Evaluation of Best-of-N Sampling Strategies for Language Model Alignment

Yuki Ichihara *ichihara.yuki.iu1@is.naist.jp*
Nara Institute of Science and Technology

Yuu Jinnai *jinnai_yu@cyberagent.co.jp*
Tetsuro Morimura *morimura_tetsuro@cyberagent.co.jp*
Kenshi Abe *abe_kenshi@cyberagent.co.jp*
Kaito Ariu *kaito_ariu@cyberagent.co.jp*
Mitsuki Sakamoto *sakamoto_mitsuki@cyberagent.co.jp*
CyberAgent

Eiji Uchibe *uchibe@atr.jp*
Advanced Telecommunications Research Institute International

Reviewed on OpenReview: *https://openreview.net/forum?id=H4S4ETc8c9*

Abstract

Best-of-N (BoN) sampling with a reward model has been shown to be an effective strategy for aligning Large Language Models (LLMs) with human preferences at the time of decoding. BoN sampling is susceptible to a problem known as *reward hacking*. Since the reward model is an imperfect proxy for the true objective, an excessive focus on optimizing its value can lead to a compromise of its performance on the true objective. Previous work proposes Regularized BoN sampling (RBoN), a BoN sampling with regularization to the objective, and shows that it outperforms BoN sampling so that it mitigates reward hacking and empirically (Jinnai et al., 2024). However, Jinnai et al. (2024) introduce RBoN based on a heuristic and they lack the analysis of *why* such regularization strategy improves the performance of BoN sampling. The aim of this study is to analyze the effect of BoN sampling on regularization strategies. Using the regularization strategies corresponds to robust optimization, which maximizes the worst case over a set of possible perturbations in the proxy reward. Although the theoretical guarantees are not directly applicable to RBoN, RBoN corresponds to a practical implementation. This paper proposes an extension of the RBoN framework, called Stochastic RBoN sampling (SRBoN), which is a theoretically guaranteed approach to worst-case RBoN in proxy reward. We then perform an empirical evaluation using the AlpacaFarm and Anthropic's hh-rlhf datasets to evaluate which factors of the regularization strategies contribute to the improvement of the true proxy reward. In addition, we also propose another simple RBoN method, the Sentence Length Regularized BoN, which has a better performance in the experiment as compared to the previous methods.

1 Introduction

Large Language Models (LLMs) have demonstrated remarkable capabilities in many NLP tasks related to natural language understanding and text generation (Stiennon et al., 2020; Ouyang et al., 2022; Touvron et al., 2023; Dubey et al., 2024; OpenAI et al., 2024). Despite the strengths, LLMs are not always adept at interpreting a wide range of instructions and can produce undesirable outputs, such as biased, hallucinated, or toxic responses (Bai et al., 2022; Lin et al., 2022; Touvron et al., 2023; Casper et al., 2023; Guan et al., 2024). This problem underscores the challenge of language model alignment; ensuring LLMs' behaviors align with human objectives and safety considerations (Ziegler et al., 2020; Stiennon et al., 2020; Ouyang et al.,

2022). There is now a rich set of approaches to address this problem (Stiennon et al., 2020; Ouyang et al., 2022; Rafailov et al., 2023). These papers have shown that training language models with human feedback can improve their performance. However, training language models is a computationally intensive task. In other words, improved performance comes at the cost of increased computational resources.

This paper focuses on the Best-of-N (BoN) sampling strategy, a method that involves generating N outputs from a model and selecting the most preferred output among the N samples. Despite its simplicity and the fact that it does not require an additional training phase, BoN sampling has been shown to be surprisingly effective in practice (Stiennon et al., 2020; Nakano et al., 2022). However, the BoN strategy does not scale with the number of samples N due to the *reward hacking problem* (Amodei et al., 2016; Ziegler et al., 2020; Stiennon et al., 2020; Skalse et al., 2022; Gao et al., 2023). Reward hacking is a behavior that satisfies the given objective without achieving the intended result. It is caused by the misspecification of the preference model used by the BoN to select the most preferred output (Pan et al., 2022; Lambert & Calandra, 2024).

Previous work to mitigate the reward hacking problem has proposed Regularized Best-of-N sampling (RBoN), BoN strategy with the addition of a regularization term to the objective (Jinnai et al., 2024). This paper shows that the RBoN strategy is effective compared to BoN sampling in various experiments. However, *why* such a regularization strategy is effective against reward uncertainty is unclear in the previous work.

In this paper, we propose Stochastic RBoN sampling (SRBoN), which adds a regularization term similar to RBoN. We then draw a connection between the Reinforcement Learning (RL) problems (Sutton & Barto, 2018) and the BoN strategies: BoN sampling corresponds to solving the RL problem, and SRBoN sampling strategies correspond to solving the Regularized Reinforcement Learning (RRL) problem (Neu et al., 2017; Geist et al., 2019; Yang et al., 2019; Derman et al., 2021).

First, we exploit the knowledge of RRL. Some work has shown that regularization terms in probability distributions over outputs provide robustness to reward perturbations (Ortega & Lee, 2014; Husain et al., 2021; Derman et al., 2021; Eysenbach & Levine, 2022; Pan et al., 2022; Derman et al., 2021). SRBoN can also apply this analysis to RRL. Its insights provide an answer to why reward hacking can be mitigated: when a regularization term is added to the BoN sampling, it also becomes an adversarial perturbation to the reward.

We then evaluate the effectiveness of our approach against alternative decoder methods in a series of experiments, with the goal of determining the relative resilience of each method to potential exploitation by reward hacking. The results show that our proposed method outperforms many existing approaches in a variety of settings. In other words, a theoretically guaranteed, effective algorithm is proposed.

In addition, while RBoN consists of complex formula structures, we proposed a simpler RBoN, Sentence Length Regularized BoN that, despite its simple implementation, shows comparable or even better performance in experiments with the methods of previous studies.

2 Background

In this paper, we formalize the problem of decoding time alignment as Regularized Markov Decision Processes (MDPs) problem (Neu et al., 2017; Geist et al., 2019; Yang et al., 2019; Derman et al., 2021). For brevity, we refer to reinforcement learning within Regularized MDPs as Regularized Reinforcement Learning (RRL) throughout this paper. In Section 2.1, we describe the Reinforcement Learning (RL) problem and the RRL problem. Then, we describe two sampling algorithms used for decoding time alignment, Best-of-N (BoN) sampling in Section 2.2, and the Regularized Best-of-N sampling (RBoN) in Section 2.3.

2.1 Adversarial Interpretation in Regularized Reinforcement Learning

We consider the problem of selecting an output y from a set of outputs $\mathcal{Y}_{\mathbf{ref}} \subseteq \mathcal{Y}$ (e.g., response text from the system) given an input $x \in \mathcal{X}$ (e.g., input prompt by a user), where the objective is to select the best output according to a reward function R: $\mathcal{X} \times \mathcal{Y}_{\mathbf{ref}} \to \mathbb{R}$. Let $\Delta(\mathcal{Y})$ denote the set of probability distributions over a set $\mathcal{Y}$. We define the goal of the Reinforcement Learning (RL) problem as finding the best policy

$\pi : \mathcal{X} \to \Delta(\mathcal{Y}_{\mathbf{ref}})$ that maximizes the expected reward for x:

$$\begin{aligned} \arg\max_{\pi \in \Pi} f_{\mathrm{RL}}(\pi) &= \arg\max_{\pi \in \Pi} \sum_{y \in \mathcal{Y}_{\mathbf{ref}}} \pi(y \mid x) R(x, y) \\ &= \arg\max_{\pi \in \Pi} \mathbb{E}_{y \sim \pi(\cdot|x)}[R(x, y)] \end{aligned} \tag{1}$$

where Π is a set of all possible policies. Note that there exists a deterministic policy that maximizes f_{RL} (Sutton & Barto, 2018) and this formulation can be seen as a contextual bandit problem. Specifically, the policy observes a context x, chooses an output y based on that context, and receives a reward $R(x, y)$. Importantly, we do not use sequential decision operations or consider state transitions. Each decision is made independently based on the current context x.

The underlying assumption of the RL problem is that the reward model R is correctly defined and observable. That is, we consider the solution that maximizes the expected reward to be the optimal solution. However, real-world applications often suffer from the *reward misspecification problem* – the reward model observable to the agent is only a proxy for the true underlying reward of the problem (Ortega & Lee, 2014; Husain et al., 2021; Derman et al., 2021; Eysenbach & Levine, 2022; Pan et al., 2022; Derman et al., 2021). Prior work has investigated strategies to optimize under the uncertainty in the observed reward. In contrast to the RL problem, Regularized Reinforcement Learning (RRL) incorporates regularization terms to achieve a solution that is robust to the reward misspecification (Neu et al., 2017; Geist et al., 2019; Yang et al., 2019; Derman et al., 2021). The objective of the RRL problem is to find the best policy $\pi : \mathcal{X} \to \Delta(\mathcal{Y}_{\mathbf{ref}})$ that maximizes the reward with an additional regularization function $\Omega(\pi) : \Delta(\mathcal{Y}_{\mathbf{ref}}) \to \mathbb{R} \cup \{+\infty\}$. Let $f^{\Omega}_{\mathrm{RRL}}(\pi) := \mathbb{E}_{y \sim \pi(\cdot|x)}[R(x, y)] - \Omega(\pi)$ be the objective function of RRL problem with Ω. Then, we define the following as the optimal solution to the RRL problem:

$$\arg\max_{\pi \in \Pi} f^{\Omega}_{\mathrm{RRL}}(\pi) = \arg\max_{\pi \in \Pi} \mathbb{E}_{y \sim \pi(\cdot|x)}[R(x, y)] - \Omega(\pi). \tag{2}$$

Note that, unlike the RL problem, there may not exist an optimal policy that is deterministic for the RRL problem (Geist et al., 2019).

Brekelmans et al. (2022) uses Legendre–Fenchel transformation (Touchette, 2005) to show that the RRL problem can be viewed as a variant of the RL problem with an adversarial agent adding perturbations to the reward $\Delta R : \mathcal{X} \times \mathcal{Y}_{\mathbf{ref}} \to \mathbb{R}$ if the regularization term Ω is convex and lower semi-continuous function (Boyd & Vandenberghe, 2004):

$$\arg\max_{\pi \in \Pi} f^{\Omega}_{\mathrm{RRL}}(\pi) = \arg\max_{\pi \in \Pi} \min_{\Delta R \in \mathcal{R}_{\Delta}} \mathbb{E}_{y \sim \pi(\cdot|x)}[R(x, y) - \Delta R(x, y)] + \Omega^*(\Delta R), \tag{3}$$

where $\Omega^* \in \mathbb{R}^{\mathcal{X} \times \mathcal{Y}_{\mathbf{ref}}}$ is the conjugate function of Ω (Boyd & Vandenberghe, 2004), and $\mathcal{R}_{\Delta} := \left\{ \Delta R \in \mathbb{R}^{\mathcal{X} \times \mathcal{Y}_{\mathbf{ref}}} \mid F_{\Omega}(\Delta R) \leq 0 \right\}$, where F_{Ω} is a function or operator dependent to Ω that imposes a constraint or condition on the values of ΔR.

Eq. (3) shows that the problem of maximizing f_{RRL} can be reformulated as the max-min problem and the regularizer Ω is effectively an adversarial reward perturbation that forces us to optimize the worst case performance (Derman et al., 2021).

2.2 Best-of-N (BoN) Sampling

BoN sampling has emerged as an effective method for preference optimization in LLMs (Stiennon et al., 2020; Nakano et al., 2022). BoN sampling has several advantages over preference learning methods. First, it is straightforward and does not require additional training in the language model. Although learning-based alignment methods require retraining the LLMs whenever human preferences are updated, BoN sampling can be applied immediately, requiring only an update of the reward model. This is particularly advantageous since training LLMs is the most resource-intensive process. Second, BoN sampling is an effective strategy in its own right, with numerous studies demonstrating that it can outperform learning-based adaptation methods (Gao et al., 2023). Recent literature has expanded our understanding of BoN sampling. In particular, Beirami

et al. (2024) conducted an analysis comparing the policies selected by BoN sampling with the base policies used for sample generation. In addition, Gui et al. (2024) showed that BoN sampling achieves an optimal balance between win rate and KL divergence when aligning large language models to human preferences.

BoN sampling has similarities to the objective function used in RL (e.g., the response with the highest reward score, determined by a proxy reward model $R(x, y)$, is selected). The objective function of BoN is given by:

$$y_{\mathrm{BoN}}(x) := \arg\max_{y \in \mathcal{Y}_{\mathbf{ref}}} R(x, y).$$

This reward model is used as a measure of the quality of the text when making an assignment. The reward model we consider in this paper is open access as described in Appendix I. We also mention that the objective function of BoN sampling is equal to the objective function of the (unregularized) RL problem (Eq. (1)):

$$\begin{aligned} y_{\mathrm{BoN}}(x) : &= \arg\max_{y \in \mathcal{Y}_{\mathbf{ref}}} R(x, y) \\ &= \arg\max_{y \in \mathcal{Y}_{\mathbf{ref}}} \max_{\pi_y \in \Pi_{det}} \mathbb{E}_{y \sim \pi_y}[R(x, y)] \\ &= \arg\max_{y \in \mathcal{Y}_{\mathbf{ref}}} \max_{\pi_y} f_{\mathrm{RL}}(\pi_y). \end{aligned} \tag{4}$$

where Π_{det} is a set of deterministic policies and π_y is a deterministic policy that selects y given x with a probability of 1.

2.3 Regularized Best-of-N Sampling (RBoN)

Although BoN sampling is shown to be effective, it is prone to the reward hacking problem (Amodei et al., 2016; Ziegler et al., 2020; Stiennon et al., 2020; Skalse et al., 2022; Gao et al., 2023). The reward hacking problem is a phenomenon where the decision to optimize the proxy reward is made without considering its potential misspecification, resulting in worse performance on the actual reward objective. Dubois et al. (2023) showed that with 25% label noise, which is the amount of disagreement observed in real-world preference annotations (Stiennon et al., 2020; Ouyang et al., 2022), BoN sampling degrades performance with N greater than 16 (Figures 12 and 13 in Dubois et al. 2023).

Regularized Best-of-N sampling (RBoN) is proposed to mitigate the reward hacking problem for BoN sampling (Jinnai et al., 2024). Jinnai et al. (2024) presented two variants of RBoN: using the KL divergence as a regularizer ($\mathrm{RBoN}_{\mathrm{KL}}$; Section 2.3.1) and using the Wasserstein Distance as a regularizer ($\mathrm{RBoN}_{\mathrm{WD}}$; Section 2.3.2). In the following, we describe the two variants of RBoN and draw its connection to the objective function of RRL (Eq. (2)).

2.3.1 KL divergence Regularized BoN Sampling ($\mathrm{RBoN}_{\mathrm{KL}}$)

The objective function of $\mathrm{RBoN}_{\mathrm{KL}}$ (KL divergence Regularized BoN Sampling) is given by:

$$\begin{aligned} y_{\mathrm{KLBoN}}(x) &= \arg\max_{y \in \mathcal{Y}_{\mathbf{ref}}} R(x, y) - \beta D_{\mathrm{KL}}\left[\pi_y(\cdot \mid x) \| \pi_{\mathbf{ref}}(\cdot \mid x)\right], \\ &= \arg\max_{y \in \mathcal{Y}_{\mathbf{ref}}} \max_{\pi_y \in \Pi_{det}} f^{\mathrm{KL}}_{\mathrm{RRL}}(\pi_y). \end{aligned}$$

where β is a regularization parameter, reference policy $\pi_{\mathbf{ref}}$: $\mathcal{X} \to \Delta(\mathcal{Y}_{\mathbf{ref}})$, and D_{KL} denotes the KL divergence. The reference policy here takes an input x and returns a meaningful output y. The regularization described in this paper is like a penalty to stay away from the reference policy and the reference policy is the language model.

By incorporating the KL divergence as a regularization term in the objective function, $\mathrm{RBoN}_{\mathrm{KL}}$ encourages the learned policy to be close to the reference policy $\pi_{\mathbf{ref}}$. A higher value of β emphasizes the regularization term, encouraging the learned policy to be closer to the reference policy, while a lower value of β prioritizes maximizing the reward function.

2.3.2 Wasserstein Distance Regularized BoN Sampling ($\mathrm{RBoN}_{\mathrm{WD}}$)

The objective function of $\mathrm{RBoN}_{\mathrm{WD}}$ (Wasserstein Distance Regularized BoN Sampling) is defined as follows:

$$\begin{aligned} y_{\mathrm{WDBoN}}(x) &= \arg\max_{y \in \mathcal{Y}_{\mathbf{ref}}} R(x,y) - \beta \mathbf{WD}\left[\pi_y(\cdot \mid x) \| \pi_{\mathbf{ref}}(\cdot \mid x)\right], \\ &= \arg\max_{y \in \mathcal{Y}_{\mathbf{ref}}} \max_{\pi_y \in \Pi_{det}} f^{\mathrm{WD}}_{\mathrm{RRL}}(\pi_y). \end{aligned}$$

where **WD** denotes 1-Wasserstein Distance.

The **WD** (Wang, 2012) is defined as:

$$\mathbf{WD}[\nu \| \mu] = \inf_{\gamma \in \Gamma(\nu,\mu)} \sum_{(i,j) \in N \times N} \gamma_{ij} C_{ij}, \tag{5}$$

where N: the total number of samples, consisting of the set $\{y_1, y_2, \ldots, y_N\}$, $\nu, \mu \in \Delta(N)$: probability measure on the aforementioned sets (ν_i, μ_i refer to the probability value $\nu(y_i), \mu(y_i)$), C: $N \times N \to \mathbb{R}$ a cost function measuring the distance between two outputs (e.g. C_{ij} refers to the amount to be transported from place y_i to palace y_j), and $\Gamma(\nu, \mu)$ denotes the set of all joint distributions γ whose marginals are ν and μ. The constraints on γ are given by:

$$\begin{aligned} \sum_{j \in n} \gamma_{ij} &= \nu_i, \quad \forall i \in n, \\ \sum_{i \in n} \gamma_{ij} &= \mu_j, \quad \forall j \in n, \\ \gamma_{ij} &\geq 0, \quad \forall i, j \in n. \end{aligned}$$

The **WD**, also known as the Earth Mover's Distance (EMD), is a metric used to quantify the dissimilarity between two probability distributions. Intuitively, it measures the minimum cost required to transform one distribution into the other. This cost is conceptualized as the amount of probability mass that must be moved multiplied by the distance that would be moved. The concept has been used in NLP to measure the dissimilarity of texts (Kusner et al., 2015; Zhao et al., 2019).

The exact computation of $\mathbf{WD}\left[\pi_y(\cdot \mid x) \| \pi_{\mathbf{ref}}(\cdot \mid x)\right]$ is intractable due to the enormous size of the output space. To address this computational challenge, prior work (Jinnai et al., 2024) has employed sample-based approximation techniques. Let $\hat{\pi}_{\mathbf{ref}}$ represent the empirical distribution computed using a set of samples $\mathcal{Y}_{\mathbf{ref}}$. This distribution is defined as: $\hat{\pi}_{\mathbf{ref}}(y \mid x) = \frac{1}{N} \sum_{y' \in \mathcal{Y}_{\mathbf{ref}}} \mathbb{I}(y = y')$ where N is the total number of samples in $\mathcal{Y}_{\mathbf{ref}}$. The objective function can then be approximated as follows:

$$y_{\mathrm{WDBoN}}(x) = \arg\max_{y \in \mathcal{Y}_{\mathbf{ref}}} R(x,y) - \beta \mathbf{WD}\left[\pi_y(\cdot \mid x) \| \hat{\pi}_{\mathbf{ref}}(\cdot \mid x)\right].$$

For practical implementation aspects, the **WD** term for Jinnai et al. (2024) is computed as follows:

$$\mathbf{WD}\left[\pi_y(\cdot \mid x) \| \hat{\pi}_{\mathbf{ref}}(\cdot \mid x)\right] = \sum_{y' \in \mathcal{Y}_{\mathbf{ref}}} \frac{1}{N} C\left(y, y'\right), \tag{6}$$

The cosine distance is used as C to measure the distance between the outputs (Reimers & Gurevych, 2019).

$$C\left(y, y'\right) = 1 - \cos\left(\mathrm{emb}(y), \mathrm{emb}\left(y'\right)\right), \tag{7}$$

where $\mathrm{emb}(y)$ and $\mathrm{emb}\left(y'\right)$ represent the embeddings of output y and y', respectively.

