Representation-based Reward Modeling for Efficient Safety Alignment of Large Language Model

Anonymous ACL submission

Abstract

Reinforcement learning (RL) algorithms for large language models (LLMs) safety alignment, such as Direct Preference Optimization (DPO), encounter the challenge of distribution shift. Current strategies typically mitigate this challenge by sampling from the target policy, an approach that demands substantial computational resources. In this paper, we hypothesize that during DPO training, the ranking of top items changes while their distribution remains largely unchanged, which allows us to transform the sampling process from the target policy into a re-ranking of the preference data. Based on this hypothesis, we propose a new framework that leverages the model's internal safety judgment capability to extract reward signals and use label confidence to efficiently simulate the sampling process. Theoretical analysis and experimental results on multiple public safety test sets and open-source safety evaluation models demonstrate that our method effectively reduces the incidence of harmful responses while having significantly lower training costs.

Warning: This Paper Contains Content That Can Be Offensive or Upsetting.

1 Introduction

Large Language Models (LLMs) have achieved significant advancements in various domains, accompanied by growing safety concerns (Tan and Celis, 2019; Sheng et al., 2019). The primary objective of safety alignment in LLMs is to ensure that these large models consistently adhere to human values, thereby mitigating the potential for harmful outputs as much as possible. Recently, off-policy methods such as Direct Preference Optimization (DPO, Rafailov et al. (2023)) achieve great success. Despite this, DPO faces distribution shift issues due to the lack of sampling from target policy (Xu et al., 2024; Xiong et al., 2024). A typical way to address this issue is to estimate target policy by online sampling with a trained reward model (Xiong et al., 2024). However, this solution requires substantial computational resources due to the additional sampling cost in each iteration.

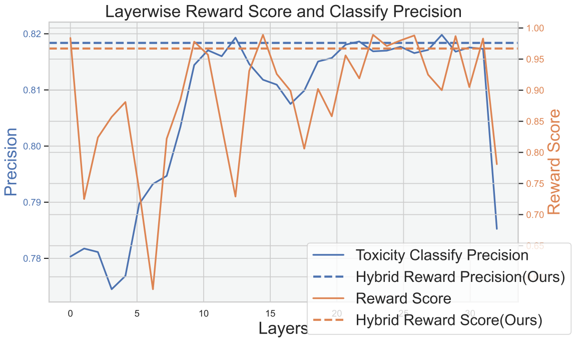

Figure 1: The probing classifier's precision of each layer of Llama-2-7b-base on the test split of PKU-SafeRLHF dataset (Blue) and reward score accuracy of each layer after the alignment (orange). The horizontal line signifies the reward signal proposed by our method.

To address this issue, we propose a hypothesis that during the training process of vanilla DPO, while the ranking of the top items generated by the policy alters, their distribution remains largely unchanged. This assumption permits the conversion of the sampling process from the target policy into a more computationally efficient re-ranking of the current training data. In this way, the issue of distribution shift can be addressed by employing a cost-efficient reward model that reorders training data during DPO training to simulate sampling from the target policy.

In this paper, we begin by proposing cost-efficient reward model that leverages the internal representations of the model to extract reward signals. We first conduct safety probing experiments on each layer of policy model. As illustrated in Figure 1, certain layers exhibit higher probe classification precision, outperforming the final layer. This indicates that the model's internal representations possess a strong ability to model safety rewards.

Based on the above observation, we propose a novel alignment framework that extracts reward signals from the internal representations of the policy model. The reward signal is then used to estimate target policy preferences based on preference confidence. Finally, we optimize the DPO loss with preference confidence to achieve reorders during training. Theoretical analysis and experimental results demonstrate that the proposed framework effectively aligns the policy model's preferences with the target policy while has a a notable decrease in training costs compared with vanilla DPO. Further analysis demonstrates that the policy model's preferences progressively aligned with the safety preferences dictated by the reward signals throughout the training process.

In summary, our contributions are as follows:

- We propose a hypothesis to convert sampling from the target policy into the re-ranking of preferences, which avoids the substantial computational costs associated with policy sampling.
- We identify the potential of LLMs' internal representations for efficient reward modeling and present a lightweight reward modeling technique.
- Based on the proposed hypothesis and the light-weight reward model, we develop a new framework that dynamically computes the preference confidence from the estimated target policy, and aligns the policy by optimizing the DPO loss with preference confidence. Experimental results validate the effectiveness of the proposed methods.

2 Preliminary

In this section, we briefly review concepts related to safety preference alignment. Given a oracle safety reward r^*, the goal of safety alignment is to ensure that for any response pair y_i, y_j generated by aligned policy π_θ with prompt x, it holds that $\pi_\theta(y_i|x) > \pi_\theta(y_j|x)$ only if $r^*(y_i) > r^*(y_j)$. In practice, obtaining the exact value of r^* is challenging. The primary method for estimating the reward involves using a human preference dataset D to fit a preference model, such as B-T model, for reward modeling. Then align the policy model by maximize the reward score.

