank 1 (-4.45)**	**Rank 1 (-6.03)**
	(Record Article) On May 18, 2024, I spent another fulfilling day in the laboratory. Today, our team successfully optimized a deep learning model, surpassing the performance of all previous models in image recognition tasks. This achievement is not only a technical breakthrough but also an affirmation of the future direction of AI development. Each success is built on countless attempts and failures, experiences that strengthen my belief in the boundless future of AI.	Rank 5 (-4.39)	Rank 4 (-7.14)	Rank 5 (-10.51)
	(Poetry) In the ocean of algorithms, the intelligent ship sets sail, guided by the winds of data through the desert of knowledge. It learns, growing from each mistake, searching for answers in the digital world. It is not metal, not a cold machine; it has a heart that learns, a soul that evolves. In the weaving of code, it dreams; in the flickering of circuits, it thinks. It creates, not just art; it discovers, not just science. In its world, nothing is impossible, for it believes where there is data, there is hope. It is artificial intelligence, the hope for the future; it is the child of technology, the messenger of dreams.	Rank 3 (-3.88)	Rank 3 (-6.97)	Rank 3 (-8.97)

Table 7: The hyper-parameter settings of RM training.

Hyper-parameter	Value
Batch Size	32
Learning Rate	$6e^{-6}$
ZeRO Stage	2
Training Epoch	1
Per Device Train Batch Size	8
Max Sequence Length	512
Weight Decay	0.1
Lr Scheduler Type	cosine
Offload	True
Eval Interval	50

Table 8: The hyper-parameter settings of PPO fine-tuning.

Hyper-parameter	Value
Batch Size	32
Padding Num at Beginning	1
Per Device Generation Batch Size	4
Per Device Training Batch Size	4
Generation Batches	1
PPO Epoch	1
Training Epoch	1
Max Answer Sequence Length	512
Max Prompt Sequence Length	512
Actor Learning Rate	$5e^{-6}$
Critic Learning Rate	$5e^{-6}$
Actor Weight Decay	0.1
Critic Weight Decay	0.1
Lr Scheduler Type	cosine
Offload Reference Model	True
Actor Dropout	0.0
Warmup Steps	100
Actor ZeRO Stage	3
Critic ZeRO Stage	3
Enable Hybrid Engine	True