3 Stochastic RBoN (SRBoN)

We propose the stochastic version of RBoN, Stochastic $\mathrm{RBoN}_{\mathrm{KL}}$ (Section 3.1) and Stochastic $\mathrm{RBoN}_{\mathrm{WD}}$ (Section 3.2). These novel algorithms, while similar to the original RBoN (deterministic version), allow for the optimal policy π to a probabilistic output distribution. By relaxing the deterministic constraint, we can apply theoretical tools that were previously inaccessible. Our approach focuses on the analysis of this stochastic version, aiming to provide theoretical results that shed light on the underlying mechanisms of RBoN's effectiveness.

3.1 Stochastic $\mathrm{RBoN}_{\mathrm{KL}}$ ($\mathrm{SRBoN}_{\mathrm{KL}}$)

First, consider a stochastic version of $\mathrm{RBoN}_{\mathrm{KL}}$. The policy of $\mathrm{SRBoN}_{\mathrm{KL}}$ is given by:

$$\begin{aligned}\pi_{\mathrm{SRBoN}_{\mathrm{KL}}}(x) &= \arg\max_{\pi\in\Pi} \mathbb{E}_{y\sim\pi(\cdot|x)}[R(x,y)] - \beta D_{\mathrm{KL}}\left[\pi(\cdot \mid x) \| \pi_{\mathbf{ref}}(\cdot \mid x)\right] \\ &= \arg\max_{\pi\in\Pi} f^{\mathrm{KL}}_{\mathrm{RRL}}(\pi).\end{aligned} \tag{8}$$

We define $\mathrm{SRBoN}_{\mathrm{KL}}$ as a method to sample a response y that follows the probability distribution of $\pi_{\mathrm{SRBoN}_{\mathrm{KL}}}$:

$$y_{\mathrm{SRBoN}_{\mathrm{KL}}}(x) \sim \pi_{\mathrm{SRBoN}_{\mathrm{KL}}}(x). \tag{9}$$

In Section 4 we evaluate the performance of this stochastic text generation algorithm defined by Eq. (9).

3.1.1 Theoretical Guarantee of $\mathrm{SRBoN}_{\mathrm{KL}}$

By relaxing the deterministic policy constraint of $\mathrm{RBoN}_{\mathrm{KL}}$, $\mathrm{SRBoN}_{\mathrm{KL}}$ follows the formulation of the RRL with adversarial perturbations studied by Brekelmans et al. (2022). As such, the computation of $\mathrm{SRBoN}_{\mathrm{KL}}$ can be transformed into a max-min problem using Legendre-Fenchel transformation (Touchette, 2005) as in Eq. (3). In this way, $\mathrm{SRBoN}_{\mathrm{KL}}$ has the following theoretical guarantee proven by Brekelmans et al. (2022):

Theorem 3.1. *(**Brekelmans et al. (2022), Proposition 1**) The problem of maximizing $f^{\mathrm{KL}}_{\mathrm{RRL}}(\pi)$ can be interpreted as a robust optimization problem with an adversarial perturbation ΔR:*

$$\arg\max_{\pi\in\Pi} f^{\mathrm{KL}}_{\mathrm{RRL}}(\pi) = \arg\max_{\pi\in\Pi} \min_{\Delta R\in\mathcal{R}_\Delta} \mathbb{E}_{y\sim\pi(\cdot|x)}[R(x,y) - \Delta R(x,y)], \tag{10}$$

where the feasible set of reward perturbations $\mathcal{R}_\Delta$ available to the adversary is bounded:

$$\mathcal{R}_\Delta := \left\{ \Delta R \in \mathbb{R}^{\mathcal{X}\times\mathcal{Y}_{\mathbf{ref}}} \mid \sum_{\mathcal{Y}_{\mathbf{ref}}} \pi_{\mathbf{ref}}(y \mid x) \exp(\beta^{-1}\Delta R(x,y)) \le 1 \right\} \tag{11}$$

The theorem shows that $\mathrm{SRBoN}_{\mathrm{KL}}$ is an algorithm that optimizes the worst-case performance under the assumption that the error between the true reward and the given proxy reward model is guaranteed to be within $\mathcal{R}_\Delta$ (Eq. (11)).

Let $\mathcal{R}'$ be a set of possible reward models under the reward perturbations: $\mathcal{R}' := \{R - \Delta R \mid \Delta R \in \mathcal{R}_\Delta\}$. Let $f^{\mathrm{KL}}_{\mathrm{RRL}}(\pi; R)$ be the objective of the policy given a (proxy) reward model R. Then,

$$\begin{aligned}\arg\max_{\pi\in\Pi} f^{\mathrm{KL}}_{\mathrm{RRL}}(\pi; R) &= \arg\max_{\pi\in\Pi} \min_{\Delta R\in\mathcal{R}_\Delta} \mathbb{E}_{y\sim\pi(\cdot|x)}[R(x,y) - \Delta R(x,y)] \\ &= \arg\max_{\pi\in\Pi} \min_{R'\in\mathcal{R}'} \mathbb{E}_{y\sim\pi(\cdot|x)}[R'(x,y)] \\ &= \arg\max_{\pi\in\Pi} \min_{R'\in\mathcal{R}'} f^{\mathrm{KL}}_{\mathrm{RRL}}(\pi; R').\end{aligned} \tag{12}$$

Thus, $\mathrm{SRBoN}_{\mathrm{KL}}$ is a robust optimization of the policy for a set of possible reward models in $\mathcal{R}'$. In other words, it assumes that the true payoff model is in $\mathcal{R}_\Delta$ and optimizes for the worst case.

The theorem is derived by translating the proposition proved by Brekelmans et al. (2022) for the generic RRL problems to the text generation scenario. The contribution of our work is to show the relation of their theoretical result to the RBoN sampling algorithm in LLMs alignment.

3.2 Stochastic $\mathrm{RBoN}_{\mathrm{WD}}$ ($\mathrm{SRBoN}_{\mathrm{WD}}$)

We now consider an optimization problem over a space of probability functions, to derive an optimal probabilistic policy π with the Wasserstein distance as the regularization term. The objective function of $\mathrm{RBoN}_{\mathrm{SWD}}$ is the following:

$$\begin{aligned}\pi_{\mathrm{SRBoN_{WD}}}(x) &= \arg\max_{\pi\in\Pi} \mathbb{E}_{y\sim\pi(\cdot|x)}[R(x,y)] - \beta\mathbf{WD}[\pi_{\mathbf{ref}}(\cdot \mid x)\|\pi(\cdot \mid x)] \\ &= \arg\max_{\pi\in\Pi} f^{\mathrm{WD}}_{\mathrm{RRL}}(\pi).\end{aligned} \tag{13}$$

3.2.1 Theoretical Guarantee of $\mathrm{SRBoN_{WD}}$

Similar to $\mathrm{SRBoN_{KL}}$, $\mathrm{SRBoN_{WD}}$ can also be reformulated as a max-min problem, and thus we can show that it optimizes the worst-case performance under certain constrain:

Theorem 3.2. *The problem of maximizing $f^{\mathrm{WD}}_{\mathrm{RRL}}(\pi)$ can be interpreted as a robust optimization problem with an adversarial perturbation ΔR:*

$$\arg\max_{\pi\in\Pi} f^{\mathrm{WD}}_{\mathrm{RRL}}(\pi) = \arg\max_{\pi\in\Pi} \min_{\Delta R\in\mathcal{R}_\Delta} \mathbb{E}_{y\sim\pi(\cdot|x)}[R(x,y) - \beta\Delta R(x,y)] + \beta \sum_{y\in\mathcal{Y}_{ref}} \pi_{\mathbf{ref}}(y \mid x)\Delta R(x,y) \tag{14}$$

where the feasible set of reward perturbations $\mathcal{R}_\Delta$ available to the adversary is bounded:

$$\mathcal{R}_\Delta := \left\{\Delta R \in \mathbb{R}^{\mathcal{X}\times\mathcal{Y}_{ref}} \mid |\Delta R(x,y) - \Delta R(x,y')| \le C(y,y') \quad \forall y,y' \in \mathcal{Y}_{\mathbf{ref}}\right\}, \tag{15}$$

The proof is provided in Appendix A.

This expression represents an optimization problem with strategies π and perturbation ΔR. The goal is to find the optimal strategy π^* under the modified reward R' $(= R - \beta\Delta R)$.

The intuition behind the second term $\sum_{y\in\mathcal{Y}_{\mathbf{ref}}} \pi_{\mathbf{ref}}(y \mid x)\Delta R(x,y)$ can be understood by examining ΔR constraints (Eq. (15)). While this feasible set does not explicitly constrain ΔR to avoid large values, the second term, $\min_{\Delta R} \mathbb{E}_\pi[R(x,y) - \beta\Delta R(x,y)]$, helps to avoid such huge values. Additionally, it reveals a mechanism that inherently limits the magnitude of perturbations for actions that have high probability under $\pi_{\mathbf{ref}}$ and this is consistent with the WD distance intuition.

We have analyzed the role of the regularization term for BoN sampling in the previous section 3.1.1 and section 3.2.1. Since the previous study (Jinnai et al., 2024) imposed deterministic constraints, the results are not exactly the same, but we consider that the analysis performed here helps to explain why the previous study performed better.

Note The feasible set of reward perturbations $\mathcal{R}_\Delta$ is bounded to be a Lipschitz continuous function with respect to a cost function C, which generally takes a non-negative value in applications. The perturbation behavior corresponds to the Lipschitz continuity condition, which has traditionally been well-treated in the RL community. For example, previous studies such as Rachelson & Lagoudakis (2010); Pirotta et al. (2015) considered continuous state and action spaces in RL and derived Lipschitz continuity for reward functions to aid their analysis.

4 Experimental Evaluation

We evaluate the performance of SRBoN compared to other text generation approaches. The datasets and models used in the experiments are all publicly available (Appendix I).

Datasets. We conduct experiments using two datasets: the AlpacaFarm dataset (Dubois et al., 2023) and Anthropic's hh-rlhf (HH) dataset, which we use the Harmlessness and Helpfulness subsets (Bai et al., 2022). For the AlpacaFarm dataset, we use the first 1000 entries of the train split (alpaca human preference) as the development set and the 805 entries of the evaluation split (alpaca farm evaluation) for evaluation. For Anthropic's datasets, we separately conduct experiments on the helpful-base (Helpfulness) and harmless-base (Harmlessness). For each dataset, we use the first 1000 entries of the train split as the development set and the first 1000 entries of the evaluation split for evaluation.

Table 1: Description of the text generation algorithms evaluated in the experiments. A checkmark (✓) indicates that the method uses the specified function, while a blank space means that it does not.

Method	Reward Function	Similarity Function	Description
Random sampling			Use an output that is randomly sampled from the reference model.
Best-of-N (BoN) (Stiennon et al., 2020)	✓		Generate N outputs, evaluate with reward function, select the best.
MBR (Eikema & Aziz, 2022)		✓	Generate N outputs, evaluate with expected utility function, select the best. (Details in section 4)
RBoN_{KL} (Jinnai et al., 2024)	✓		Maximize the mixture of the reward function and KL divergence with a constraint that the resulting policy is deterministic.
RBoN_{WD} (Jinnai et al., 2024)	✓	✓	Maximize the mixture of the reward function and WD distance with a constraint that the resulting policy is deterministic.
SRBoN_{KL} **(Section 3.1)**	✓		Maximize the mixture of the reward function and KL divergence.
SRBoN_{WD} **(Section 3.2)**	✓	✓	Maximize the mixture of the reward function and WD distance.
RBoN_{L} **(Section 4)**	✓		Consider both the reward function and the token length of the sentence. (Details in section 4 and Appendix G)

Language Model, Reward Model, and Embedding Model. We employ Mistral 7B SFT β (Jiang et al., 2023a) as the language models. We set the maximum entry length and the maximum output length to be 256 tokens. We sample response texts using nucleus sampling (Holtzman et al., 2020) with temperature set to 1.0 and top-p set to 0.9. For each entry, in the AlpacaFarm dataset and Anthropic's datasets, 128 responses are generated using Mistral 7B SFT β.

To evaluate the performance of the algorithms under different preferences, we use OASST (reward-model-deberta-v3-large-v2), SHP-Large (SteamSHP-flan-t5-large), SHP-XL (SteamSHP-flan-t5-xl), PairRM, RM-Mistral-7B and Eurus-RM-7b (Köpf et al., 2023; Ethayarajh et al., 2022; Jiang et al., 2023b; Dong et al., 2023; Yuan et al., 2024) as reward models. For the text embedding model we use all-mpnet-base-v2 (Song et al., 2020), a sentence transformer model (Reimers & Gurevych, 2019) shown to be effective in various sentence embedding and semantic search tasks.

Baselines. The list of text generation methods we evaluate is present in Table 1. The baseline methods include random sampling (nucleus sampling; Holtzman et al. 2020), Best-of-N (BoN) sampling, Minimum Bayes Risk (MBR) decoding, and RBoN_{L}, which we describe in the following.

Minimum Bayes Risk (MBR) decoding (Kumar & Byrne, 2002; 2004; Eikema & Aziz, 2022) is a text generation strategy that selects an output from N outputs that maximizes the expected utility (Berger, 1985). Let a utility function $u(h, y)$ quantify the benefit of choosing $h \in \mathcal{Y}_{\mathbf{ref}}$ if y is the correct output. Then, MBR decoding is defined as follows:

$$y_{\text{MBR}}(x) = \underset{h \in \mathcal{Y}_{\mathbf{ref}}}{\arg\max} \sum_{y \in \mathcal{Y}_{\mathbf{ref}}} \frac{1}{N} u\left(h, y\right). \tag{16}$$

We include MBR decoding as one of the baselines because it has been shown to be effective in a variety of text generation tasks (Suzgun et al., 2023; Bertsch et al., 2023; Li et al., 2024; Heineman et al., 2024). We

follow the implementation of Jinnai et al. (2024) and use the cosine similarity of the sentence embedding as the utility function. We use the same embedding model as the RBoN_{WD}, all-mpnet-base-v2. Note that MBR corresponds to RBoN_{WD} with $u(h, y) = 1 - C(h, y)$ with no reward function or $\beta \to +\infty$ (Eq. (6)) (Jinnai et al., 2024).

As an additional evaluation method, we propose **Sentence Length Regularized BoN** (RBoN_{L}), a simple baseline that adjusts the output token length to the target reward model. In RBoN_{KL} and SRBoN_{KL}, $\pi_{\mathbf{ref}}$ was used for regularization. However, we have observed a bias in language models with respect to sentence length, namely that these models tend to produce shorter sentences with higher probability (Appendix B). To this end, we propose a simple implementation of RBoN that regularizes the generation probability of the sequence token length instead of the generation probability of each sequence. The objective function of RBoN_{L} is given by:

$$y_{\text{RBoN}_{\text{L}}}(x) = \underset{y \in \mathcal{Y}_{\mathbf{ref}}}{\arg\max}\ R(x, y) - \frac{\beta}{|y|}, \tag{17}$$

where β is a regularization parameter and $|y|$ denotes the sequence length (i.e., the number of tokens).

The rationale for this particular form of the regularization term and the experimental details of this approach are described in Appendix G.

4.1 Evaluation of the Algorithms

Setup. We compare the 7 methods using win rates vs. BoN sampling on the evaluation splits of the datasets. Since the RBoN method has a hyperparameter β, we first find the optimal β^* on the train splits. Hyperparameter β range is $\{1.0 \times 10^{-4}, 2.0 \times 10^{-4}, 5.0 \times 10^{-4}, 1.0 \times 10^{-3}, ..., 2.0 \times 10^{1}\}$. We first find the optimal beta value β^* in the train split, then we use the optimal values in the development split for the evaluation split. In this experiment, we use OASST, SHP-Large, SHP-XL, PairRM, and RM-Mistral-7B as proxy reward models. As as the gold reward model, we use Eurus-RM-7B to evaluate the performance of the algorithms. We evaluate the performance of the algorithms as the win rate against BoN sampling according to the reward score of the gold reward model (we count ties as 0.5 wins). We use Eurus-RM-7B as the gold reward model because it is reproducible as it is open source and has been shown to have a high correlation with human preference in RewardBench (Lambert et al., 2024b).

Results. Table 2 reveals several noteworthy results for the AlpacaFarm, Harmlessness, and Helpfulness datasets and the optimal beta β^* is Table 4. The win rate result shows that higher Spearman rank correlation values (Table 3) correspond to better BoN sampling accuracy. This observation is intuitive.

Table 2 shows that the win rate of SRBoN_{KL} is inferior to the deterministic version RBoN_{KL}. While SRBoN_{KL} is proposed as a theoretically robust algorithm (section 3.2.1), its performance in our experiments did not fully meet expectations. One possible factor contributing to this discrepancy could be related to the perturbation range of ΔR. In our experimental setup, it is plausible that the actual perturbations of ΔR may have exceeded the assumed theoretical limits.

Other reasons for suboptimal performance, applicable to both deterministic and stochastic versions, concern the relationship between the reference policy $\pi_{\mathbf{ref}}$ and the reward model. If the correlation between $\pi_{\mathbf{ref}}$ and the reward model is weak, the regularization effect may not contribute positively to the performance of the algorithm (Appendix B).

SRBoN_{WD} shows superior performance across several settings and achieves comparable performance to RBoN_{WD}. This robust performance is remarkable given the low positive correlation between the reference policy $\pi_{\mathbf{ref}}$ and the reward model.

A plausible explanation for this effectiveness, especially in contrast to SRBoN_{KL}, is the constraint on the reward perturbation ΔR in SRBoN_{WD}. Unlike SRBoN_{KL}, the constraint on ΔR in SRBoN_{WD} is independent of $\pi_{\mathbf{ref}}$, which mitigates low performance when there is no correlation between the reward model and $\pi_{\mathbf{ref}}$.

Table 2: The win rate of various methods against BoN sampling.