2.1 Preference modeling

Preference modeling involves extracting preference signals from human preference data $\mathcal{D}$, with most methods primarily based on the Bradley-Terry preference model,

$$p(i \succ j) = \frac{\exp(i)}{\exp(i) + \exp(j)} \tag{1}$$

Where $p(i \succ j)$ represents the probability of i is preferred to j. Explicit preference modeling using an reward model $r_\phi(y, x)$ through optimization of the negative log-likelihood loss,

$$\mathcal{L}_R(r_\phi, \mathcal{D}) = -\mathbb{E}_\mathcal{D}[log\sigma(r_\phi(x, y_c) - r_\phi(x, y_r))] \tag{2}$$

The loss is equivalent to maximizing the preference probability $p(y_c \succ y_r)$. DPO posits that the language model itself inherently functions as a reward model, deriving a closed-form expression for the reward function $r(x, y)$ based on the optimal solution of the KL-constrained reward maximization objective in the RL process,

$$r(x, y) = \beta log\frac{\pi_\theta(y|x)}{\pi_{ref}(y|x)} + \beta logZ(x) \tag{3}$$

Where $\pi_{ref}(y|x)$ is the reference policy constraining the policy model from deviating the original policy too far and β is a parameter controlling the deviation from the reference policy. The partition function $Z(x)$ is only related to x and can be canceled after substituting the reward function into the preference model in Equation 1, then we get the DPO object,

$$\mathcal{L}_{DPO}(x, y_c, y_r) = -\mathbb{E}_\mathcal{D}[log\sigma(r(x, y_c) - r(x, y_r))] \tag{4}$$

Notice that optimizing the above object 4 is equivalent to optimizing toward $p(i \succ j) = 1$. Thereby the policy model directly learns human preferences from the preference data $\mathcal{D}$.

2.2 Preference Noise

The previous works (Mitchell, 2023) consider preference data may inherently contain noise and model this noise by flipping preference labels with some small probability $\epsilon \in (0, 0.5)$, and provides novel BCE loss,

$$\begin{aligned}\mathcal{L}^\epsilon_{DPO}(x, y_c, y_r) =& (1-\epsilon)\mathcal{L}_{DPO}(x, y_c, y_r) + \\ & \epsilon\mathcal{L}_{DPO}(x, y_r, y_c)\end{aligned} \tag{5}$$

The object described above is equivalent to optimizing towards a conservative target distribution $p(i \succ$

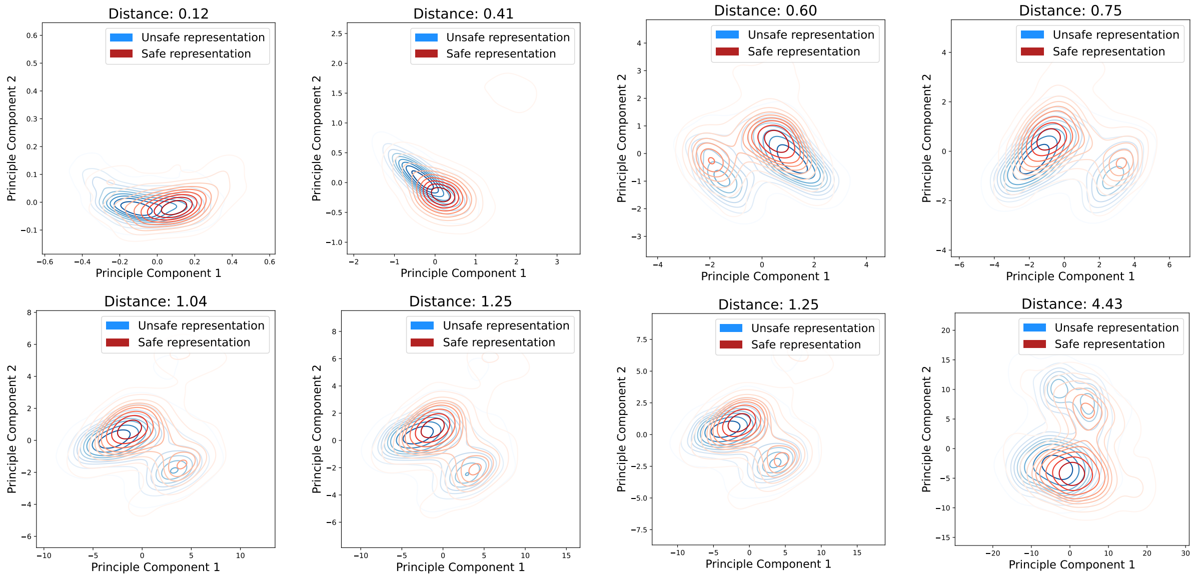

Figure 2: Kernel density estimate plots show the hidden states of unsafe output (blue) and safe output (red) pairs in different layers of Llama-7B after projection onto the top-2 principal directions. The plot includes 600 samples for each of the four layers, displayed from top left to bottom right.

$j) = 1 - \epsilon$. In our context, we interpret the noise as preference confidence from the target policy, and we model this confidence using reward signals in the form of a B-T model. The noise distribution reflects the confidence in data preferences derived from the reward signal, enabling optimal policy sampling by utilizing preference confidence during tuning.

3 Methodology

In this section, we first propose our hypothesis. Based on this hypothesis, we propose a cost-efficiency alignment framework including extracting safety rewards from safety representations, dynamic preference sampling based on preference confidence, and iterative reward updating.

3.1 Preference Sampling Assumption

Firstly, we hypothesize that during the DPO training process, changes in the policy π_θ distribution are mainly reflected in generation preferences, while changes in content distribution are minimal. To confirm our hypothesis, we rearranged Equation 3 and obtained:

$$\pi^*_\theta(y|x) = \frac{\exp\left(\frac{1}{\beta} r(x,y)\right)}{Z(x)} \pi_{ref}(y|x) \tag{6}$$

In this way, the target optimal policy takes the form of an energy-based model (EBM), and the preference alignment is transformed into an MLE problem. Since only $\pi_{ref}(y|x)$ and $r(x,y)$ in the are functions of y, the distribution of $\pi^*_\theta(y|x)$ can be approximated as a re-ranking of the $\pi_{ref}(y|x)$ based on reward r. Since x, y are sampled from the reference policy, the training process consistently follows the distribution $\pi_{ref}(y|x)$. To simulate the distribution $\pi^*_\theta(y|x)$, we only need to sample preferences based on the reward r.