Method	**OASST**	**SHP-Large**	**SHP-XL**	**PairRM**	**RM-Mistral-7B**
			AlpacaFarm		
BoN	50.0	50.0	50.0	50.0	50.0
MBR	36.0	42.8	40.8	39.1	13.0
Random	20.5	30.3	29.4	27.1	3.0
RBoN_{WD}	50.6	50.2	49.0	**50.7**	49.9
RBoN_{KL}	47.7	26.4	26.2	50.0	48.6
RBoN_{L}	**52.0**	50.3	**50.2**	50.1	**50.8**
SRBoN_{WD}	50.1	**50.6**	49.5	50.0	50.1
SRBoN_{KL}	12.6	20.9	18.7	28.0	4.7
	OASST	**SHP-Large**	**SHP-XL**	**PairRM**	**RM-Mistral-7B**
			Harmlessness		
BoN	50.0	50.0	50.0	50.0	50.0
MBR	40.8	57.4	50.7	42.7	14.8
Random	26.7	52.7	46.3	28.0	7.1
RBoN_{WD}	52.1	**62.2**	**57.1**	50.0	49.9
RBoN_{KL}	48.2	46.9	40.4	50.0	47.4
RBoN_{L}	**52.2**	54.8	54.2	50.0	**51.6**
SRBoN_{WD}	49.7	51.2	49.8	50.0	49.9
SRBoN_{KL}	20.5	42.3	37.1	30.4	5.5
	OASST	**SHP-Large**	**SHP-XL**	**PairRM**	**RM-Mistral-7B**
			Helpfulness		
BoN	50.0	50.0	50.0	50.0	50.0
MBR	41.4	39.2	33.2	40.0	6.1
Random	23.6	23.7	15.1	23.3	0.8
RBoN_{WD}	52.5	**52.4**	50.1	**50.1**	49.9
RBoN_{KL}	44.9	19.9	13.9	50.0	50.0
RBoN_{L}	**52.7**	49.9	**50.8**	50.0	**50.2**
SRBoN_{WD}	50.4	49.5	49.6	50.0	50.0
SRBoN_{KL}	13.4	18.5	11.8	24.3	1.4

Table 3: Spearman's rank correlation between Eurus-RM-7B and each proxy reward. The comprehensive Spearman's rank correlation results for all the aforementioned analyses are presented in Appendix D.

Dataset	**OASST**	**SHP-Large**	**SHP-XL**	**PairRM**	**RM-Mistral-7B**
AlpacaFarm	0.39	0.29	0.35	0.33	0.62
Harmlessness	0.37	0.09	0.14	0.36	0.60
Helpfulness	0.39	0.38	0.50	0.34	0.75

Despite its simple implementation, RBoN_{L} consistently outperformed BoN sampling, achieving a higher win rate on almost all tasks and models with no instances of underperformance. A detailed discussion of RBoN_{L} is presented in Appendix G.

4.2 RBoN Sensitiveness of Parameters

Setup. In this section, we evaluate the generalization performance of the model using β. values $\{1.0\times10^{-4}$, 2.0×10^{-4}, 5.0×10^{-4},1.0×10^{-3},..., $2.0\times10^{1}\}$ to the evaluation splits. We also use several models as proxy reward models, including OASST, SHP-Large, SHP-XL, PairRM, and RM-Mistral-7B. As a gold reward model, we use Eurus-RM-7B to evaluate the performance of the proxy models. The results are visualized as

Table 4: Optimal beta β^* in the train split

Method	**OASST**	**SHP-Large**	**SHP-XL**	**PairRM**	**RM-Mistral-7B**
			AlpacaFarm		
RBoN_{WD}	20	0.5	0.5	20	0.1
RBoN_{KL}	0.0001	0.0001	0.0001	0.0001	0.0001
RBoN_{L}	20	0.5	0.2	20	15.0
SRBoN_{WD}	0.5	0.0002	0.0001	0.0001	1.0
SRBoN_{KL}	20	0.05	0.05	20	20
			Harmlessness		
RBoN_{WD}	20	1.0	1.0	0.0001	5.0
RBoN_{KL}	0.0001	0.0001	0.0001	0.0001	0.0001
RBoN_{L}	20	5.0	5.0	0.0001	20
SRBoN_{WD}	0.05	0.0001	0.0001	0.0001	0.02
SRBoN_{KL}	20	0.05	20	20	20
			Helpfulness		
RBoN_{WD}	15.0	0.05	0.1	20	0.5
RBoN_{KL}	0.0001	0.0001	0.0001	0.0001	0.0001
RBoN_{L}	20	0.02	0.2	5.0	20
SRBoN_{WD}	0.5	0.001	0.005	5.0	0.0002
SRBoN_{KL}	20	0.05	20	20	20

a plot showing the win rates of each method compared to BoN sampling on the evaluation splits. We assign 1 point for a win and 0.5 points for a tie.

Results The performance result of RBoN method in AlpacaFarm is illustrated in Figures 1. This result reveals that the optimal parameters for the RBoN_{WD} and SRBoN_{WD} method vary between different models and reveals the performance of SRBoN_{WD} across various problem settings, as the value of the regularization parameter β increases, we observe a degradation performance. Intuitively, upon examining the adversarial formulation of SRBoN_{WD}, we can infer that as the regularization parameter β increases, the magnitude of potential perturbations ΔR also increases. Furthermore, as evidenced in Table 4, the optimal β value for SRBoN_{WD} is typically smaller than that for RBoN_{WD}.

This result shows that SRBoN_{KL} consistently underperforms within the β range examined in our experiments. In particular, as shown in Table 4, the optimal regularization parameter β^* for SRBoN_{KL} is often found to be $\beta^* = 20$ across different problem settings. This observation leads to an intriguing hypothesis, that the performance of SRBoN_{KL} could potentially improve with higher values of β.

The performance result of RBoN_{L} demonstrates superior performance across a wide range of β values, exhibiting performance characteristics comparable to RBoN_{WD}. Notably, this robust performance across varying β values indicates that RBoN_{L} exhibits low sensitivity to changes in the regularization parameter.

The results for Harmlessness and Helpfulness datasets are presented in Appendix C.

5 Related Work

Robust MDPs Several studies have investigated RL considering the worst-case scenario for rewards. Ortega & Lee (2014) considers only a single-step analysis for the reward robust problem. Husain et al. (2021) proposes a deep RL algorithm related to Q learning for the reward robust problem. Derman et al. (2021) considers both a reward function and the transition probability as unknown. The policy regularization is considered a perturbation of the rewards, while the transition probability perturbations address the worst-case scenario with respect to the associated set of value functions. They define specific uncertainty sets and conduct thorough experiments. Eysenbach & Levine (2022) shows that incorporating the policy's Shannon

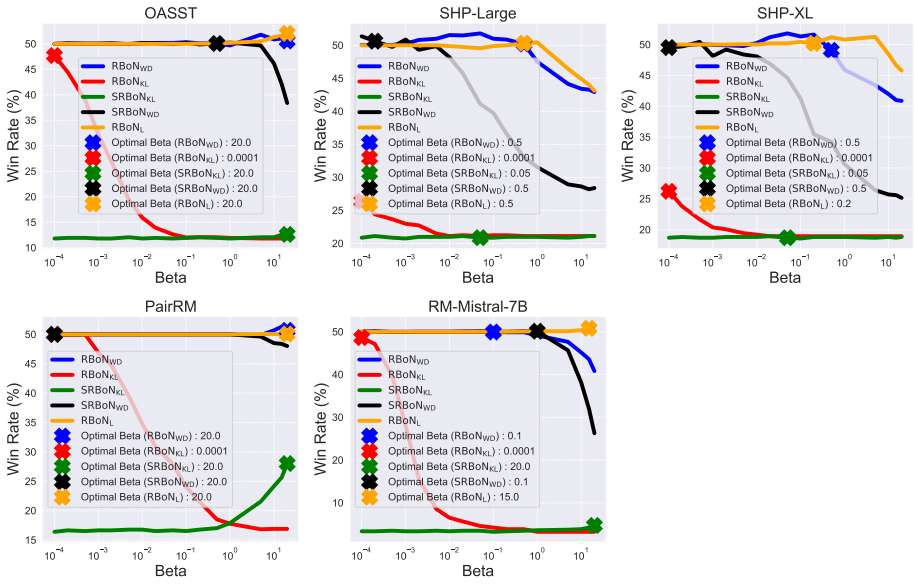

Figure 1: Evaluation of RBoN sensitiveness on the AlpacaFarm dataset with varying parameter β. We use proxy reward models, OASST, SHP-Large, SHP-XL, PairRM, and RM-Mistral-7B. As the gold reward model, we utilize Eurus-RM-7B.

entropy into the reinforcement learning objective function represents the worst-case scenario for a given uncertainty set of rewards.

Alignment Strategies Two notable adaptation strategies have recently gained attention: Reinforcement Learning from Human Feedback (RLHF) and Direct Preference Optimization (DPO) (Stiennon et al., 2020; Rafailov et al., 2023). RLHF incorporates human feedback into the reinforcement learning process to align the agent's behavior with human preferences. By using human feedback as a reward signal, RLHF aims to optimize it. This approach has been successfully used in LLM (Ouyang et al., 2022). On the other hand, DPO uses the same objective function as RLHF without an explicit reward function. However, it still suffers more than RLHF from overoptimization when dealing with out-of-distribution data (Xu et al., 2024). Beyond the methods discussed, there is research in robust optimization that addresses the development of robust algorithms for scenarios with unstable preference information (Wu et al., 2024). In particular, Chowdhury et al. (2024) have introduced a robust DPO approach that achieves robustness without explicitly employing robust optimization techniques. Another technique (Mudgal et al., 2024) is to train a token-level scoring value function module to select the optimal output. Khanov et al. (2024) is a novel decoding method that does not require additional learning and uses both the language model and the reward model knowledge. There is a parameter that determines which is more important, and depending on its value, it can be a conventional method.

6 Conclusions

This paper introduces three novel BoN sampling methods: SRBoN_{KL}, SRBoN_{WD}, and RBoN_{L}. To rigorously evaluate the effectiveness of these proposed methods, we conducted extensive experiments using two datasets: AlpacaFarm and Anthropic's hh-rlhf.

The SRBoN_{KL} and SRBoN_{WD} methods extend the previous RBoN_{KL} and RBoN_{WD} methods, respectively. In particular, SRBoN_{KL} and SRBoN_{WD} produce a stochastic optimal policy that differs from their deter-

ministic counterparts. The theoretical guarantees of their robustness increase the reliability of the methods in different scenarios.

The RBoN_{L} method is a contribution to the field of RBoN sampling, providing a simple yet effective approach. Despite its simplicity, our experiments show that RBoN_{L} achieves performance comparable to the more complex RBoN_{WD}. This finding highlights the potential of RBoN_{L} as a computationally efficient alternative to more complicated methods, making it particularly attractive for applications with limited resources or stringent performance requirements.

In conclusion, this paper presents three innovative BoN sampling methods that significantly contribute to the field. The experimental results and theoretical guarantees underscore the effectiveness and reliability of these methods. Our work lays the foundation for further research and applications of robust BoN sampling techniques in a wide range of domains.

7 Limitations

While our proposed method demonstrates promising results, there are several limitations to note. The proposed method requires no fine-tuning of the LLMs but inevitably increases computational overhead at inference. In contrast, fine-tuning approaches incur a one-time cost during training while eliminating overhead at inference. Another concern is that the proposed method considers a max-min problem, so if, for example, the correlation between the proxy reward and the gold reward is strong, performance is reduced due to conservative output selection.

Our study lacks an analysis of whether the reward perturbations satisfy the conditions outlined in Theorems 3.1 and 3.2. Evaluating the error of the reward and utility function in experiments remains an area for future work. Additionally, the selection of the parameter β requires a validation set in the current setting, and developing an automated method to determine β is a promising direction for further research.

Furthermore, our approach relies on a specific utility function, which is a prerequisite for applying the proposed method, and the method does not account for process reward models, which may limit its applicability in some scenarios. It is also worth noting that the experiments conducted in this study were limited to three English datasets, leaving open the question of its generalizability to other languages or domains.

In addition, the proposed method is based on a probabilistic framework, which, while effective for uncertainty, may not align with real-world applications where deterministic versions (RBoN) are often preferred for their predictability and safety. Based on the analysis in this paper, the analysis of the deterministic RBoN is a possible direction for future work.

Finally, while the current formulation is specific, the proposed method has the potential to be extended to other divergence measures, such as f-divergences, offering an exciting avenue for future investigation.

Acknowledgments

We sincerely thank the Action Editor, Pascal Poupart, and the anonymous reviewers for their insightful comments and suggestions. Kaito Ariu's research is supported by JSPS KAKENHI Grant No. 23K19986.

References

Dario Amodei, Chris Olah, Jacob Steinhardt, Paul Christiano, John Schulman, and Dan Mané. Concrete Problems in AI Safety. *arXiv preprint arXiv:1606.06565*, 2016.

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

Ahmad Beirami, Alekh Agarwal, Jonathan Berant, Alexander D'Amour, Jacob Eisenstein, Chirag Nagpal, and Ananda Theertha Suresh. Theoretical guarantees on the best-of-n alignment policy. *arXiv preprint arXiv:2401.01879*, 2024.

James O Berger. *Statistical decision theory and Bayesian analysis; 2nd ed.* Springer series in statistics. Springer, New York, 1985. doi: 10.1007/978-1-4757-4286-2. URL https://cds.cern.ch/record/1327974.

Amanda Bertsch, Alex Xie, Graham Neubig, and Matthew Gormley. It's MBR all the way down: Modern generation techniques through the lens of minimum Bayes risk. In Yanai Elazar, Allyson Ettinger, Nora Kassner, Sebastian Ruder, and Noah A. Smith (eds.), *Proceedings of the Big Picture Workshop*, pp. 108–122, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.bigpicture-1.9. URL https://aclanthology.org/2023.bigpicture-1.9.

Stephen Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

Rob Brekelmans, Tim Genewein, Jordi Grau-Moya, Gregoire Detetang, Markus Kunesch, Shane Legg, and Pedro A Ortega. Your Policy Regularizer is Secretly an Adversary. *Transactions on Machine Learning Research*, 2022. ISSN 2835-8856. URL https://openreview.net/forum?id=berNQMTYWZ. Expert Certification.

Stephen Casper, Xander Davies, Claudia Shi, Thomas Krendl Gilbert, Jérémy Scheurer, Javier Rando, Rachel Freedman, Tomasz Korbak, David Lindner, Pedro Freire, Tony Tong Wang, Samuel Marks, Charbel-Raphael Segerie, Micah Carroll, Andi Peng, Phillip Christoffersen, Mehul Damani, Stewart Slocum, Usman Anwar, Anand Siththaranjan, Max Nadeau, Eric J Michaud, Jacob Pfau, Dmitrii Krasheninnikov, Xin Chen, Lauro Langosco, Peter Hase, Erdem Biyik, Anca Dragan, David Krueger, Dorsa Sadigh, and Dylan Hadfield-Menell. Open Problems and Fundamental Limitations of Reinforcement Learning from Human Feedback. *Transactions on Machine Learning Research*, 2023. ISSN 2835-8856. URL https://openreview.net/forum?id=bx24KpJ4Eb. Survey Certification.

Sayak Ray Chowdhury, Anush Kini, and Nagarajan Natarajan. Provably Robust DPO: Aligning Language Models with Noisy Feedback. In *Forty-first International Conference on Machine Learning*, 2024. URL https://openreview.net/forum?id=yhpDKSw7yA.

Josef Dai, Xuehai Pan, Ruiyang Sun, Jiaming Ji, Xinbo Xu, Mickel Liu, Yizhou Wang, and Yaodong Yang. Safe RLHF: Safe reinforcement learning from human feedback. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=TyFrPOKYXw.

Esther Derman, Matthieu Geist, and Shie Mannor. Twice regularized MDPs and the equivalence between robustness and regularization. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, volume 34, pp. 22274–22287. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/file/bb1443cc31d7396bf73e7858cea114e1-Paper.pdf.

Shizhe Diao, Rui Pan, Hanze Dong, KaShun Shum, Jipeng Zhang, Wei Xiong, and Tong Zhang. LMFlow: An Extensible Toolkit for Finetuning and Inference of Large Foundation Models. In Kai-Wei Chang, Annie Lee, and Nazneen Rajani (eds.), *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 3: System Demonstrations)*, pp. 116–127, Mexico City, Mexico, June 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.naacl-demo.12. URL https://aclanthology.org/2024.naacl-demo.12.

Hanze Dong, Wei Xiong, Deepanshu Goyal, Yihan Zhang, Winnie Chow, Rui Pan, Shizhe Diao, Jipeng Zhang, KaShun SHUM, and Tong Zhang. RAFT: Reward rAnked Finetuning for Generative Foundation Model Alignment. *Transactions on Machine Learning Research*, 2023. ISSN 2835-8856. URL https://openreview.net/forum?id=m7p5O7zblY.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Yann Dubois, Chen Xuechen Li, Rohan Taori, Tianyi Zhang, Ishaan Gulrajani, Jimmy Ba, Carlos Guestrin, Percy S Liang, and Tatsunori B Hashimoto. Alpacafarm: A Simulation Framework for Methods that

Learn from Human Feedback. In A. Oh, T. Neumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 30039–30069. Curran Associates, Inc., 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/5fc47800ee5b30b8777fdd30abcaaf3b-Paper-Conference.pdf.

Bryan Eikema and Wilker Aziz. Sampling-Based Approximations to Minimum Bayes Risk Decoding for Neural Machine Translation. In Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang (eds.), *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pp. 10978–10993, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.emnlp-main.754. URL https://aclanthology.org/2022.emnlp-main.754.

Kawin Ethayarajh, Yejin Choi, and Swabha Swayamdipta. Understanding Dataset Difficulty with $\mathcal{V}$-usable Information. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato (eds.), *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 5988–6008. PMLR, 17–23 Jul 2022. URL https://proceedings.mlr.press/v162/ethayarajh22a.html.

Benjamin Eysenbach and Sergey Levine. MaximumEntropy RL (provably) Solves Some Robust RL Problems. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=PtSAD3caaA2.

Leo Gao, John Schulman, and Jacob Hilton. Scaling Laws for Reward Model Overoptimization. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (eds.), *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 10835–10866. PMLR, 23–29 Jul 2023. URL https://proceedings.mlr.press/v202/gao23h.html.

Matthieu Geist, Bruno Scherrer, and Olivier Pietquin. A Theory of Regularized Markov Decision Processes. In Kamalika Chaudhuri and Ruslan Salakhutdinov (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 2160–2169. PMLR, 09–15 Jun 2019. URL https://proceedings.mlr.press/v97/geist19a.html.

Tianrui Guan, Fuxiao Liu, Xiyang Wu, Ruiqi Xian, Zongxia Li, Xiaoyu Liu, Xijun Wang, Lichang Chen, Furong Huang, Yaser Yacoob, Dinesh Manocha, and Tianyi Zhou. Hallusionbench: An Advanced Diagnostic Suite for Entangled Language Hallucination and Visual Illusion in Large Vision-Language Models. *arXiv preprint arXiv:2310.14566*, 2024.

Lin Gui, Cristina Gârbacea, and Victor Veitch. BoNBoN Alignment for Large Language Models and the Sweetness of Best-of-n Sampling. *arXiv preprint arXiv:2406.00832*, 2024.

David Heineman, Yao Dou, and Wei Xu. Improving Minimum Bayes Risk Decoding with Multi-Prompt. *arXiv preprint arXiv:2407.15343*, 2024.

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. The Curious Case of Neural Text Degeneration. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=rygGQyrFvH.

Hisham Husain, Kamil Ciosek, and Ryota Tomioka. Regularized policies are reward robust. In *International Conference on Artificial Intelligence and Statistics*, pp. 64–72. PMLR, 2021.

Hamish Ivison, Yizhong Wang, Jiacheng Liu, Zeqiu Wu, Valentina Pyatkin, Nathan Lambert, Noah A Smith, Yejin Choi, and Hannaneh Hajishirzi. Unpacking DPO and PPO: Disentangling Best Practices for Learning from Preference Feedback. *arXiv preprint arXiv:2406.09279*, 2024.

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. Mistral 7B. *arXiv preprint arXiv:2310.06825*, 2023a.

Dongfu Jiang, Xiang Ren, and Bill Yuchen Lin. LLM-Blender: Ensembling Large Language Models with Pairwise Ranking and Generative Fusion. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki (eds.), *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 14165–14178, Toronto, Canada, July 2023b. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-long.792. URL https://aclanthology.org/2023.acl-long.792.

Yuu Jinnai, Tetsuro Morimura, Kaito Ariu, and Kenshi Abe. Regularized best-of-n sampling to mitigate reward hacking for language model alignment. In *ICML 2024 Workshop on Models of Human Feedback for AI Alignment*, 2024. URL https://openreview.net/forum?id=ewRlZPAReR.

Maxim Khanov, Jirayu Burapacheep, and Yixuan Li. ARGS: Alignment as Reward-Guided Search. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=shgx0eqdw6.

Andreas Köpf, Yannic Kilcher, Dimitri von Rütte, Sotiris Anagnostidis, Zhi Rui Tam, Keith Stevens, Abdullah Barhoum, Duc Nguyen, Oliver Stanley, Richárd Nagyfi, Shahul ES, Sameer Suri, David Glushkov, Arnav Dantuluri, Andrew Maguire, Christoph Schuhmann, Huu Nguyen, and Alexander Mattick. Openassistant conversations - democratizing large language model alignment. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 47669–47681. Curran Associates, Inc., 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/949f0f8f32267d297c2d4e3ee10a2e7e-Paper-Datasets_and_Benchmarks.pdf.

Shankar Kumar and William Byrne. Minimum Bayes-risk word alignments of bilingual texts. In *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP 2002)*, pp. 140–147. Association for Computational Linguistics, July 2002. doi: 10.3115/1118693.1118712. URL https://aclanthology.org/W02-1019.

Shankar Kumar and William Byrne. Minimum Bayes-risk decoding for statistical machine translation. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics: HLT-NAACL 2004*, pp. 169–176, Boston, Massachusetts, USA, May 2 - May 7 2004. Association for Computational Linguistics. URL https://aclanthology.org/N04-1022.

Matt Kusner, Yu Sun, Nicholas Kolkin, and Kilian Weinberger. From word embeddings to document distances. In Francis Bach and David Blei (eds.), *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pp. 957–966, Lille, France, 07–09 Jul 2015. PMLR. URL https://proceedings.mlr.press/v37/kusnerb15.html.

Nathan Lambert and Roberto Calandra. The Alignment Ceiling: Objective Mismatch in Reinforcement Learning from Human Feedback. *arXiv preprint arXiv:2311.00168*, 2024.

Nathan Lambert, Valentina Pyatkin, Jacob Morrison, LJ Miranda, Bill Yuchen Lin, Khyathi Chandu, Nouha Dziri, Sachin Kumar, Tom Zick, Yejin Choi, Noah A. Smith, and Hannaneh Hajishirzi. Rewardbench: Evaluating reward models for language modeling. https://huggingface.co/spaces/allenai/reward-bench, 2024a.

Nathan Lambert, Valentina Pyatkin, Jacob Morrison, LJ Miranda, Bill Yuchen Lin, Khyathi Chandu, Nouha Dziri, Sachin Kumar, Tom Zick, Yejin Choi, Noah A. Smith, and Hannaneh Hajishirzi. RewardBench: Evaluating Reward Models for Language Modeling. *arXiv preprint arXiv:2403.13787*, 2024b.

Junyou Li, Qin Zhang, Yangbin Yu, Qiang Fu, and Deheng Ye. More Agents Is All You Need. *arXiv preprint arXiv:2402.05120*, 2024.

Stephanie Lin, Jacob Hilton, and Owain Evans. TruthfulQA: Measuring how models mimic human falsehoods. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 3214–3252, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-long.229. URL https://aclanthology.org/2022.acl-long.229.

Sidharth Mudgal, Jong Lee, Harish Ganapathy, YaGuang Li, Tao Wang, Yanping Huang, Zhifeng Chen, Heng-Tze Cheng, Michael Collins, Trevor Strohman, Jilin Chen, Alex Beutel, and Ahmad Beirami. Controlled Decoding from Language Models. In *Forty-first International Conference on Machine Learning*, 2024. URL https://openreview.net/forum?id=bVIcZb7Qa0.

Reiichiro Nakano, Jacob Hilton, Suchir Balaji, Jeff Wu, Long Ouyang, Christina Kim, Christopher Hesse, Shantanu Jain, Vineet Kosaraju, William Saunders, Xu Jiang, Karl Cobbe, Tyna Eloundou, Gretchen Krueger, Kevin Button, Matthew Knight, Benjamin Chess, and John Schulman. Webgpt: Browser-assisted question-answering with human feedback. *arXiv preprint arXiv:2112.09332*, 2022.

Gergely Neu, Anders Jonsson, and Vicenç Gómez. A unified view of entropy-regularized markov decision processes. *arXiv preprint arXiv:1705.07798*, 2017.

OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin, Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, Mohammad Bavarian, Jeff Belgum, Irwan Bello, Jake Berdine, Gabriel Bernadett-Shapiro, Christopher Berner, Lenny Bogdonoff, Oleg Boiko, Madelaine Boyd, Anna-Luisa Brakman, Greg Brockman, Tim Brooks, Miles Brundage, Kevin Button, Trevor Cai, Rosie Campbell, Andrew Cann, Brittany Carey, Chelsea Carlson, Rory Carmichael, Brooke Chan, Che Chang, Fotis Chantzis, Derek Chen, Sully Chen, Ruby Chen, Jason Chen, Mark Chen, Ben Chess, Chester Cho, Casey Chu, Hyung Won Chung, Dave Cummings, Jeremiah Currier, Yunxing Dai, Cory Decareaux, Thomas Degry, Noah Deutsch, Damien Deville, Arka Dhar, David Dohan, Steve Dowling, Sheila Dunning, Adrien Ecoffet, Atty Eleti, Tyna Eloundou, David Farhi, Liam Fedus, Niko Felix, Simón Posada Fishman, Juston Forte, Isabella Fulford, Leo Gao, Elie Georges, Christian Gibson, Vik Goel, Tarun Gogineni, Gabriel Goh, Rapha Gontijo-Lopes, Jonathan Gordon, Morgan Grafstein, Scott Gray, Ryan Greene, Joshua Gross, Shixiang Shane Gu, Yufei Guo, Chris Hallacy, Jesse Han, Jeff Harris, Yuchen He, Mike Heaton, Johannes Heidecke, Chris Hesse, Alan Hickey, Wade Hickey, Peter Hoeschele, Brandon Houghton, Kenny Hsu, Shengli Hu, Xin Hu, Joost Huizinga, Shantanu Jain, Shawn Jain, Joanne Jang, Angela Jiang, Roger Jiang, Haozhun Jin, Denny Jin, Shino Jomoto, Billie Jonn, Heewoo Jun, Tomer Kaftan, Łukasz Kaiser, Ali Kamali, Ingmar Kanitscheider, Nitish Shirish Keskar, Tabarak Khan, Logan Kilpatrick, Jong Wook Kim, Christina Kim, Yongjik Kim, Jan Hendrik Kirchner, Jamie Kiros, Matt Knight, Daniel Kokotajlo, Łukasz Kondraciuk, Andrew Kondrich, Aris Konstantinidis, Kyle Kosic, Gretchen Krueger, Vishal Kuo, Michael Lampe, Ikai Lan, Teddy Lee, Jan Leike, Jade Leung, Daniel Levy, Chak Ming Li, Rachel Lim, Molly Lin, Stephanie Lin, Mateusz Litwin, Theresa Lopez, Ryan Lowe, Patricia Lue, Anna Makanju, Kim Malfacini, Sam Manning, Todor Markov, Yaniv Markovski, Bianca Martin, Katie Mayer, Andrew Mayne, Bob McGrew, Scott Mayer McKinney, Christine McLeavey, Paul McMillan, Jake McNeil, David Medina, Aalok Mehta, Jacob Menick, Luke Metz, Andrey Mishchenko, Pamela Mishkin, Vinnie Monaco, Evan Morikawa, Daniel Mossing, Tong Mu, Mira Murati, Oleg Murk, David Mély, Ashvin Nair, Reiichiro Nakano, Rajeev Nayak, Arvind Neelakantan, Richard Ngo, Hyeonwoo Noh, Long Ouyang, Cullen O'Keefe, Jakub Pachocki, Alex Paino, Joe Palermo, Ashley Pantuliano, Giambattista Parascandolo, Joel Parish, Emy Parparita, Alex Passos, Mikhail Pavlov, Andrew Peng, Adam Perelman, Filipe de Avila Belbute Peres, Michael Petrov, Henrique Ponde de Oliveira Pinto, Michael, Pokorny, Michelle Pokrass, Vitchyr H. Pong, Tolly Powell, Alethea Power, Boris Power, Elizabeth Proehl, Raul Puri, Alec Radford, Jack Rae, Aditya Ramesh, Cameron Raymond, Francis Real, Kendra Rimbach, Carl Ross, Bob Rotsted, Henri Roussez, Nick Ryder, Mario Saltarelli, Ted Sanders, Shibani Santurkar, Girish Sastry, Heather Schmidt, David Schnurr, John Schulman, Daniel Selsam, Kyla Sheppard, Toki Sherbakov, Jessica Shieh, Sarah Shoker, Pranav Shyam, Szymon Sidor, Eric Sigler, Maddie Simens, Jordan Sitkin, Katarina Slama, Ian Sohl, Benjamin Sokolowsky, Yang Song, Natalie Staudacher, Felipe Petroski Such, Natalie Summers, Ilya Sutskever, Jie Tang, Nikolas Tezak, Madeleine B. Thompson, Phil Tillet, Amin Tootoonchian, Elizabeth Tseng, Preston Tuggle, Nick Turley, Jerry Tworek, Juan Felipe Cerón Uribe, Andrea Vallone, Arun Vijayvergiya, Chelsea Voss, Carroll Wainwright, Justin Jay Wang, Alvin Wang, Ben Wang, Jonathan Ward, Jason Wei, CJ Weinmann, Akila Welihinda, Peter Welinder, Jiayi Weng, Lilian Weng, Matt Wiethoff, Dave Willner, Clemens Winter, Samuel Wolrich, Hannah Wong, Lauren Workman, Sherwin Wu, Jeff Wu, Michael Wu, Kai Xiao, Tao Xu, Sarah Yoo, Kevin Yu, Qiming Yuan, Wojciech Zaremba, Rowan Zellers, Chong Zhang,

Marvin Zhang, Shengjia Zhao, Tianhao Zheng, Juntang Zhuang, William Zhuk, and Barret Zoph. GPT-4 technical report. *arXiv preprint arXiv:2303.08774*, 2024.

Pedro Ortega and Daniel Lee. An adversarial interpretation of information-theoretic bounded rationality. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 28, 2014.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 27730–27744. Curran Associates, Inc., 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/b1efde53be364a73914f58805a001731-Paper-Conference.pdf.

Alexander Pan, Kush Bhatia, and Jacob Steinhardt. The Effects of Reward Misspecification: Mapping and Mitigating Misaligned Models. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=JYtwGwIL7ye.

Matteo Pirotta, Marcello Restelli, and Luca Bascetta. Policy gradient in Lipschitz Markov Decision Processes. *Machine Learning*, 100:255 – 283, 2015. URL https://api.semanticscholar.org/CorpusID:254741544.

Emmanuel Rachelson and Michail G. Lagoudakis. On the locality of action domination in sequential decision making. In *International Symposium on Artificial Intelligence and Mathematics*, 2010. URL https://api.semanticscholar.org/CorpusID:14029770.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct Preference Optimization: Your Language Model is Secretly a Reward Model. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 53728–53741. Curran Associates, Inc., 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/a85b405ed65c6477a4fe8302b5e06ce7-Paper-Conference.pdf.

Nils Reimers and Iryna Gurevych. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. 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)*, pp. 3982–3992, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1410. URL https://aclanthology.org/D19-1410.

Joar Skalse, Nikolaus Howe, Dmitrii Krasheninnikov, and David Krueger. Defining and Characterizing Reward Gaming. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 9460–9471. Curran Associates, Inc., 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/3d719fee332caa23d5038b8a90e81796-Paper-Conference.pdf.

Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. Mpnet: Masked and Permuted Pre-training for Language Understanding. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 16857–16867. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/c3a690be93aa602ee2dc0ccab5b7b67e-Paper.pdf.

Charles Spearman. The Proof and Measurement of Association between Two Things. *The American Journal of Psychology*, 15(1), 1904.

Nisan Stiennon, Long Ouyang, Jeffrey Wu, Daniel Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul F Christiano. Learning to summarize with human feedback. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 3008–3021. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/1f89885d556929e98d3ef9b86448f951-Paper.pdf.

Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*. MIT press, 2018.

Mirac Suzgun, Luke Melas-Kyriazi, and Dan Jurafsky. Follow the wisdom of the crowd: Effective text generation via minimum Bayes risk decoding. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki (eds.), *Findings of the Association for Computational Linguistics: ACL 2023*, pp. 4265–4293, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-acl.262. URL https://aclanthology.org/2023.findings-acl.262.

Qwen Team. Qwen2.5: A party of foundation models, September 2024. URL https://qwenlm.github.io/blog/qwen2.5/.

Hugo Touchette. Legendre-Fenchel transforms in a nutshell. *URL http://www. maths. qmul. ac. uk/˜ ht/archive/lfth2. pdf*, 2005.

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

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

R.J. Vanderbei. *Linear Programming: Foundations and Extensions*. International Series in Operations Research & Management Science. Springer International Publishing, 2020. ISBN 9783030394158. URL https://books.google.co.jp/books?id=6yjfDwAAQBAJ.

Feng-Yu Wang. Coupling and applications. In *Stochastic Analysis and Applications to Finance: Essays in Honour of Jia-An Yan*, pp. 411–424. World Scientific, 2012.

Junkang Wu, Yuexiang Xie, Zhengyi Yang, Jiancan Wu, Jiawei Chen, Jinyang Gao, Bolin Ding, Xiang Wang, and Xiangnan He. Towards Robust Alignment of Language Models: Distributionally Robustifying Direct Preference Optimization. *arXiv preprint arXiv:2407.07880*, 2024.

Shusheng Xu, Wei Fu, Jiaxuan Gao, Wenjie Ye, Weilin Liu, Zhiyu Mei, Guangju Wang, Chao Yu, and Yi Wu. Is DPO superior to PPO for LLM Alignment? A Comprehensive Study. In *Forty-first International Conference on Machine Learning*, 2024. URL https://openreview.net/forum?id=6XH8R7YrSk.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jialong Tang, Jialin Wang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Ma, Jin Xu, Jingren Zhou, Jinze Bai, Jinzheng He, Junyang Lin, Kai Dang, Keming Lu, Keqin Chen, Kexin Yang, Mei Li, Mingfeng Xue, Na Ni, Pei Zhang, Peng Wang, Ru Peng, Rui Men, Ruize Gao, Runji Lin, Shijie Wang, Shuai Bai, Sinan Tan, Tianhang Zhu, Tianhao Li, Tianyu Liu, Wenbin Ge, Xiaodong Deng, Xiaohuan Zhou, Xingzhang Ren, Xinyu Zhang, Xipin Wei, Xuancheng Ren, Yang Fan, Yang Yao, Yichang Zhang, Yu Wan, Yunfei Chu, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zhihao Fan. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*, 2024.

Wenhao Yang, Xiang Li, and Zhihua Zhang. A Regularized Approach to Sparse Optimal Policy in Reinforcement Learning. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc,

E. Fox, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper_files/paper/2019/file/3f4366aeb9c157cf9a30c90693eafc55-Paper.pdf.

Lifan Yuan, Ganqu Cui, Hanbin Wang, Ning Ding, Xingyao Wang, Jia Deng, Boji Shan, Huimin Chen, Ruobing Xie, Yankai Lin, Zhenghao Liu, Bowen Zhou, Hao Peng, Zhiyuan Liu, and Maosong Sun. Advancing LLM Reasoning Generalists with Preference Trees. In *AI for Math Workshop @ ICML 2024*, 2024. URL https://openreview.net/forum?id=2Y1iiCqM5y.

Wei Zhao, Maxime Peyrard, Fei Liu, Yang Gao, Christian M. Meyer, and Steffen Eger. MoverScore: Text generation evaluating with contextualized embeddings and earth mover distance. 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)*, pp. 563–578, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1053. URL https://aclanthology.org/D19-1053.

Daniel M. Ziegler, Nisan Stiennon, Jeffrey Wu, Tom B. Brown, Alec Radford, Dario Amodei, Paul Christiano, and Geoffrey Irving. Fine-tuning language models from human preferences. *arXiv preprint arXiv:1909.08593*, 2020.

A Detailed proof of Theorem 3.2

Definition A.1 (Similarity-based Lipschitz Continuity)**.** *A function f is said to have Similarity-based Lipschitz Continuity if, for any $y, y' \in \mathcal{Y}$, the following holds:*

$$|f(y) - f(y')| \leq C(y, y')$$

where

$$C(y, y') = 1 - \cos\left(\text{emb}(y), \text{emb}(y')\right)$$

We first explain how the objective function is reformulated to a max-min problem. Let us focus on the regularization term, 1-**WD** term rewrite related to π, $\pi_{\mathbf{ref}}$

The following analysis is done in the framework of finite probability spaces. To simplify the following proof, we introduce the following notation. Let $x_1, x_2, \cdots, x_n$ be n places and consider the function f, where f_i refers to the value $f(x_i)$.

$$\begin{aligned} \mathbf{WD}[\nu \| \mu] &= \min_{\gamma \in \Gamma(\nu,\mu)} \sum_{(i,j) \in \mathcal{Y} \times \mathcal{Y}} C_{ij} \gamma_{ij} \\ &= \min_{\gamma \in \mathbb{R}^{Y \times Y'}} \sum_{(i,j) \in \mathcal{Y} \times \mathcal{Y}} C_{ij} \gamma_{ij} + \Psi(\gamma), \end{aligned} \tag{18}$$

where γ is a coupling of the probability measure ν and μ, $\Gamma(\nu, \mu) = \left\{\gamma \in \mathbb{R}^{Y \times Y'} | \sum_{j \in \mathcal{Y}} \gamma_{ij} = \nu_i, \sum_{i \in \mathcal{Y}}, \gamma_{ij} = \mu_j, \gamma_{ij} \geq 0 \text{ for all } i, j\right\}$, Y and Y' $(= \mathcal{Y})$ is sample space respectively, corresponding to outcomes i and j respectively and $\Psi(\gamma) = 0$ if $\gamma \in \Gamma(\nu, \mu), +\infty$ otherwise.

Constraint terms, a coupling of the probability measure γ needs to satisfy:

$$\begin{aligned} \sum_j \gamma_{ij} &= \nu_i \quad \forall i \in \mathcal{Y} \\ \sum_i \gamma_{ij} &= \mu_j \quad \forall j \in \mathcal{Y} \\ \gamma_{ij} &\geq 0 \quad \forall i, j \in \mathcal{Y} \end{aligned} \tag{19}$$

This constraint can be expressed in $\mathbf{A}\gamma = \mathbf{b}$, indicating its linear nature. Specifically, $\mathbf{A}$ and $\mathbf{b}$ are defined as $\mathbf{A} = \binom{I_i \otimes \mathbf{1}_j^\top}{I_j \otimes \mathbf{1}_i^\top}$, $\mathbf{b} = \binom{\nu}{\mu}$. In this formulation, I_i and I_j denote identity matrices of dimension $\mathcal{Y} \times \mathcal{Y}$, while $\mathbf{1}_i$ and $\mathbf{1}_j$ represent column vectors of dimension $\mathcal{Y}$ with all components equal to 1. The symbol $\otimes$ denotes the Kronecker product.