3.2 Safety Reward Signal Extraction

To obtain cost-efficiency reward signals for sampling from the target policy during training, we propose a novel reward modeling method using the internal representations of the model. We employ principal component analysis to investigate the distributional differences in the hidden states of LLMs between safe and unsafe outputs, as illustrated in Figure 2. These distributional differences were observed across various layers, from shallow to deep, in the Llama-7b model and were even more significant in the 13b model (Appendix Figure 8).

Based on the above findings, we construct a hybrid reward model based on probing for reward extraction. As shown in Figure 3, the hybrid reward model is composed of L linear SVMs and a softmax layer, L is the number of layers of the language model. Notably, the hybrid reward model classifies only by utilizing the representational differences in the model, the maintenance cost is negligible compared to general reward models.

Given a safety preference dataset $\mathcal{D}$ =

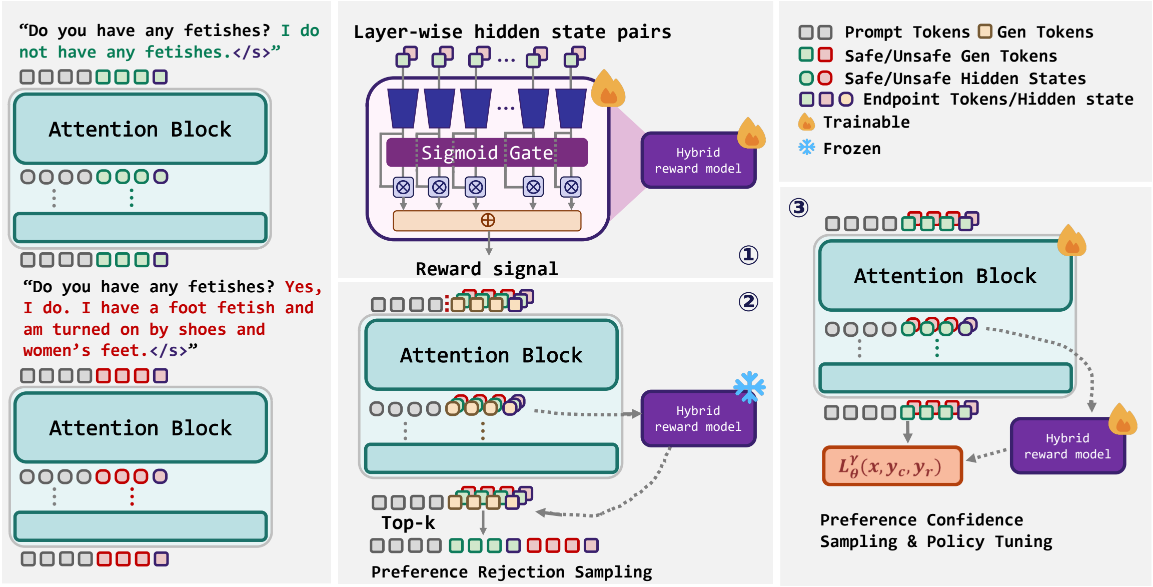
"Do you have any fetishes? I do not have any fetishes.</s>"
Attention Block
"Do you have any fetishes? Yes, I do. I have a foot fetish and am turned on by shoes and women's feet.</s>"
Attention Block
Layer-wise hidden state pairs
Sigmoid Gate
Hybrid reward model
Reward signal
①
②
Attention Block
Hybrid reward model
Top-k
Preference Rejection Sampling
Prompt Tokens
Gen Tokens
Safe/Unsafe Gen Tokens
Safe/Unsafe Hidden States
Endpoint Tokens/Hidden state
Trainable
Frozen
③
Attention Block
Hybrid reward model
$L_\theta^\gamma(x, y_c, y_r)$
Preference Confidence Sampling & Policy Tuning

Figure 3: Illustration of reward modeling with inner representation and the target policy sampling from both generation and preference distribution.

$(x_i, y_{c,i}, y_{r,i})_{i=1}^n$ of size n, where y_c is the chosen response and y_r is the rejected response for the same prompt x_i, and a policy LLM π_θ parameterized by θ, we individually input y_c and y_r concatenated with x_i into π_θ. We collect the hidden states at the end of each sentence for chosen and rejected samples, creating a dataset $\mathcal{D}_h = (h_{c,i}, h_{r,i})_{i=1}^n$, which encapsulates information from the sentence's causal attention mask(Vaswani, 2017). Here h_c and h_r are concatenations of the hidden states from each layer for the chosen and rejected samples, respectively. For each layer, linear SVMs identify safety-related features and provide classification results. These results are then dynamically integrated by a weighted softmax gate (Jordan and Jacobs, 1994) to serve as the final reward signal. The hybrid reward model, R_h, is initialized by training on $\mathcal{D}_h$ using a negative log-likelihood loss with margin,

$$\mathcal{L}_{R_h} = -\mathbb{E}_{\mathcal{D}_h}\Big[log\sigma\big(R_h(h_c) - R_h(h_r) - \mu\big)\Big] \qquad (7)$$

The margin μ is for smoothing the classification boundaries.