Lemma A.2. *Eq. (18) is reformulated as a max problem from a min problem.*

$$\max_{\substack{f \\ |f_i - f_j| \leq C_{ij}}} \sum_i f_i \nu_i - \sum_j f_j \mu_j \tag{20}$$

Proof. Taking into account the constraints specified in Eq. (19), we proceed with the application of the Lagrange multiplier method:

$$\mathbf{WD}[\nu \| \mu] = \min_{\gamma \in \mathbb{R}^{Y \times Y'}} \sum_{i,j} C_{ij} \gamma_{ij} + \max_{f,g} \{\sum_i f_i \nu_i + \sum_j g_j \mu_j - \sum_{i,j} (f_i + g_j) \gamma_{ij}\}$$

For a more intuitive understanding, f and g can be considered analogous to Lagrange multipliers. Except for the first term, all subsequent entries refer to constraints on γ.

$$\mathbf{WD}[\nu \| \mu] = \min_{\gamma \in \mathbb{R}^{Y \times Y'}} \max_{f,g} \sum_{i,j} (C_{ij} - f_i - g_j) \gamma_{ij} + \sum_i f_i \nu_i + \sum_j g_j \mu_j$$

can be seen from Eq. (19), these constraints are linear. From Theorem 5.2 (Vanderbei, 2020), in linear programming, there is never a gap between the primal and the dual optimal objective values. Under the strong duality theorem (e.g., $\min_x \max_y f(x,y) = \max_y \min_x f(x,y)$), we can exchange the min max term.

$$\mathbf{WD}[\nu\|\mu] = \max_{f,g} \min_{\gamma \in \mathbb{R}^{Y \times Y'}} \sum_{i,j} (C_{ij} - f_i - g_j)\gamma_{ij} + \sum_i f_i \nu_i + \sum_j g_j \mu_j$$

If $C_{ij} - f_i - g_j \geq 0$ for all i, j, the optimal value of $\min_\gamma \sum_{i,j} (C_{ij} - f_i - g_j)\gamma_{ij}$ is 0, otherwise ∞. This observation allows us to derive the inequality constraint for the first item. We can include this as a constraint in the equation:

$$\mathbf{WD}[\nu\|\mu] = \max_{\substack{f,g \\ f_i + g_j \leq C_{ij}}} \sum_i f_i \nu_i + \sum_j g_j \mu_j$$

Our next goal is to express the above function, currently represented by f and g, exclusively in terms of the function f. From the given constraints, we have established that $f_i + g_j \leq C_{ij}$ for all i and j. We can express this as follows:

$$g_j \leq \min_i \{C_{ij} - f_i\} \tag{21}$$

To fix $i = i^*$, since $\min_i$ picks the minimum value. The index i^* gives this minimum, and fixing i to i^* turns the inequality in Eq. (21) into the equality in Eq. (22).

$$g_j = \{C_{i^* j} - f_{i^*}\} \tag{22}$$

Eq. (22) gives us a function which is called the c-transform of f_j and is often denoted by f_j^c,

$$f_j^c = g_j = \{C_{i^* j} - f_{i^*}\}$$

We can now rewrite **WD** with f_j^c as

$$\mathbf{WD}[\nu\|\mu] = \max_f \sum_i f_i \nu_i + \sum_j f_j^c \mu_j \tag{23}$$

If f is similarity-based Lipschitz, f^c is also similarity-based Lipschitz, for all $\boldsymbol{i}$ and $\boldsymbol{j}$ we have

$$\begin{aligned} &\left|f_j^c - f_i^c\right| \leq C_{ij} \\ &\implies -C_{ij} \leq f_j^c - f_i^c \leq C_{ij} \\ &\implies -f_i^c \leq C_{ij} - f_j^c \end{aligned}$$

$$\implies -f_i^c \leq \min_j \left\{C_{ij} - f_j^c\right\}$$

Upper bound of $\min_j \left\{C_{ij} - f_j^c\right\}$ is choosing $j \to i$

$$\min_j \left\{C_{ij} - f_j^c\right\} \leq -f_i^c$$

It can be shown that $f_i^{cc} = f_i = \min_j \left\{C_{ij} - f_j^c\right\}$. This means that $-g = -f^c = f$. Substituting $f_j^c = -f_j$ into Eq. 23, we get

$$\max_{\substack{f \\ |f_i - f_j| \leq C_{ij}}} \sum_i f_i \nu_i - \sum_j f_j \mu_j \tag{24}$$

which is the dual form of 1-Wasserstein distance.

□

Finally, by substituting ΔR for f, we get:

$$\begin{aligned}
\pi_{\text{SRBoN}_{\text{WD}}}(x) &= \max_{\pi \in \Pi} \mathbb{E}_{y \sim \pi(\cdot|x)}[R(x,y)] - \Omega(\pi) \\
&= \max_{\pi \in \Pi} \mathbb{E}_{y \sim \pi(\cdot|x)}[R(x,y)] - \max_{\Delta R \in \mathcal{R}_\Delta} \beta \left(\sum_{\mathcal{Y}_{\mathbf{ref}}} \Delta R(x,y) \pi_{\mathbf{ref}}(y \mid x) - \sum_{\mathcal{Y}_{\mathbf{ref}}} \Delta R(x,y) \pi(y \mid x) \right) \\
&= \max_{\pi \in \Pi} \mathbb{E}_{y \sim \pi(\cdot|x)}[R(x,y)] - \min_{\Delta R \in \mathcal{R}_\Delta} \beta \left(- \sum_{\mathcal{Y}_{\mathbf{ref}}} \Delta R(x,y) \pi_{\mathbf{ref}}(y \mid x) + \sum_{\mathcal{Y}_{\mathbf{ref}}} \Delta R(x,y) \pi(y \mid x) \right)
\end{aligned}$$

where $\Omega(\pi) = \beta \mathbf{WD}[\pi_{\mathbf{ref}}(\cdot \mid x) \| \pi(\cdot \mid x)]$.

$$\pi_{\text{SRBoN}_{\text{WD}}}(x) = \max_{\pi \in \Pi} \min_{\Delta R \in \mathcal{R}_\Delta} \mathbb{E}_{y \sim \pi(\cdot|x)} \left[R(x,y) - \beta \Delta R(x,y) \right] + \beta \sum_{y \in \mathcal{Y}_{\mathbf{ref}}} \pi_{\mathbf{ref}}(y \mid x) \Delta R(x,y)$$

$$\text{where} \quad \mathcal{R}_\Delta := \left\{ \Delta R \in \mathbb{R}^{\mathcal{X} \times \mathcal{Y}_{\mathbf{ref}}} \mid |\Delta R(x,y) - \Delta R(x,y')| \leq C(y,y') \quad \forall y, y' \in \mathcal{Y}_{\mathbf{ref}} \right\}$$

B Relationship Between $\pi_{\mathbf{ref}}$ and the Proxy Reward Model

Despite the theoretical robustness of SRBoN_{KL} demonstrated in the analyses presented in section 3.1.1, the experimental results (section 4.1 and section 4.2) did not show comparable robustness. This section aims to explain the reasons for this discrepancy. Recall the objective function of SRBoN_{KL}:

$$\begin{aligned}\pi_{\text{SRBoN}_{\text{KL}}}(x) &= \max_{\pi} \mathbb{E}_{y\sim\pi(\cdot|x)}[R(x,y)] - \Omega(\pi) \\ &= \max_{\pi} \mathbb{E}_{y\sim\pi(\cdot|x)}[R(x,y)] - \sum_{\mathcal{Y}_{\mathbf{ref}}} \pi(y \mid x) \log \frac{\pi(y \mid x)}{\pi_{\mathbf{ref}}(y \mid x)}\end{aligned}$$

This implies that ideally, $\pi_{\mathbf{ref}}$ and the reward function R should have some form of relationship (e.g. positive correlation) that facilitates learning. However, $\pi_{\mathbf{ref}}$ is influenced by complex factors such as length bias.

To verify this hypothesis, we examine two aspects: (1) the correlation between the Eurus-RM-7B reward values, which were used as the gold reward model in our experiments, and the probabilities assigned by $\pi_{\mathbf{ref}}$; (2) the relationship between the length of the outputs generated by $\pi_{\mathbf{ref}}$ and the generation probabilities of those outputs.

Table 5: The correlation between the Eurus-RM-7B reward values and the probabilities assigned by $\pi_{\mathbf{ref}}$

AlpacaFarm	**Harmlessness**	**Helpfulness**
−0.224	0.088	−0.097

Table 6: The relationship between the length of the outputs generated by $\pi_{\mathbf{ref}}$ and the generation probabilities of these outputs.

AlpacaFarm	**Harmlessness**	**Helpfulness**
−0.877	−0.924	−0.854

As can be seen from Table 5, there is negligible correlation between $\pi_{\mathbf{ref}}$ and Eurus-RM-7B (gold reward model) in terms of Harmlessness and Helpfulness. In addition, the domain of the AlpacaFarm dataset tends to be negatively correlated.

These results explain the performance degradation observed when this relationship is included in the regularization term. Table 6 shows that $\pi_{\mathbf{ref}}$ has a bias towards shorter sentences, with output probabilities increasing as sentence length decreases.

C Supplemently Results

Figures 2 and 3 show evaluation of RBoN sensitivity on the Harmlessness subset and Helpfulness of the hh-rlhf dataset. These results were similar to those seen in AlpacaFarm using section 4.2. This means that each method is not necessarily dependent on the dataset.

Figures 4 to 6 compare RBoN_{WD} and SRBoN_{WD} and Figures 7 to 9 compare RBoN_{KL} and SRBoN_{KL}. These results show that SRBoN is not superior to RBoN. This is for reasons also discussed in section 4

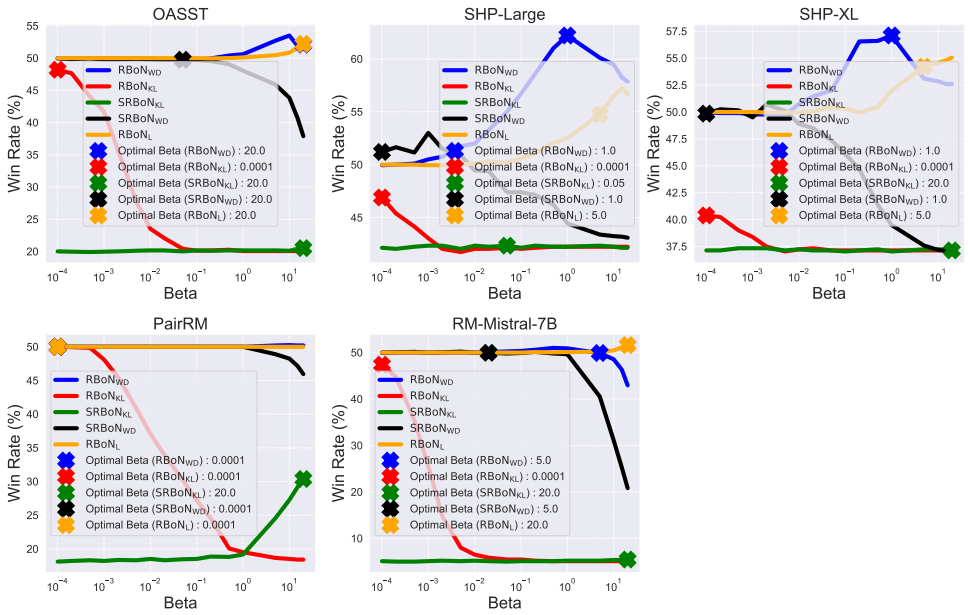

Figure 2: Evaluation of RBoN sensitiveness on the Harmlessness subset of the hh-rlhf dataset with varying parameter β. We use proxy reward models, OASST, SHP-Large, SHP-XL, PairRM, and RM-Mistral-7B. As the gold reward model, we utilize Eurus-RM-7B.

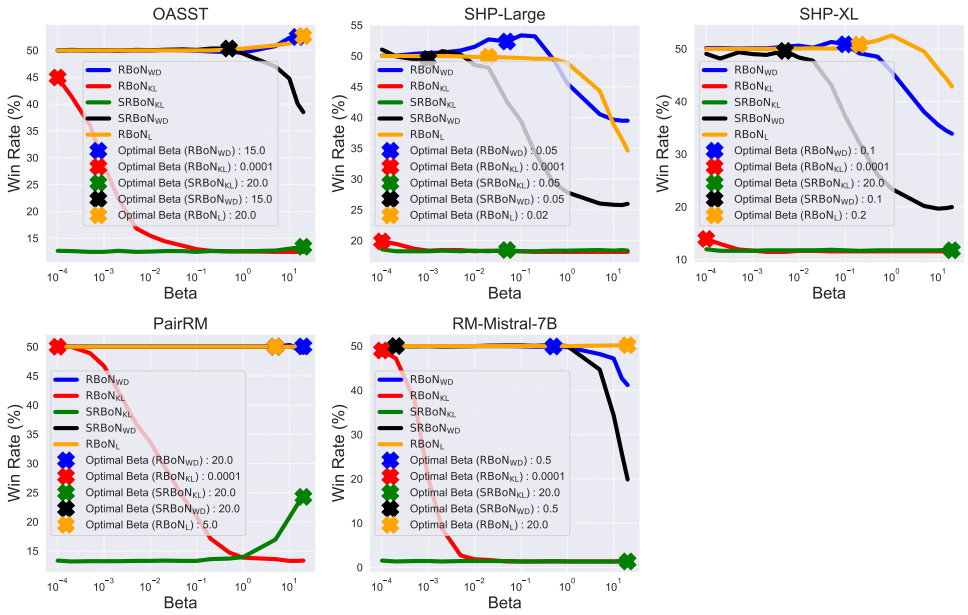

Figure 3: Evaluation of RBoN sensitiveness on the Helpfulness subset of the hh-rlhf dataset with varying parameter β. We use proxy reward models, OASST, SHP-Large, SHP-XL, PairRM, and RM-Mistral-7B. As the gold reward model, we utilize Eurus-RM-7B.

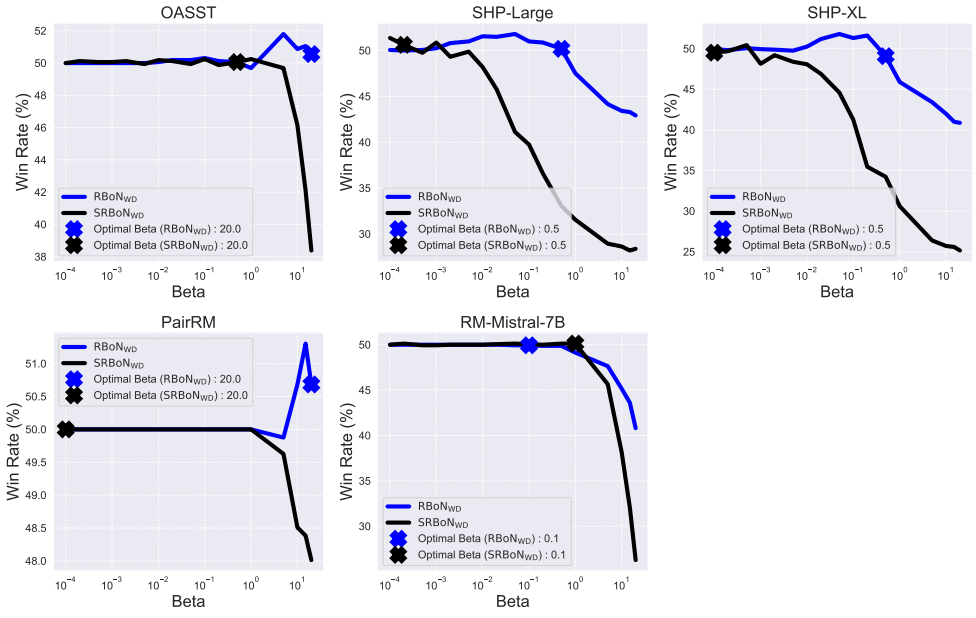

Figure 4: Evaluation of RBoN_{WD} and SRBoN_{WD} sensitiveness on the AlpacaFarm dataset with varying parameter β. We use proxy reward models, OASST, SHP-Large, SHP-XL, PairRM, and RM-Mistral-7B. As the gold reward model, we utilize Eurus-RM-7B.

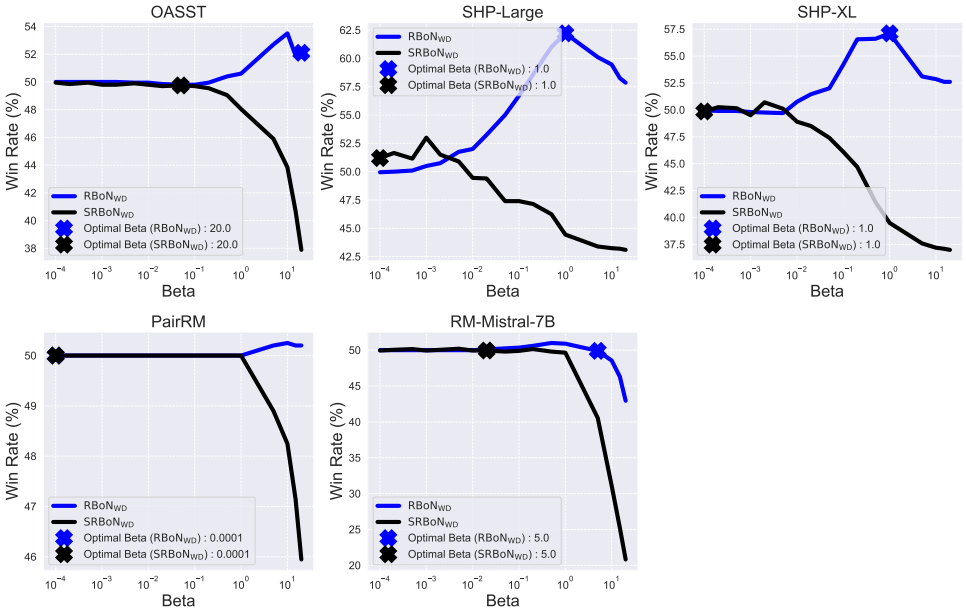

Figure 5: Evaluation of RBoN_{WD} and SRBoN_{WD} sensitiveness on the Harmlessness subset of the hh-rlhf dataset with varying parameter β. We use proxy reward models, OASST, SHP-Large, SHP-XL, PairRM, and RM-Mistral-7B. As the gold reward model, we utilize Eurus-RM-7B.

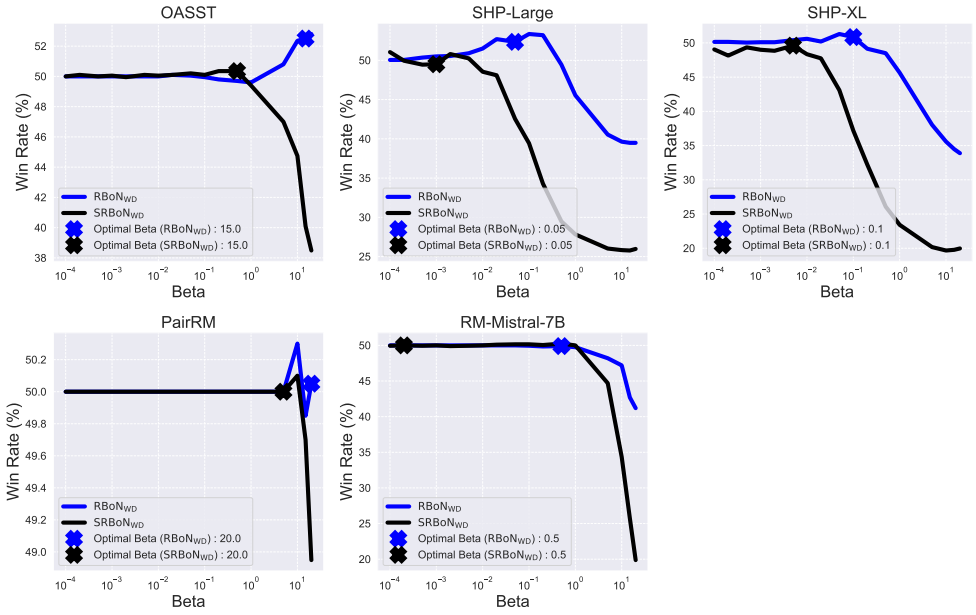

Figure 6: Evaluation of RBoN_{WD} and SRBoN_{WD} sensitiveness on the Helpfulness subset of the hh-rlhf dataset with varying parameter β. We use proxy reward models, OASST, SHP-Large, SHP-XL, PairRM, and RM-Mistral-7B. As the gold reward model, we utilize Eurus-RM-7B.

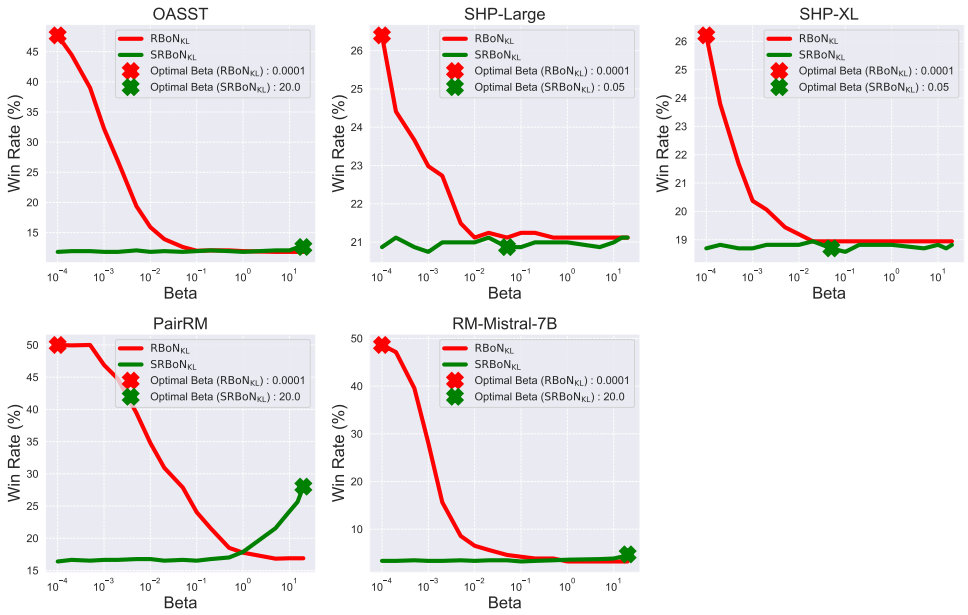

Figure 7: Evaluation of RBoN_{KL} and SRBoN_{KL} sensitiveness on the AlpacaFarm dataset with varying parameter β. We use proxy reward models, OASST, SHP-Large, SHP-XL, PairRM, and RM-Mistral-7B. As the gold reward model, we utilize Eurus-RM-7B.

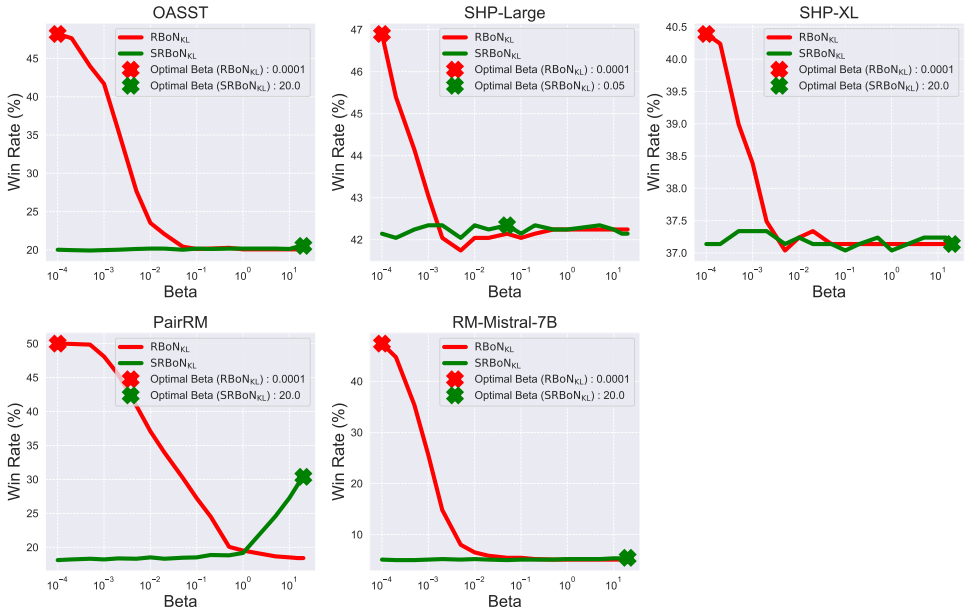

Figure 8: Evaluation of RBoN_{KL} and SRBoN_{KL} sensitiveness on the Harmlessness subset of the hh-rlhf dataset with varying parameter β. We use proxy reward models, OASST, SHP-Large, SHP-XL, PairRM, and RM-Mistral-7B. As the gold reward model, we utilize Eurus-RM-7B.

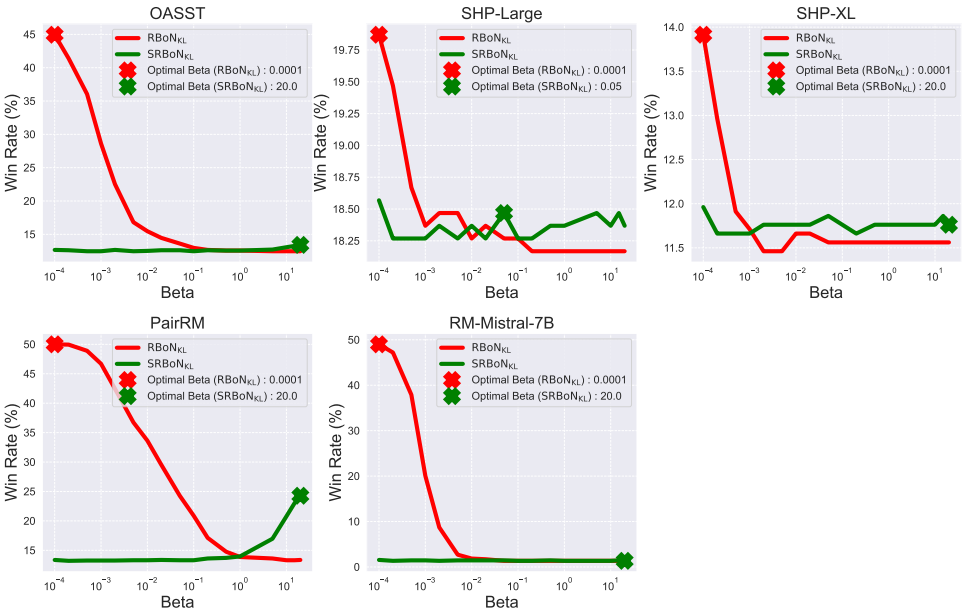

Figure 9: Evaluation of RBoN_{KL} and SRBoN_{KL} sensitiveness on the Helpfulness subset of the hh-rlhf dataset with varying parameter β. We use proxy reward models, OASST, SHP-Large, SHP-XL, PairRM, and RM-Mistral-7B. As the gold reward model, we utilize Eurus-RM-7B.

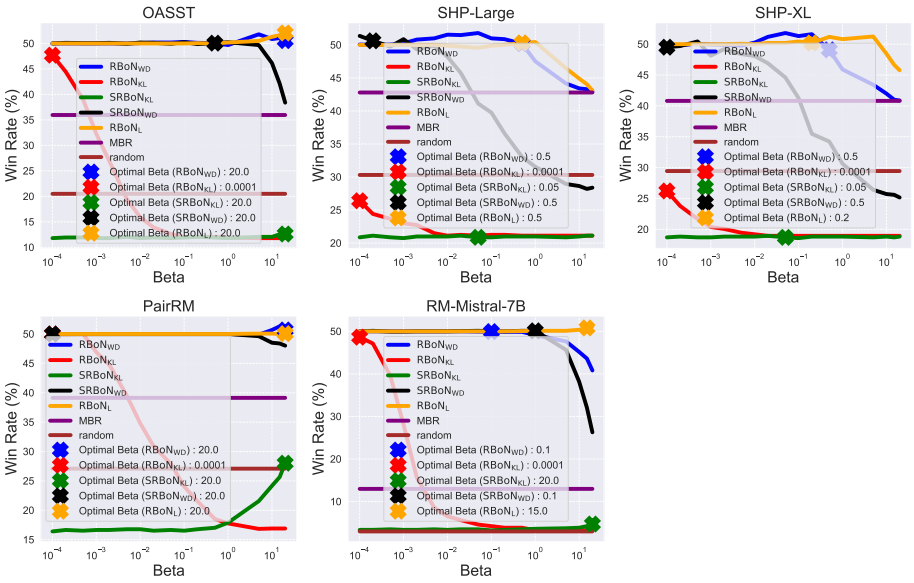

Figure 10: Evaluation of the decoder method on the AlpacaFarm dataset with varying parameter β. We use proxy reward models, OASST, SHP-Large, SHP-XL, PairRM, and RM-Mistral-7B. As the gold reward model, we utilize Eurus-RM-7B.

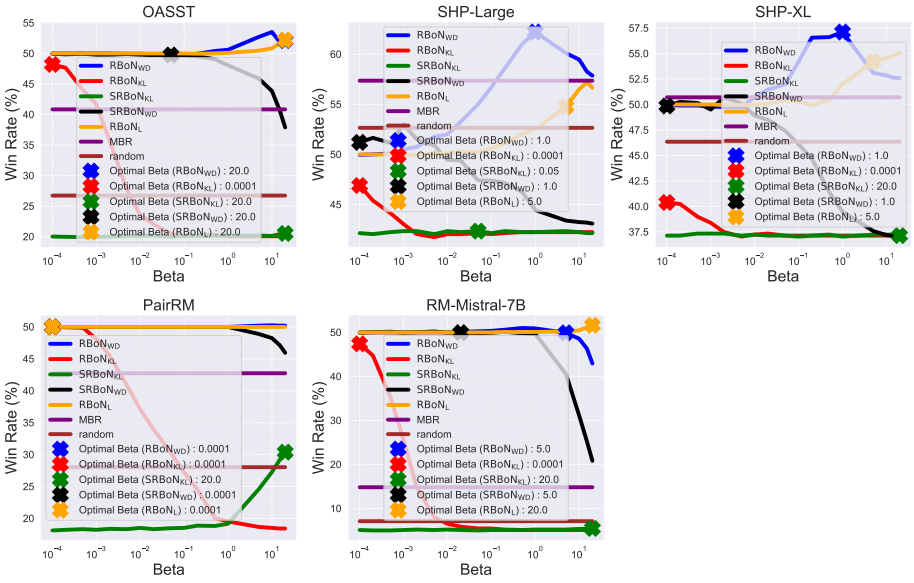

Figure 11: Evaluation of the decoder method on the Harmlessness dataset with varying parameter β. We use proxy reward models, OASST, SHP-Large, SHP-XL, PairRM, and RM-Mistral-7B. As the gold reward model, we utilize Eurus-RM-7B.

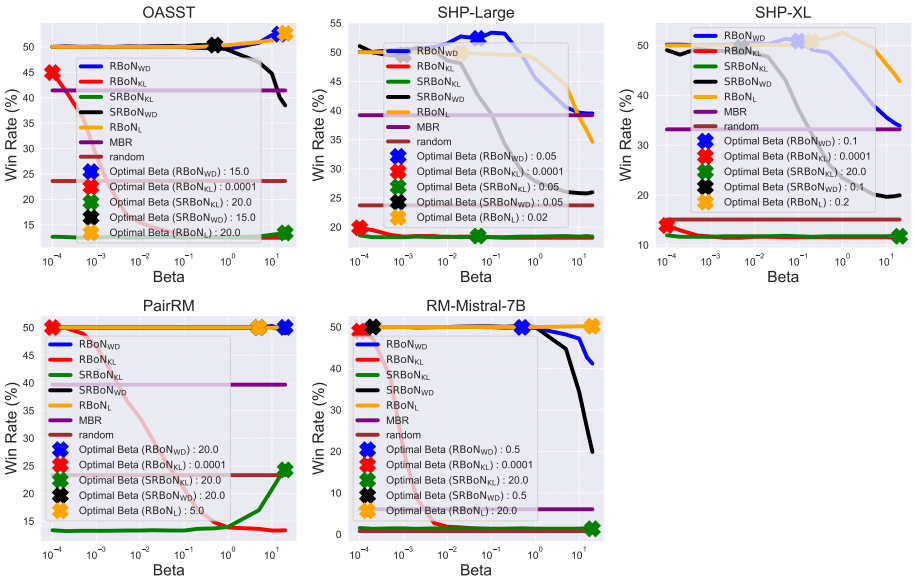

Figure 12: Evaluation of the decoder method on the Helpfulness dataset with varying parameter β. We use proxy reward models, OASST, SHP-Large, SHP-XL, PairRM, and RM-Mistral-7B. As the gold reward model, we utilize Eurus-RM-7B.

D Spearman's Rank Correlation (Spearman, 1904)

Figure 13, Figure 14, and Figure 15 show the average Spearman's rank correlation coefficient (ρ) between pairs of reward models (Spearman, 1904). These results suggest that pairs of reward models with higher correlation values are more similar, indicating a preference for greedy methods in such cases.

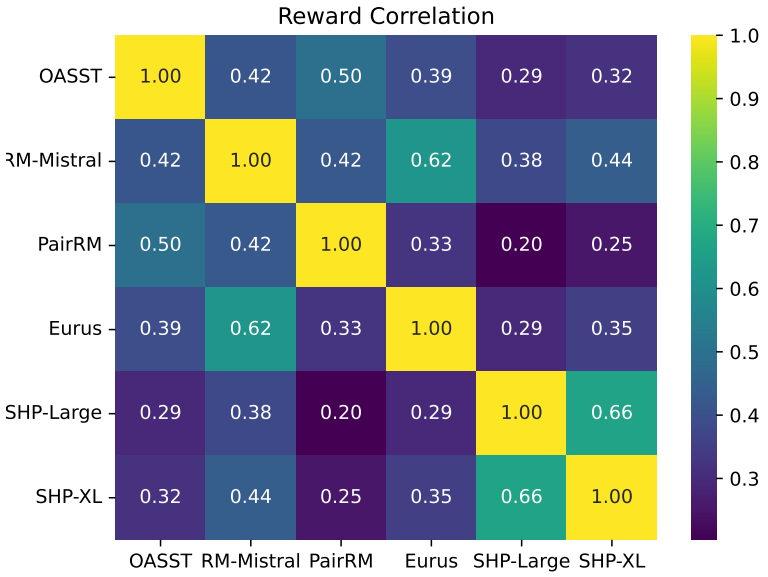

Figure 13: The average Spearman's rank correlation coefficient (ρ) between pairs of reward models in the AlpacaFarm dataset.

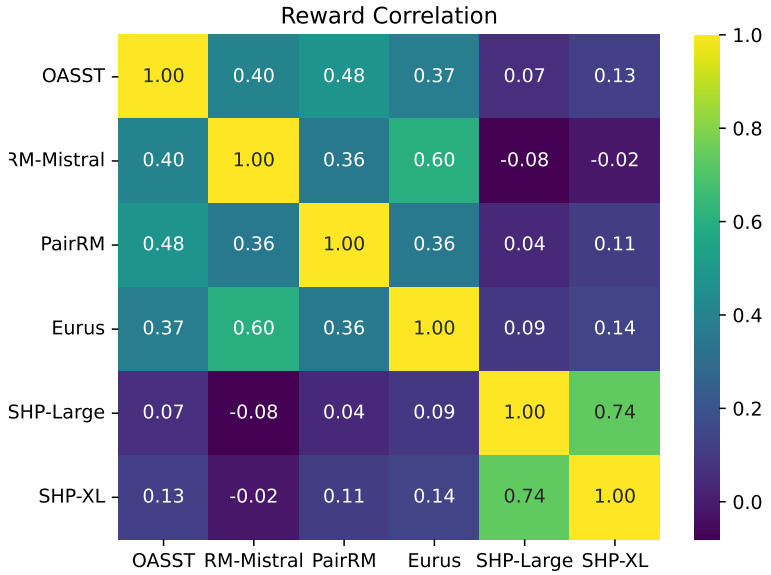

Figure 14: The average Spearman's rank correlation coefficient (ρ) between pairs of reward models in the Harmlessness dataset.

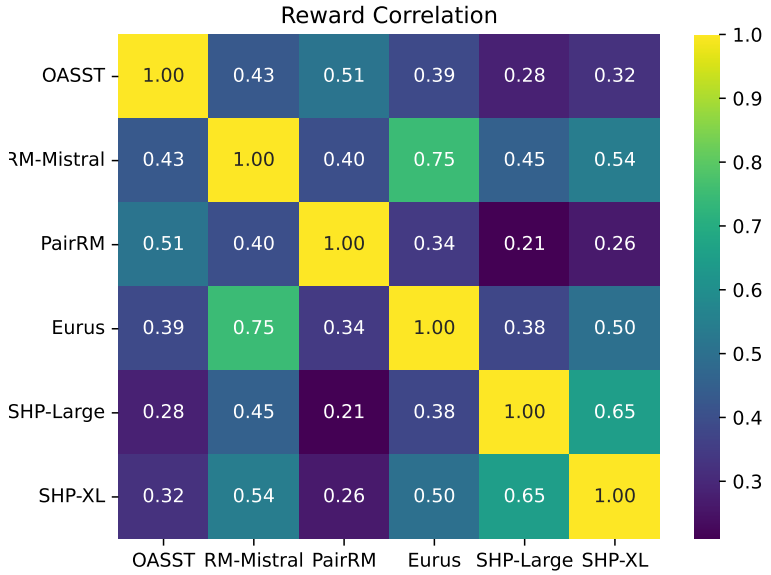

Figure 15: The average Spearman's rank correlation coefficient (ρ) between pairs of reward models in the Helpfulness dataset.

E Supplementary Result on Meta-Llama-3-8B-Instruct (Dubey et al., 2024)

We compared the average Spearman's rank correlation coefficient of the reward model and the performance of RBoN_{WD} on the evaluation split using the Llama (Meta-Llama-3-8B-Instruct) language model. The purpose of this analysis is to verify the performance of RBoN_{WD}, even when applied to samples generated by state-of-the-art language models.

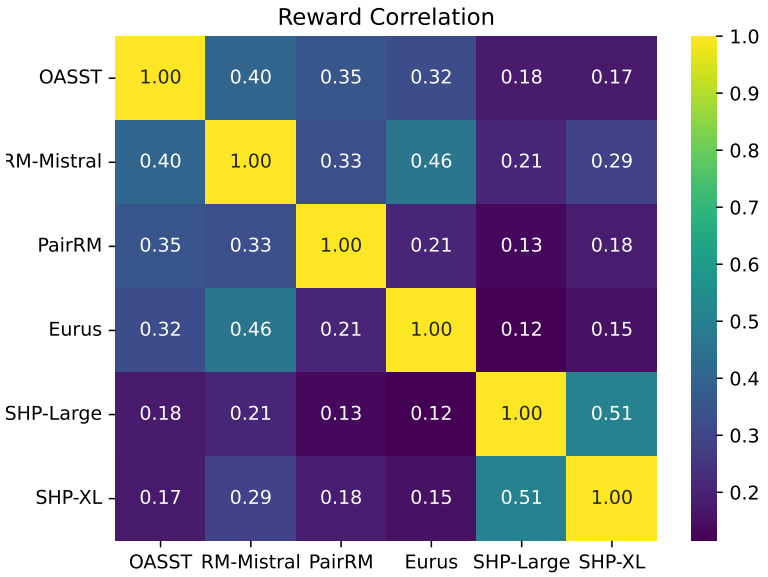

Figure 16: The average Spearman's rank correlation coefficient (ρ) between pairs of reward models in the AlpacaFarm dataset, using Llama as the language model.