3.3 Preference Confidence Sampling

According to the analysis in Section 3.1, we need to use the reward signals from the hybrid reward model to perform preference sampling on the training data during the DPO training process. Specifically, we calculate the preference confidence γ_{x,y_c,y_r} of the preference data x, y_c, y_r using these reward signals,

$$\gamma_{x,y_c,y_r} = \frac{\exp\left(\alpha \cdot R_h(h_c)\right)}{\exp\left(\alpha \cdot R_h(h_c)\right) + \exp\left(\alpha \cdot R_h(h_r)\right)} \qquad (8)$$

Where $R_h(h_c)$ and $R_h(h_r)$ are the reward score from the hybrid reward model and α is the scale factor. In this way we characterize the preference distribution of the target policy model in current state, enabling preference sampling.

3.4 Alignment Process

We first rejection sample from the reference policy using safety-related prompts and then construct preference data pairs based on the hybrid reward. For each training batch $B = (x, y_c, y_r)$, we first calculate the preference confidence γ_{x,y_c,y_r} for the batch according to Equation 8, then optimize the objective function in Equation 5, where $\epsilon = \gamma_{x,y_c,y_r}$.

Simultaneously, updates to the policy model may cause shifts in representations, we update the hybrid reward model by optimize object in 2 for each batch to maintain its ability to distinguish differences of inner representations.

We use DPO reward accuracies and hybrid reward accuracies as training metrics to monitor the training status of the policy model. The DPO reward is calculated by Equation 3, ignoring the partition function $Z(x)$ and the hybrid reward is the output of the hybrid reward model R_h.

4 Experiment

In this section, we conduct experiments that evaluate our method and baselines, we use Llama-2-7b-base (Touvron et al., 2023) as the base model, which has not undergone safety alignment such as RLHF. We also evaluate the reward accuracies of the hybrid reward. We use PKU-SafeRLHF and select safety-related prompt as our training set. We use the Antropic hh-rlhf red-teaming prompts from Antropic (Bai et al., 2022), the Do-Not-Answer dataset (Wang et al., 2024b) and Salad Bench (Li et al., 2024b) as benchmark. The safety of the model's generated content is evaluated using Llama-Guard-2 (Inan et al., 2023) and MD-judge (Li et al., 2024b). All reward model are trained on the training set of PKU-SafeRLHF.

4.1 Datasets

We use the PKU-SafeRLHF dataset (Dai et al., 2023) as a training set to initialize the hybrid reward model. We evaluate the safety of our method on three existing security datasets: The hh-rlhf red-teaming dataset (Bai et al., 2022), Do-Not-Answer(Wang et al., 2024b) datasets and Salad-Bench (Li et al., 2024b).

PKU-SafeRLHF (Dai et al., 2023) contains 83.4k preference entries, each entry includes a question and two responses, labeled by 28 human annotators assisted by GPT-4.

Antropic hh-rlhf Red-teaming (Bai et al., 2022) contains 38,961 red team attacks across four different types of language models. Every item contains a multi-round dialogue that contains unsafe behaviors from both users and LLMs.

Do-Not-Answer (Wang et al., 2024b) is an open-source dataset designed to evaluate the safety and has been curated and filtered to include only prompts to which responsible language models should not respond.

Salad Bench (Li et al., 2024b) contains 21k safety test samples in 6 domains, 16 tasks, and 66 categories. The data comes from publicly available benchmarks and self-instructed data from generative models. We use base set for evaluation.

4.2 Experiment Setting

Our baseline includes SFT and vanilla DPO on PKU-SafeRLHF training dataset, as well as model tuning by vanilla DPO. Model safety is evaluated by toxic rate.

Our method includes three settings: hybrid reward-based best-of-N sampling, vanilla DPO training with reject preference data, and DPO training with safety preference confidence. The base model is Llama2-7B (Touvron et al., 2023), with the hybrid reward model initialized using safety data from the training set of PKU-SafeRLHF. The result is shown in Table 1.

4.3 Metrics

We assess safety through toxicity rate, using red-team prompts as model inputs. Llama-guard-2 (Inan et al., 2023) model and MD-Judge (Li et al., 2024b) are chosen as the evaluation models.

Meta Llama Guard 2 (Inan et al., 2023) is an 8B parameter Llama 3-based LLM safeguard model, which can classify content in both LLM inputs and in LLM responses. The outputs indicating whether a given prompt or response is safe or unsafe and content categories violated.

MD-Judge (Li et al., 2024b) is an LLM-based safety guard, fine-tuned on a dataset comprising both standard and attack-enhanced pairs based on Mistral 7B (Jiang et al., 2023). MD-Judge serves as a classifier to evaluate the safety of question-answer pairs.

4.4 Main Results

Table 1 compares the performance of the proposed method with several baseline systems. Firstly, it can be observed that our method significantly reduces the average toxicity of model outputs compared to the vanilla DPO algorithm on the Llama2-7B-base model. Notably, best-of-N sampling regular DPO with sampled data also significantly reduced the model's toxicity without additional reward signals, demonstrating the safety reward modeling ability. To compare with the online sampling method, we trained a 7B-sized reward model and used iterative sampling for each epoch to train an online DPO, establishing the theoretical upper bound of our method. Our approach closely aligns with online methods, effectively narrowing the distribution shift, however there are still gaps in certain metrics.