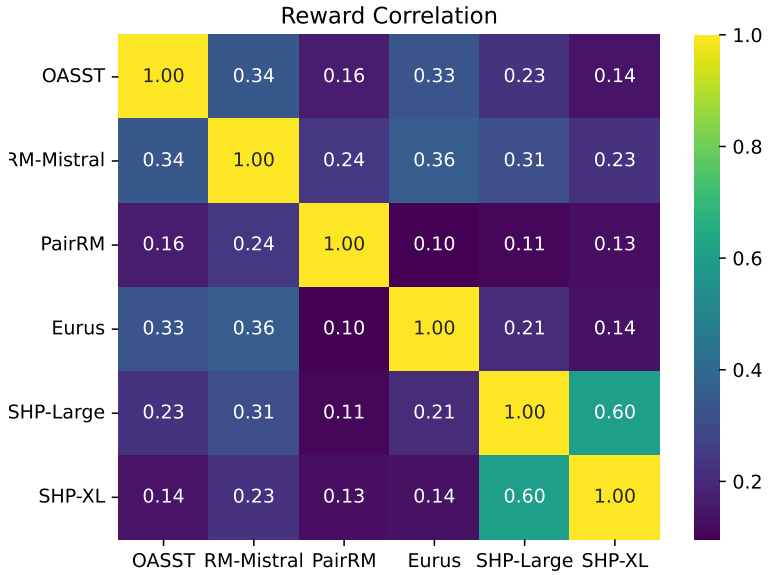

Figure 17: The average Spearman's rank correlation coefficient (ρ) between pairs of reward models in the Harmlessness dataset, using Llama as the language model.

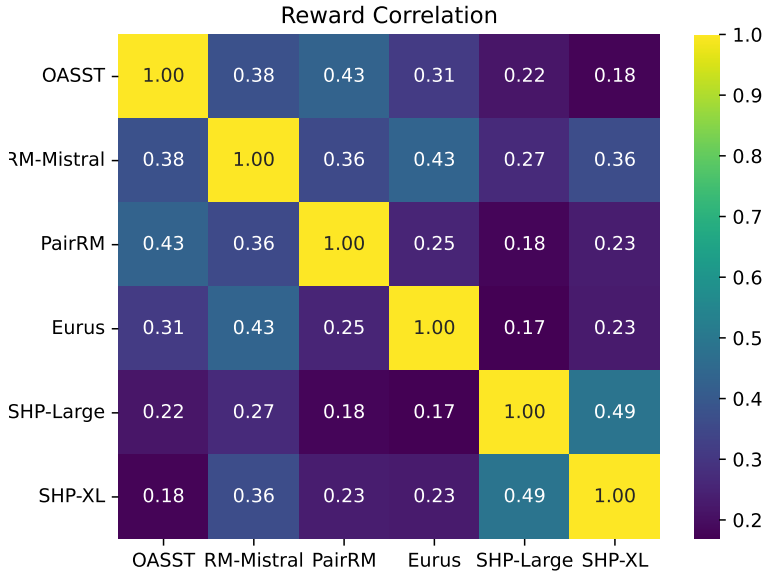

Figure 18: The average Spearman's rank correlation coefficient (ρ) between pairs of reward models in the Helpfulness dataset, using Llama as the language model.

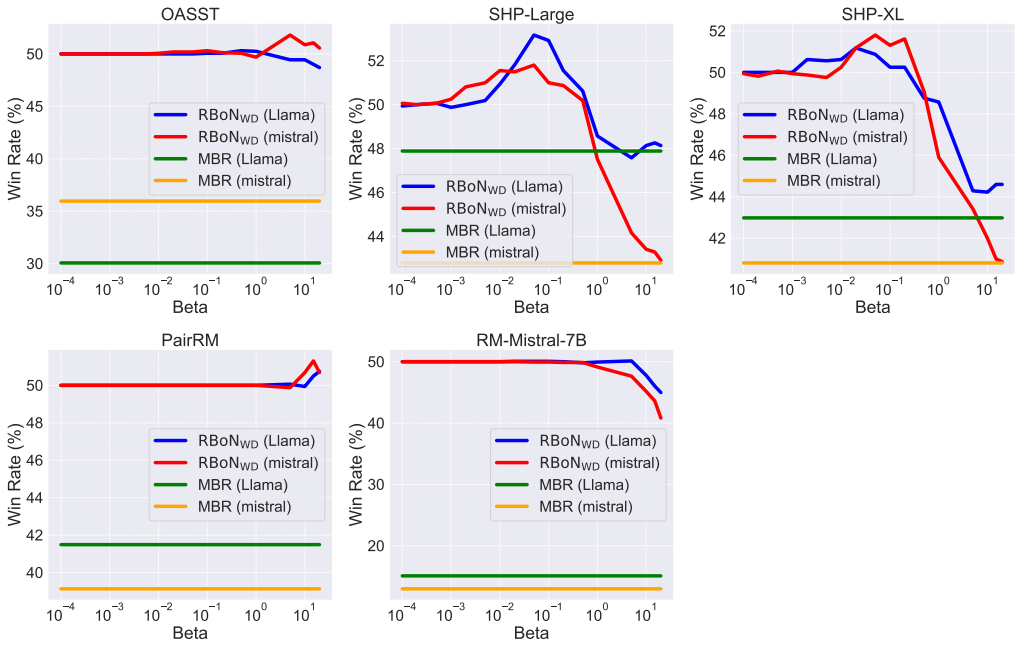

Figure 19: Evaluation of the RBoN method on the AlpacaFarm dataset with varying parameter β. We use proxy reward models, OASST, SHP-Large, SHP-XL, PairRM, and RM-Mistral-7B. As the gold reward model, we utilize Eurus-RM-7B, and Llama as the language model.

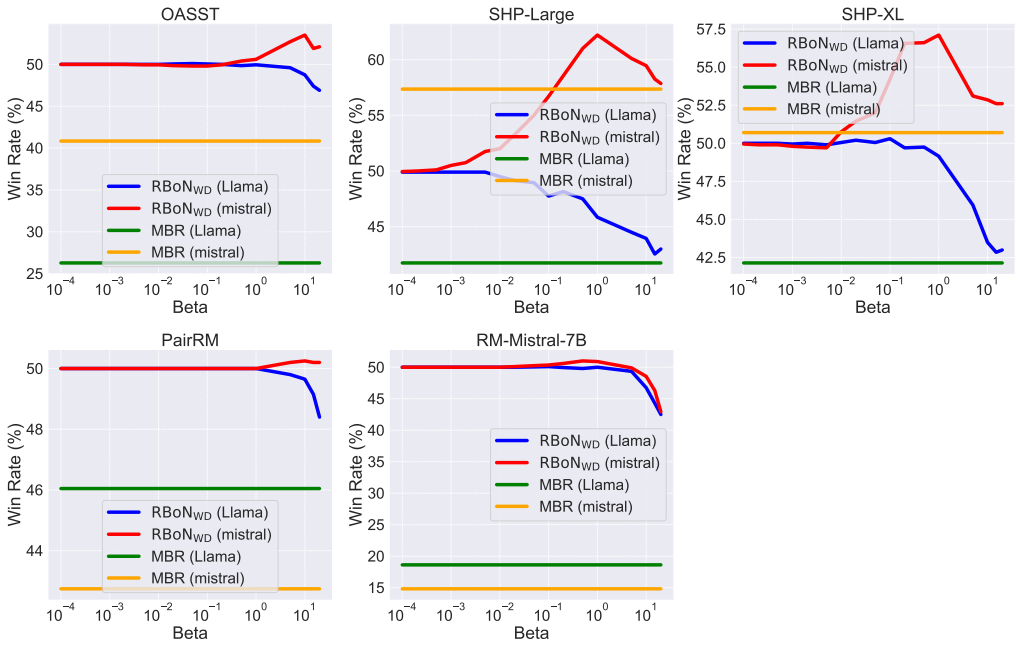

Figure 20: Evaluation of the RBoN method on the Harmlessness dataset with varying parameter β. We use proxy reward models, OASST, SHP-Large, SHP-XL, PairRM, and RM-Mistral-7B. As the gold reward model, we utilize Eurus-RM-7B, and Llama as the language model.

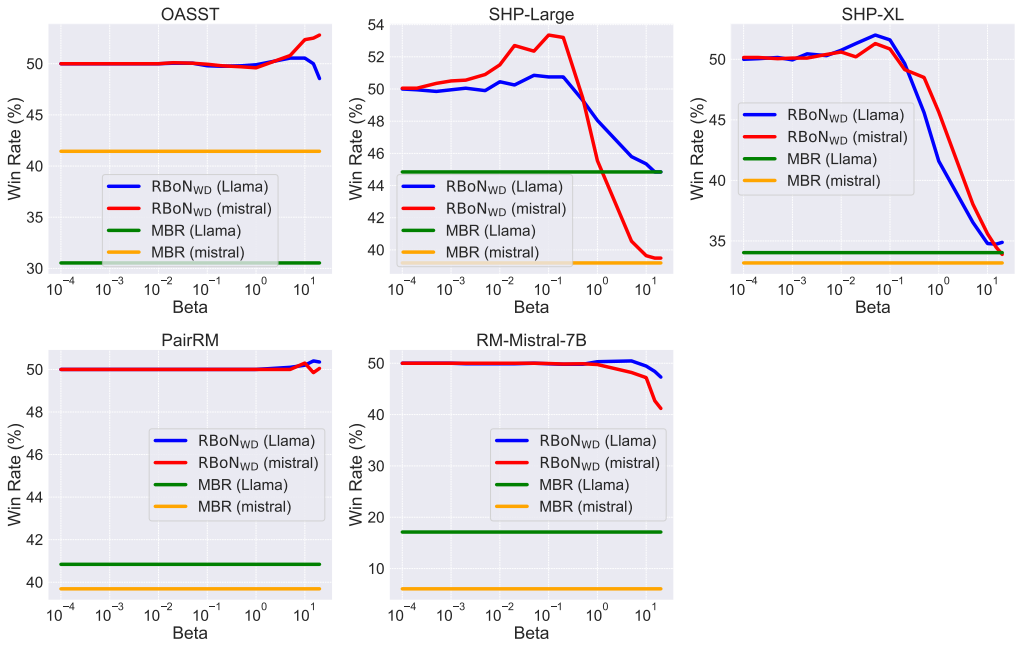

Figure 21: Evaluation of the RBoN method on the Helpfulness dataset with varying parameter β. We use proxy reward models, OASST, SHP-Large, SHP-XL, PairRM, and RM-Mistral-7B. As the gold reward model, we utilize Eurus-RM-7B, and Llama as the language model.

F Robustness of RBoN Under Suboptimal Reward Models

We evaluate the performance of suboptimal reward models, Beaver (beaver-7b-v1.0-reward) (Dai et al., 2024), Open Llama (hh-rlhf-rm-open-llama 3b) (Diao et al., 2024), and Tulu (tulu-v2.5-13b-uf-rm) (Ivison et al., 2024) selected from Lambert et al. (2024a), which underperforms compared to other reward models in some cases. We set these models as proxy models, set Eurus-RM-7B (Eurus) as the gold model, and also show the reward correlation of these models.

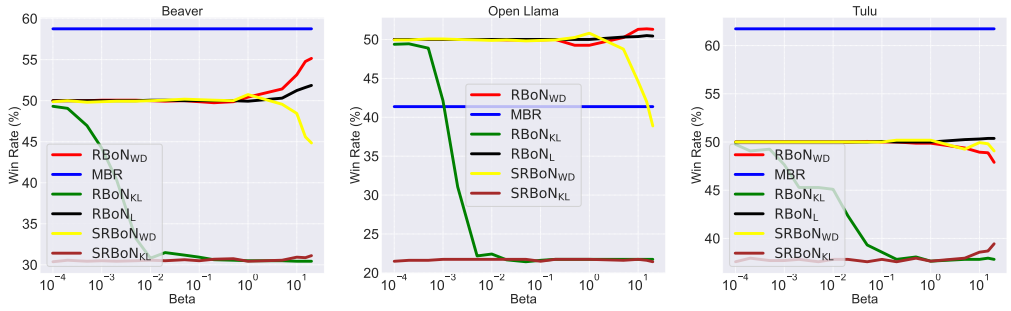

Figure 22: Evaluation of RBoN sensitiveness on the AlpacaFarm dataset with varying parameter β. We use proxy reward models, Beaver, Open Llama, and Tulu. As the gold reward model, we utilize Eurus.

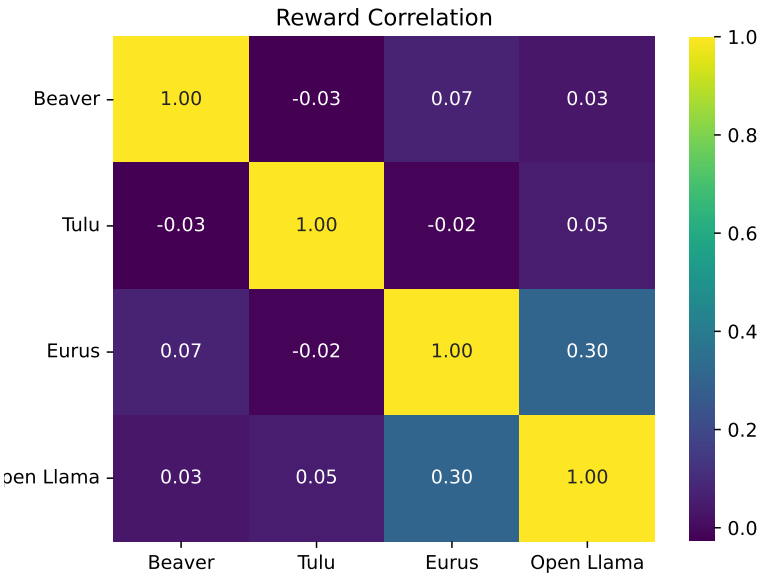

Figure 23: The average Spearman's rank correlation coefficient (ρ) between pairs of reward models in the AlpacaFarm dataset.

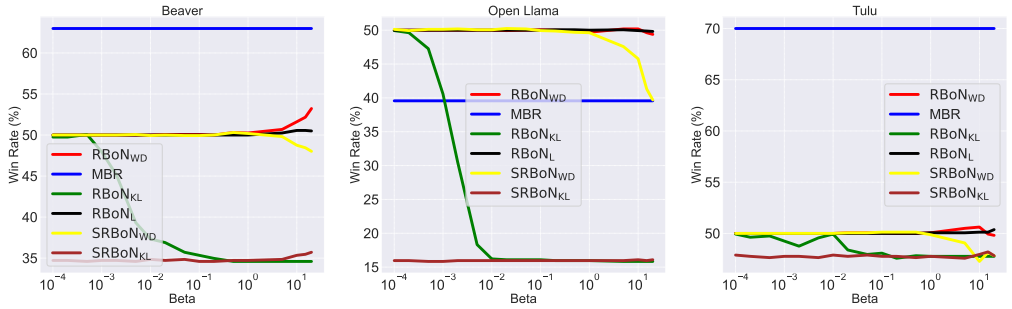

Figure 24: Evaluation of RBoN sensitiveness on the Helpfulness dataset with varying parameter β. We use proxy reward models, Beaver, Open Llama, and Tulu. As the gold reward model, we utilize Eurus.

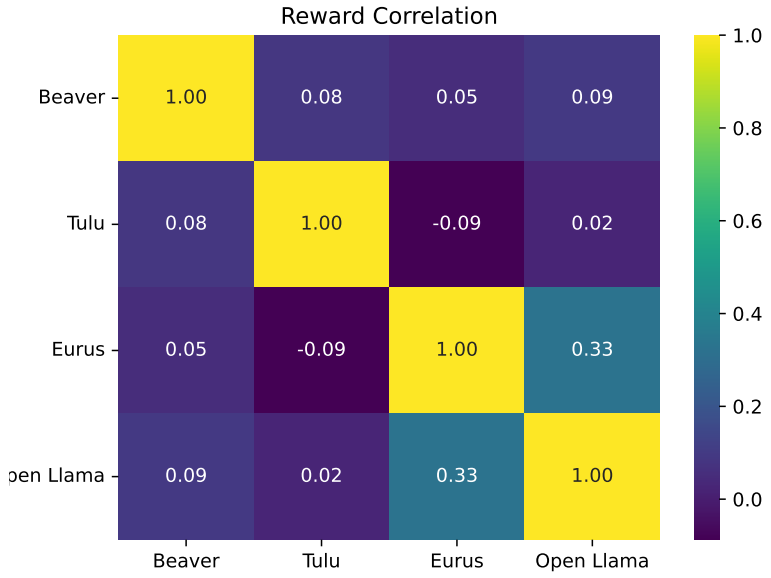

Figure 25: The average Spearman's rank correlation coefficient (ρ) between pairs of reward models in the Helpfulness dataset.

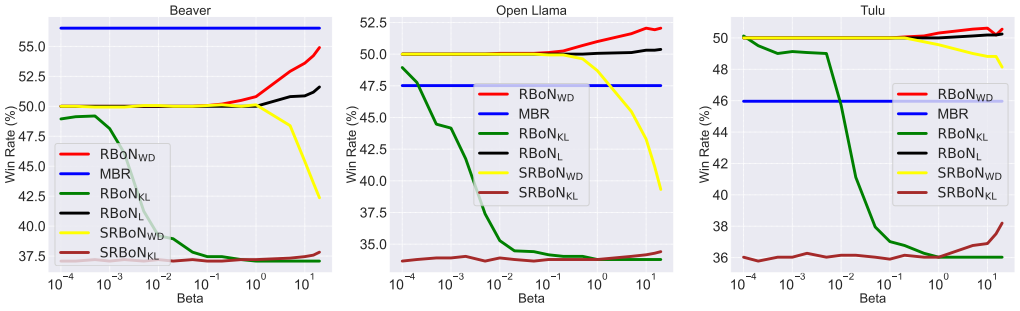

Figure 26: Evaluation of RBoN sensitiveness on the Harmlessness dataset with varying parameter β. We use proxy reward models, Beaver, Open Llama, and Tulu. As the gold reward model, we utilize Eurus.

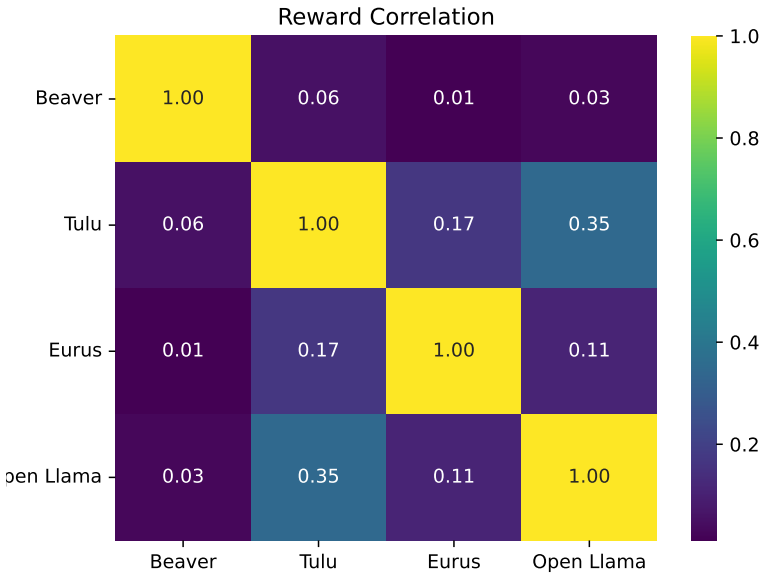

Figure 27: The average Spearman's rank correlation coefficient (ρ) between pairs of reward models in the Harmlessness dataset.