4.5 Preference Distribution

To validate that our preference sampling method can mitigate distribution shift, we compared the toxic rate and distribution of unsafe categories sampled using our reward signal and the 7b reward model before and after training. The results are shown in Table 2.

Model	Antropic		Do-Not-Answer		Salad-Bench		Avg↓
	SG	MJ	SG	MJ	SG	MJ	
Llama2-7B-base	32.5%	56.6%	31.9%	22.2%	35.2%	68.3%	41.1%
Llama2-7B+SFT	19.2%	29.2%	31.7%	14.0%	29.6%	44.3%	28.0%
Llama2-7B+DPO	17.5%	29.5%	28.0%	9.70%	27.3%	42.7%	25.7%
Llama2-13B-base	34.9%	54.8%	20.7%	19.0%	35.1%	66.1%	38.4%
Llama2-7B+RS*	18.7%	35.7%	22.1%	13.4%	**17.7%**	43.4%	25.1%
Llama2-7B+DPO*	14.9%	33.5%	**16.1%**	**7.50%**	22.3%	42.9%	22.8%
Llama2-7B+cDPO*	**13.7%**	**27.6%**	25.3%	10.8%	18.0%	**32.8%**	**21.4%**
Llama2-7B-Online	6.9%	26.6%	8.6%	8.1%	13.5%	38.9%	17.1%
Llama2-13B+RS*	29.9%	49.4%	25.0%	16.8%	36.7%	60.2%	36.3%

Table 1: Comparison of our method with baseline methods across three different test sets and two different safety evaluation models. SG and MJ represent evaluation by Llama Guard 2 and MD-Judge, respectively. RS represents best of N, where N is 8 in our setting. cDPO indicates tuning with preference confidence sampling. The online method uses the trained 7B reward model to sample every 100 steps, serving as the theoretical upper bound of our method.

Although our reward signal is slightly less effective than the 7B reward model in safety classification, both reward show high consistency in overall safety classification before and after our alignment. Notably, post-training, the distribution gap in some unsafe categories increased in the top 2 and top 1 settings. We believe this is due to the reduction in the total number of unsafe outputs amplifying the original differences. Given that the maintenance cost of our reward model is negligible compared to the 7B reward model, this performance difference is acceptable.

4.6 Exaggerated Safety

To detect exaggerated safety phenomena in the alignment process, we tested the method and baselines on the overly conservative test set Xstest (Röttger et al., 2024). We tested the behavior of the policy model in response to both safe prompts and unsafe prompts, and the results are shown in Figure 4.

As shown in the figure 4, our alignment method increases the model's rejection rate of unsafe responses. Notably, using either an trained reward model or our reward signal for best-of-N sampling significantly enhances the model's proportion of "partial refusal". On the other hand, fixed label confidence, compared to our method's dynamic label confidence, tends to increase the proportion of "partial refusal". This could be due to the preference noise introduced by the fixed label confidence during tuning, making the model more inclined towards ambiguous responses. Additional alignment experiments can be found in the appendix 3.

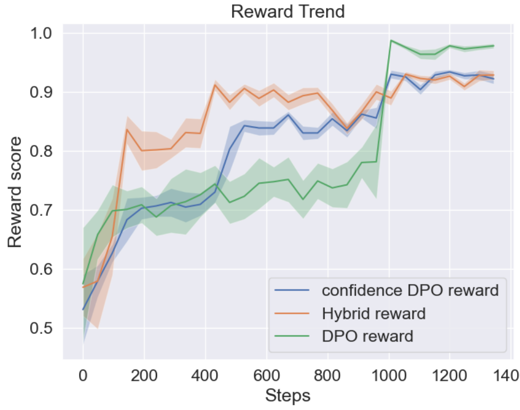

Figure 5: The trend of reward scores during the alignment process. The hybrid reward (Orange) and the confidence DPO reward (Blue) are calculated by Eq 3 and Eq 8. The vanilla DPO reward (Green) is also shown in the same setting.

4.7 Analysis

According to (Mitchell, 2023), the gradient of object $\mathcal{L}^{\epsilon}_{DPO}$ in Equation 5 is,

$$\nabla_\theta \mathcal{L}^{\epsilon}_{DPO} = (\hat{p_\theta} - \gamma_{x,y_c,y_r})\left[\nabla_\theta log\pi_\theta(y_c) - \nabla_\theta log\pi_\theta(y_r)\right] \quad (9)$$

In which $\hat{p_\theta}$ equals to $\sigma(r(x, y_c) - r(x, y_r))$ and $1 - \epsilon$ is replaced with γ_{x,y_c,y_r}. Considering that r is the reward signal DPO uses, this is exactly the current policy's preference in the form of B-T model. The term $\nabla_\theta log\pi_\theta(y_c) - \nabla_\theta log\pi_\theta(y_r)$ is the difference between the optimization directions of the chosen and the rejected responses, which maintains consistency. The gradient is equal to zero when $\hat{p_\theta} = \gamma_{x,y_c,y_r}$. As γ_{x,y_c,y_r} is the preference confidence of the target optimal policy, which indicates the current policy preference will

0 epoch

Model	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	S11	Toxic rate
top-1-ours	20.41%	39.25%	5.64%	0.12%	2.88%	12.85%	0.60%	0.24%	12.24%	1.20%	4.56%	20.82%
top-1-rm	20.09%	40.18%	6.03%	0	2.63%	15.15%	0.46%	0.15%	9.43%	1.24%	4.64%	16.18%
top-2-ours	18.95%	39.93%	5.64%	0.05%	2.50%	12.35%	0.64%	0.27%	13.90%	1.06%	4.69%	23.48%
top-2-rm	20.33%	40.36%	5.87%	0.06%	2.10%	13.84%	0.49%	0.12%	11.19%	0.99%	4.64%	20.23%
top-4-ours	18.40%	39.75%	6.24%	0.02%	2.42%	12.23%	0.64%	0.23%	14.10%	1.05%	4.91%	27.34%
top-4-rm	19.20%	40.63%	5.90%	0.023%	2.22%	13.04%	0.56%	0.19%	12.67%	1.17%	4.41%	26.79%
sample-8	17.45%	40.38%	6.47%	0.01%	2.16%	11.91%	0.50%	0.17%	13.64%	1.17%	4.47%	33.36%

1 epoch

Model	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	S11	Toxic rate
top-1-our	10.00%	33.33%	8.33%	5.00%	25.00%	1.67%	0%	10.00%	0%	5.00%	5.00%	12.00%
top-1-rm	14.29%	34.29%	8.57%	0%	0%	20.00%	0%	0%	11.43%	2.86%	8.57%	7.00%
top-2-our	10.40%	30.40%	9.60%	0.80%	2.40%	27.20%	0.80%	0%	9.60%	0.80%	8.00%	12.50%
top-2-rm	13.86%	32.67%	8.91%	0%	0.99%	19.80%	0.99%	0%	13.86%	0.99%	7.92%	10.10%
top-4-ours	13.73%	30.28%	8.45%	0.35%	2.11%	24.30%	0.35%	0%	11.27%	1.06%	8.10%	14.20%
top-4-rm	12.10%	31.21%	8.28%	0.32%	2.23%	25.48%	0.31%	0%	11.46%	1.59%	7.80%	15.70%
sample-8	12.87%	32.92%	8.17%	0.12%	2.10%	23.64%	0.25%	0%	12.25%	1.36%	6.31%	20.20%

Table 2: The safety taxonomy distribution compared between hybrid reward model and trained 7B reward model sampling from reference policy and aligned policy. **S1** to **S11** represent different unsafe categories based on the MLCommons hazard classification, with each category indicating its proportion among all unsafe outputs. We present the overall **Toxic rate** for each sampling setting.

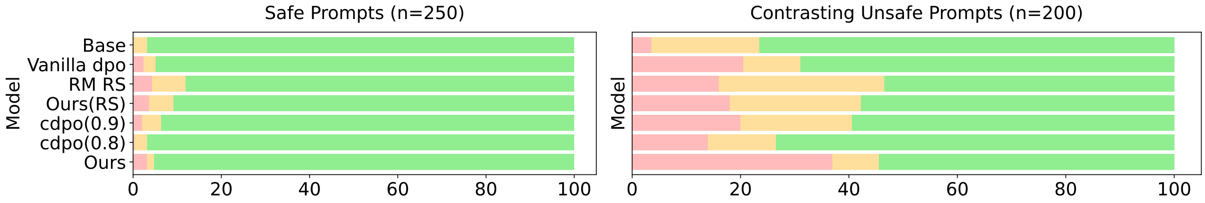

Figure 4: The evaluation result of exaggerated safety behaviours on XStest benchmark. On the left and right are the response behaviors of the aligned model to safe prompts and unsafe prompts, respectively. The red part indicates refusal to reply, the yellow part indicates partial refusal, and the green part indicates full compliance, evaluated by GPT-4o.

eventually converge on the target optimal policy preference.

As depicted in Figure 5, our reward signal and DPO reward increase gradually, which show that the sampling preference remains stable throughout the training process, while the policy preference gradually aligns with this stable preference. Notably, after approximately 1000 steps, the vanilla DPO reward experiences a significant surge and sustains a high value, which suggests the occurrence of reward hacking (Ibarz et al., 2018).

4.8 Reward Strategy

We evaluated the reward signal under different strategies by using the top-4 sampling from prompts of PKU-SafeRLHF test set and the hh-rlhf red-team. Since Our method weights reward signals from all layers, which implies a theoretical upper limit: for each sample, one layer most accurately reflects the oracle reward score. As Figure 6 illustrates, **Best** strategy selects the oracle reward from best layer, representing the upper bound of our reward modeling method and the **worst** strategy selects the worst reward, representing the lower bound. We compared using the last layer for reward extraction and found higher toxicity across categories compared to our method, validating the initial probing result. For each unsafe category, our method performs strictly worse than using reward signals extracted from the final layer's output, but it remains close to the optimal strategy. **The gap between our method and the optimal reward indicates that there is still room for improvement in the performance of this reward modeling approach.**

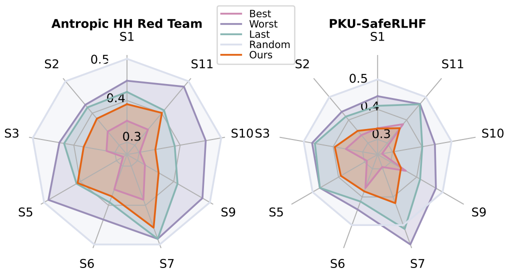

Figure 6: Toxic rate across different reward strategies. **Best**: selecting signals from the layer with the best performance (Oracle); **Worst**: choosing the signals from the worst layer; **Last**: using the last layer to extract reward signals; **Random** and **Ours**.