G Sentence Length Regularized BoN ($\mathrm{RBoN_L}$)

The objective function of $\mathrm{RBoN_L}$ (Sentence Length Regularized BoN) is given by:

$$y_{\mathbf{LBoN}}(x) = \arg\max_{y \in \mathcal{Y}_{\mathbf{ref}}} R(x, y) - \frac{\beta}{|y|}$$

where β is a regularization parameter, $|y|$ denotes the token length of the sentence y .

This approach aims to address the inherent bias toward shorter outputs often observed in a large language model we used in experiments. We now explain the rationale behind the specific form of the regularization term in $\mathrm{RBoN_L}$. Let μ represent a probability that is inversely proportional to the token length of the text y.

For example, we could define $\mu(y|x) = 1/|y|$ (e.g. $\mu(y'|x) = 1/|y'|, \mu(y''|x) = 1/|y''|...$), where $|y|$ represents the token length of output y.

Definition G.1. *We define a newly normalized distribution μ':*

$$\begin{aligned}\mu'(y \mid x) &= \frac{1/|y|}{\sum_{\mathcal{Y}_{\mathbf{ref}}} \mu(\cdot \mid x)} \\ &= \frac{1/|y|}{Z} \quad \left(\text{where } \sum_{\mathcal{Y}_{\mathbf{ref}}} \mu(\cdot \mid x) = Z\right)\end{aligned}$$

Proposition G.2. *The objective function of* $\mathrm{RBoN_L}$ *is derived by considering the TV distance between the output probability $\mathbb{1}_y(\cdot \mid x)$ and $\mu'(\cdot \mid x)$ as a regularization term.*

Proof. Let us examine how the objective function of $\mathrm{RBoN_L}$ is derived using definition G.1.

$$
\begin{aligned}
y_{\mathrm{LBoN}}(x) &= \underset{y \in \mathcal{Y}_{\mathbf{ref}}}{\arg\max}\ R(x,y) + \beta \mathbf{TV}\left[\mathbb{1}_y(\cdot \mid x) \| \mu'(\cdot \mid x)\right], \\
&= \underset{y \in \mathcal{Y}_{\mathbf{ref}}}{\arg\max}\ R(x,y) + \frac{\beta}{2} \sum_{y \in \mathcal{Y}_{\mathbf{ref}}} |\mathbb{1}_y(\cdot \mid x) - \mu'(\cdot \mid x)| \\
&= \underset{y \in \mathcal{Y}_{\mathbf{ref}}}{\arg\max}\ R(x,y) + \frac{\beta}{2} \left(\left| 1 - \frac{1}{Z|y|} \right| + \underbrace{\frac{1}{Z|y'|} + \frac{1}{Z|y''|} + \cdots + \frac{1}{Z|y'''|}}_{=1-\frac{1}{Z|y|}} \right) \\
&= \underset{y \in \mathcal{Y}_{\mathbf{ref}}}{\arg\max}\ R(x,y) + \beta \left(1 - \frac{1}{Z|y|} \right) \\
&= \underset{y \in \mathcal{Y}_{\mathbf{ref}}}{\arg\max}\ R(x,y) - \frac{\beta}{Z|y|} \\
&= \underset{y \in \mathcal{Y}_{\mathbf{ref}}}{\arg\max}\ R(x,y) - \frac{\beta'}{|y|} \quad \left(\beta' = \frac{\beta}{Z} \right)
\end{aligned}
$$

where β' is a regularization parameter and $\mathbf{TV}$ denotes TV distance. □

The purpose of this normalization is to counteract the effect of $\mathrm{SRBoN_{KL}}$, which tends to favor shorter outputs. This formulation provides a theoretical basis for understanding how $\mathrm{RBoN_L}$ achieves its length-aware behavior, and offers insight into its potential advantages over other decoding methods that may inadvertently bias toward shorter outputs.

Our methodological approach to assessing the divergence of output distributions from the length distribution μ' involves a comparative analysis of BoN sampling and $\mathrm{RBoN_L}$. For each output y selected, we construct the corresponding $\mathbb{1}_y(\cdot \mid x)$ distribution. We then measure the TV distance between $\mathbb{1}_y(\cdot \mid x)$ and $\mu'(\cdot \mid x)$.

The results of this comparative analysis are visualized in Figure 28, Figure 29, and Figure 30.

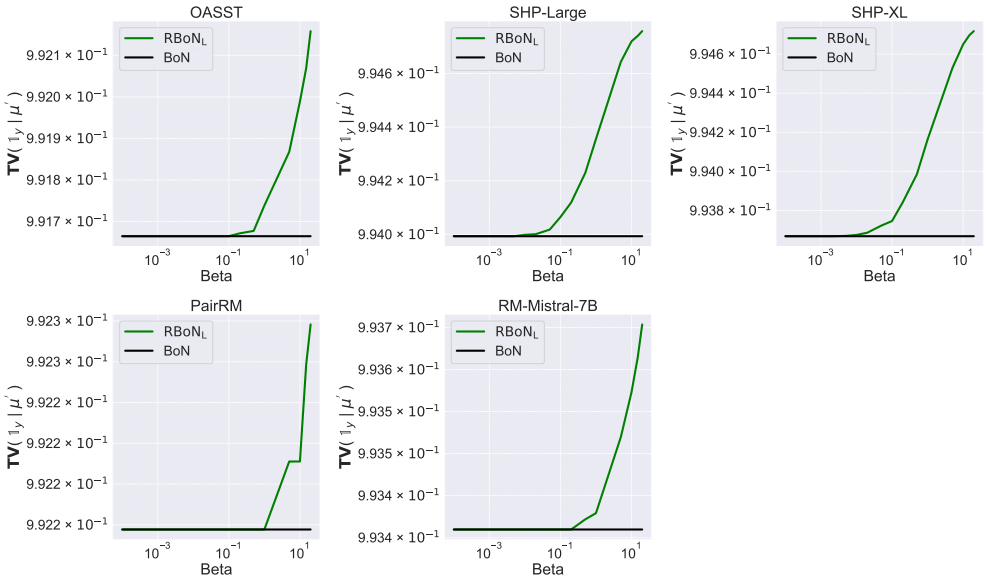

Figure 28: BoN sampling and RBoN$_L$ methods by measuring the TV distance between their output distributions and sentence length distribution μ' in AlpacaFarm. This allows us to evaluate how closely each method's outputs align with the desired distribution, with a smaller TV distance indicating a preference for shorter sentences.

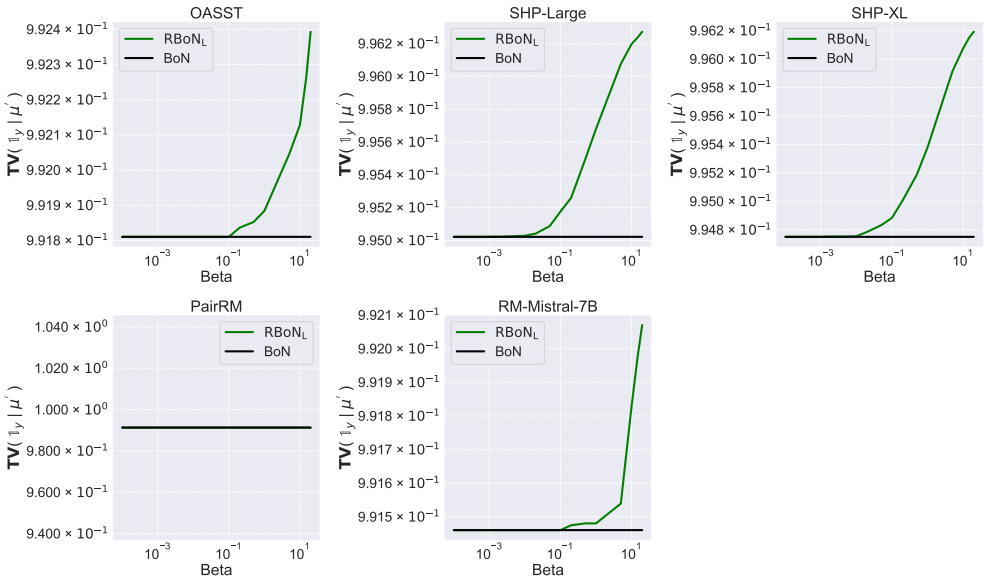

Figure 29: BoN sampling and RBoN$_L$ methods by measuring the TV distance between their output distributions and sentence length distribution μ' in Harmlessness. This allows us to evaluate how closely each method's outputs align with the desired distribution, with a smaller TV distance indicating a preference for shorter sentences.

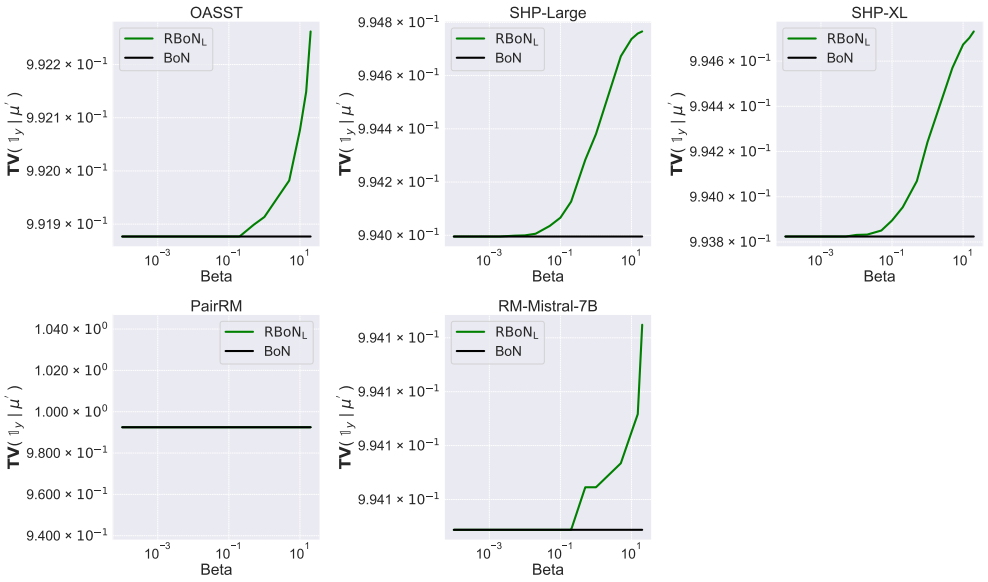

Figure 30: BoN sampling and RBoN$_{\text{L}}$ methods by measuring the TV distance between their output distributions and sentence length distribution μ' in Helpfulness. This allows us to evaluate how closely each method's outputs align with the desired distribution, with a smaller TV distance indicating a preference for shorter sentences.

Our analysis shows that the output probability of RBoN$_{\text{L}}$ deviates more from μ' than the output probability of BoN sampling. Table 7 illustrates the correlation between the length of the sequence and the values of gold reference reward (Eurus-RM-7B), focusing on subsets of sentences comprising the top 5, 10, 15 based on the proxy reward values. The strength of this correlation is an indication of the effectiveness of RBoN$_{\text{L}}$; a stronger correlation indicates greater effectiveness of the method.

In Table 7, we have highlighted in **bold** the instances of high correlation compared to all samples used correlation, which corresponds to superior performance as shown in Table 2. In contrast, areas with lower correlation tend to show lower performance. This pattern shows a consistent relationship between correlation strength and method effectiveness. We also explored an alternative view of PairRM that had a high correlation but did not produce correspondingly strong results in Table 2.

We hypothesized that this discrepancy might be due to the range of the regularization parameter β. To investigate this hypothesis and to demonstrate the potential of RBoN$_{\text{L}}$, we performed an extensive analysis by varying β over a wide range, from 10 to 5000 Figure 31.

H Experiment with Qwen2.5-7B-Instruct

As an ablation study, we evaluate the methods using the Qwen (Qwen2.5-7B-Instruct) as the language model. Overall, we observe the same results as with Mistral-7B-SFT, where RBoN$_{\text{WD}}$ outperforms the baseline algorithms (Figure 32).

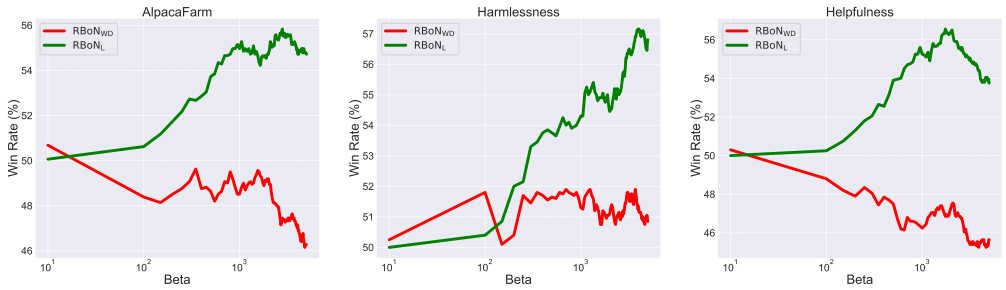

Figure 31: Performance analysis of RBoN$_{\mathrm{L}}$ with varying β (10 to 5000) across AlpacaFarm, Harmlessness, and Helpfulness datasets. PairRM and Eurus-RM-7B are used as proxy and gold reward models, respectively.

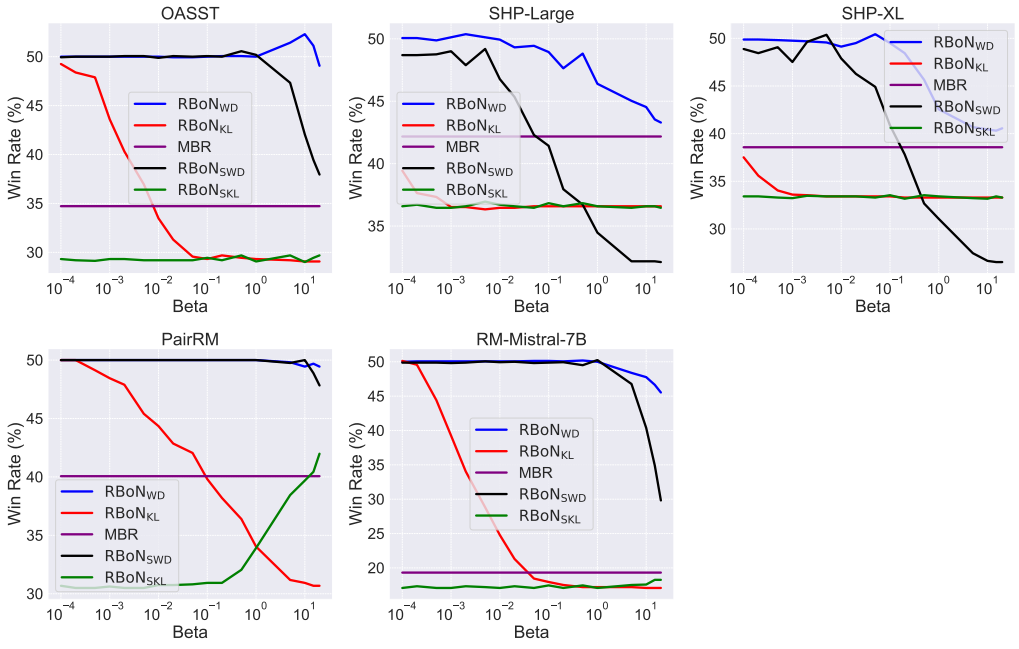

Figure 32: Evaluation of the RBoN method on the AlpacaFarm dataset with varying parameter β. We use proxy reward models, OASST, SHP-Large, SHP-XL, PairRM, and RM-Mistral-7B. As the gold reward model, we utilize Eurus-RM-7B, and Qwen as the language model.

Table 7: The correlation between sequence length and gold reference reward (Eurus-RM-7B) values, focusing on a subset of sentences that include the top 5, 10, 15 based on proxy reward values.

Top N	OASST	SHP-Large	SHP-XL	PairRM	RM-Mistral-7B
AlpacaFarm					
All	0.11 (0.33)	0.11 (0.33)	0.11 (0.33)	0.11 (0.33)	0.11 (0.33)
5	**0.27** (0.55)	-0.04 (0.56)	0.05 (0.55)	0.15 (0.59)	0.10 (0.56)
10	**0.24** (0.44)	-0.02 (0.44)	0.06 (0.41)	**0.17** (0.48)	0.09 (0.56)
20	**0.21** (0.39)	-0.02 (0.37)	0.06 (0.36)	**0.16** (0.41)	0.08 (0.44)
Harmlessness					
All	0.08 (0.45)	0.08 (0.45)	0.08 (0.45)	0.08 (0.45)	0.08 (0.45)
5	**0.24** (0.58)	0.10 (0.57)	0.13 (0.58)	**0.20** (0.62)	**0.37** (0.51)
10	**0.25** (0.50)	0.11 (0.46)	0.12 (0.47)	**0.19** (0.54)	**0.36** (0.41)
20	**0.22** (0.47)	0.11 (0.41)	0.11 (0.43)	**0.21** (0.49)	**0.34** (0.39)
Helpfulness					
All	0.07 (0.40)	0.07 (0.40)	0.07 (0.40)	0.07 (0.40)	0.07 (0.40)
5	**0.28** (0.56)	-0.04 (0.58)	0.11 (0.54)	**0.14** (0.62)	0.06 (0.54)
10	**0.27** (0.47)	-0.05 (0.45)	0.11 (0.42)	**0.15** (0.52)	0.06 (0.40)
20	**0.24** (0.43)	-0.06 (0.40)	0.10 (0.37)	**0.17** (0.46)	0.03 (0.36)

I Reproducibility Statement

All datasets and models used in the experiments are publicly available (Table 8). Our code will be available as open source upon acceptance.

Table 8: List of datasets and models used in the experiments.

Name	Reference
AlpacaFarm	Dubois et al. (2023) https://huggingface.co/datasets/tatsu-lab/alpaca_farm
Anthropic's hh-rlhf	Bai et al. (2022) https://huggingface.co/datasets/Anthropic/hh-rlhf
mistral-7b-sft-beta (Mistral)	Jiang et al. (2023a); Tunstall et al. (2023) https://huggingface.co/HuggingFaceH4/mistral-7b-sft-beta
Meta-Llama-3-8B-Instruct (Llama)	Dubey et al. (2024) https://huggingface.co/meta-llama/Meta-Llama-3-8B-Instruct
Qwen2.5-7B-Instruct (Qwen)	Yang et al. (2024); Team (2024) https://huggingface.co/Qwen/Qwen2.5-7B-Instruct
SHP-Large	Ethayarajh et al. (2022) https://huggingface.co/stanfordnlp/SteamSHP-flan-t5-large
SHP-XL	Ethayarajh et al. (2022) https://huggingface.co/stanfordnlp/SteamSHP-flan-t5-xl
OASST	Köpf et al. (2023) https://huggingface.co/OpenAssistant/reward-model-deberta-v3-large-v2
PairRM	Jiang et al. (2023b) https://huggingface.co/llm-blender/PairRM
RM-Mistral-7B	Dong et al. (2023) https://huggingface.co/weqweasdas/RM-Mistral-7B
Eurus-RM-7B	Yuan et al. (2024) https://huggingface.co/openbmb/Eurus-RM-7b
Beaver	Dai et al. (2024)https://huggingface.co/PKU-Alignment/beaver-7b-v1.0-reward
Tulu	Ivison et al. (2024) https://huggingface.co/allenai/tulu-v2.5-ppo-13b-uf-mean-70b-uf-rm
Open Llama	Diao et al. (2024) https://huggingface.co/weqweasdas/hh_rlhf_rm_open_llama_3b
MPNet	Song et al. (2020) https://huggingface.co/sentence-transformers/all-mpnet-base-v2