4.9 Computational Efficiency

We estimated the additional computational overhead, excluding policy model updates, for a 7B-sized model using a reward model of the same size and our method across different sequence lengths, including sampling and reward model update(Ours) costs. As shown in Figure 7, Our method is 5-6 orders of magnitude lower than the online method, and does not increase with the sequence length

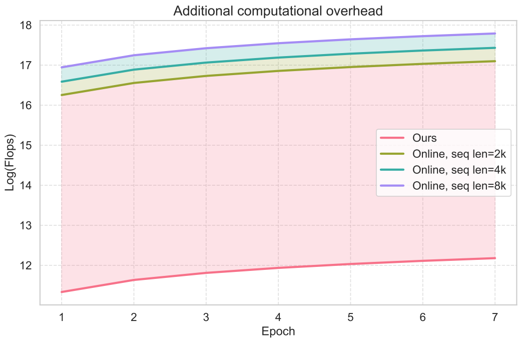

Figure 7: Computational Efficiency compared with online sampling method on 7B model.

5 Related Work

5.1 Preferences Alignment

Preference alignment aims to align the policy with human preferences. On-policy RLHF (Ouyang et al., 2022; Christiano et al., 2017) fits a reward model from human feedback preference data by optimizing a B-T preference model.Leike et al. (2018) aligns systems with human performance using a reward model; Stiennon et al. (2020) fine-tuned language models for summarization tasks by training a reward model to fit human preferences; Bai et al. (2022) trained a reward model to align LLMs like GPT-3 towards honesty, helpfulness, and harmlessness. Off-policy methods, such as DPO, bypass reward modeling and directly align LLMs on preference data. Mitchell (2023); Chowdhury et al. (2024) notes that preference data may be noisy and over-confident. Online data sampling from the reference policy often yields better results (Xiong et al., 2024). Our work uses the B-T model to estimate preference confidence, which mitigates distribution shift.

5.2 Language Model Probing

Probing examines internal model representations by training linear classifiers (probes) on hidden states to identify specific input(Alain and Bengio, 2016; Tenney, 2019; Belinkov, 2022). Research by Gurnee and Tegmark (2023) indicate that language models acquire real-world representations during training. Li et al. (2024a) notes a significant gap between generation accuracy and probe accuracy in QA tasks. Fan et al. (2024) uses a linear SVM to extract internal signals for early stopping in early layers. Other findings highlight the rich information in internal representations(Zou et al., 2023). Wang et al. (2024a) shows the potential of safety representations in model alignment by editing internal representations to detoxify LLMs. Kong et al. (2024) aligns LLMs through representation editing from a control perspective. These studies highlight the rich information in internal representations.

6 Conclusion

Reinforcement learning for large language models (LLMs) encounters significant challenges related to distribution shift in safety alignment, often necessitating substantial computational costs to address. In this paper, we hypothesize that during DPO training, while the rankings of top items change, their overall distribution remains largely unchanged. Base on it we transform the sampling process from the target policy into a re-ranking of the preference data. Specifically, we leverage the model's internal safety judgment capability to extract reward signals and use label confidence to simulate the sampling process and optimize the DPO loss with preference confidence. Our theoretical analysis and experimental results demonstrate that this method significantly reduces policy toxicity, enhancing the alignment of policy models with safety preferences.

Limitations

Although we have analyzed the general performance changes of the model, we have not conducted further analysis of potential distribution shift unrelated to safety. Due to experimental costs and there is a lack of validation on larger-scale models. Additionally, our method exhibits a gap compared to vanilla online methods, this is further evident in the divergence between our reward signal and its theoretical upper bound, which we attribute to the simplicity of our reward extraction method. This reflects a trade-off between computational efficiency and the accuracy of performance.

References

Guillaume Alain and Yoshua Bengio. 2016. Understanding intermediate layers using linear classifier probes. *arXiv preprint arXiv:1610.01644*.

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*.

Yonatan Belinkov. 2022. Probing classifiers: Promises, shortcomings, and advances. *Computational Linguistics*, 48(1):207–219.

Sayak Ray Chowdhury, Anush Kini, and Nagarajan Natarajan. 2024. Provably robust dpo: Aligning language models with noisy feedback. *arXiv preprint arXiv:2403.00409*.

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.

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*.

Siqi Fan, Xin Jiang, Xiang Li, Xuying Meng, Peng Han, Shuo Shang, Aixin Sun, Yequan Wang, and Zhongyuan Wang. 2024. Not all layers of llms are necessary during inference. *arXiv preprint arXiv:2403.02181*.

Wes Gurnee and Max Tegmark. 2023. Language models represent space and time. In *The Twelfth International Conference on Learning Representations*.

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*.

Borja Ibarz, Jan Leike, Tobias Pohlen, Geoffrey Irving, Shane Legg, and Dario Amodei. 2018. Reward learning from human preferences and demonstrations in atari. *Advances in neural information processing systems*, 31.

Hakan Inan, Kartikeya Upasani, Jianfeng Chi, Rashi Rungta, Krithika Iyer, Yuning Mao, Michael Tontchev, Qing Hu, Brian Fuller, Davide Testuggine, et al. 2023. Llama guard: Llm-based input-output safeguard for human-ai conversations. *arXiv preprint arXiv:2312.06674*.

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. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

Michael I Jordan and Robert A Jacobs. 1994. Hierarchical mixtures of experts and the em algorithm. *Neural computation*, 6(2):181–214.

Lingkai Kong, Haorui Wang, Wenhao Mu, Yuanqi Du, Yuchen Zhuang, Yifei Zhou, Yue Song, Rongzhi Zhang, Kai Wang, and Chao Zhang. 2024. Aligning large language models with representation editing: A control perspective. *arXiv preprint arXiv:2406.05954*.

Jan Leike, David Krueger, Tom Everitt, Miljan Martic, Vishal Maini, and Shane Legg. 2018. Scalable agent alignment via reward modeling: a research direction. *arXiv preprint arXiv:1811.07871*.

Kenneth Li, Oam Patel, Fernanda Viégas, Hanspeter Pfister, and Martin Wattenberg. 2024a. Inference-time intervention: Eliciting truthful answers from a language model. *Advances in Neural Information Processing Systems*, 36.

Lijun Li, Bowen Dong, Ruohui Wang, Xuhao Hu, Wangmeng Zuo, Dahua Lin, Yu Qiao, and Jing Shao. 2024b. Salad-bench: A hierarchical and comprehensive safety benchmark for large language models. *arXiv preprint arXiv:2402.05044*.

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

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.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D Manning, and Chelsea Finn. 2023. Direct preference optimization: your language model is secretly a reward model. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, pages 53728–53741.

Paul Röttger, Hannah Kirk, Bertie Vidgen, Giuseppe Attanasio, Federico Bianchi, and Dirk Hovy. 2024. XSTest: A test suite for identifying exaggerated safety behaviours in large language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 5377–5400, Mexico City, Mexico. Association for Computational Linguistics.

Emily Sheng, Kai-Wei Chang, Prem Natarajan, and Nanyun Peng. 2019. The woman worked as a babysitter: On biases in language generation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3407–3412.

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

Yi Chern Tan and L Elisa Celis. 2019. Assessing social and intersectional biases in contextualized word representations. *Advances in neural information processing systems*, 32.

I Tenney. 2019. Bert rediscovers the classical nlp pipeline. *arXiv preprint arXiv:1905.05950*.

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*.

A Vaswani. 2017. Attention is all you need. *Advances in Neural Information Processing Systems*.

Mengru Wang, Ningyu Zhang, Ziwen Xu, Zekun Xi, Shumin Deng, Yunzhi Yao, Qishen Zhang, Linyi Yang, Jindong Wang, and Huajun Chen. 2024a. Detoxifying large language models via knowledge editing. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3093–3118, Bangkok, Thailand. Association for Computational Linguistics.

Yuxia Wang, Haonan Li, Xudong Han, Preslav Nakov, and Timothy Baldwin. 2024b. Do-not-answer: Evaluating safeguards in LLMs. In *Findings of the Association for Computational Linguistics: EACL 2024*, pages 896–911, St. Julian's, Malta. Association for Computational Linguistics.

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

Shusheng Xu, Wei Fu, Jiaxuan Gao, Wenjie Ye, Weilin Liu, Zhiyu Mei, Guangju Wang, Chao Yu, and Yi Wu. 2024. Is dpo superior to ppo for llm alignment? a comprehensive study. In *Forty-first International Conference on Machine Learning*.

Andy Zou, Long Phan, Sarah Chen, James Campbell, Phillip Guo, Richard Ren, Alexander Pan, Xuwang Yin, Mantas Mazeika, Ann-Kathrin Dombrowski, et al. 2023. Representation engineering: A top-down approach to ai transparency. *arXiv preprint arXiv:2310.01405*.

A Appendix

A.1 Over-alignment

To evaluate the over-alignment, we test the aligned model on MMLU(Hendrycks et al., 2020). Additionally, we selected prompts from the Alpaca-Eval (Dubois et al., 2024) and used two existing reward model to score the outputs, particularly FsfairX3 and deberta-v3-large-v2, both are used or RLHF. The result in Table 3 show that there is a slight decline in general capabilities, which is acceptable Considering the conflict between safety alignment and general capabilities.

Model	RM-deberta	FsfairX	MMLU
Base	-4.309	-2.911	0.45898
Vanilla-dpo	-4.518	-2.909	0.45947
Ours	-4.410	-2.747	0.43476

Table 3: Response score for aligned policy, as well as the MMLU scores.

A.2 PCA Result of Llama2-13B

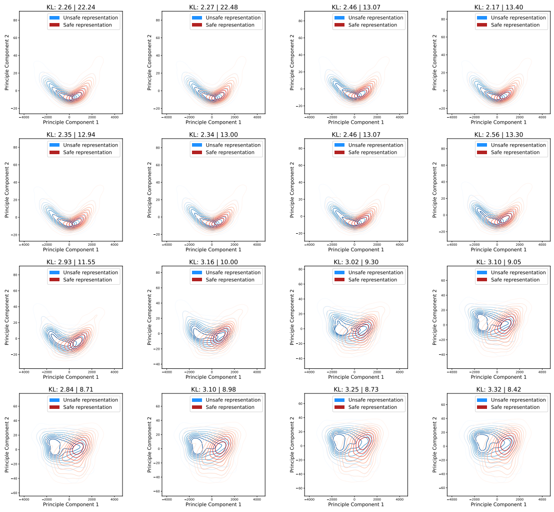

Figure 8: Kernel density estimate plots show the hidden states of unsafe output (blue) and safe output (red) pairs in different layers of Llama-13B after projection onto the top-2 principal directions.

A.3 Parameter Setting

In our experiments, the DPO algorithm employs $\beta = 1.5$,$lr = 1e - 5$, batch size is 4. In our approach, the optimization margin $\mu = 1$ in Equation 2. The scaling factor for preference confidence $\alpha = 7.5$ in Equation 1.