Token-level Accept or Reject: A Micro Alignment Approach for Large Language Models

Yang Zhang[1], **Yu Yu**[2], **Bo Tang**[3,2*], **Yu Zhu**[4], **Chuxiong Sun**[5], **Wenqiang Wei**[2], **Jie Hu**[5], **Zipeng Xie**[6], **Zhiyu Li**[2], **Feiyu Xiong**[2] and **Edward Chung**[1]

[1]Hong Kong Polytechnic University, Hong Kong SAR, China
[2]MemTensor (Shanghai) Technology Co., Ltd, Shanghai, China
[3]University of Science and Technology of China, Suzhou Institute for Advanced Research, Suzhou, China
[4]University of Science and Technology of China, Hefei, China
[5]China Telecom Corporation Limited Beijing Research Institute, Beijing, China
[6]Nanjing University of Information Science and Technology, Nanjing, China

Abstract

With the rapid development of Large Language Models (LLMs), aligning these models with human preferences and values is critical to ensuring ethical and safe applications. However, existing alignment techniques such as RLHF or DPO often require direct fine-tuning on LLMs with billions of parameters, resulting in substantial computational costs and inefficiencies. To address this, we propose **M**icro token-level **A**ccept-**R**eject **A**ligning (MARA) approach designed to operate independently of the language models. MARA simplifies the alignment process by decomposing sentence-level preference learning into token-level binary classification, where a compact three-layer fully-connected network determines whether candidate tokens are "Accepted" or "Rejected" as part of the response. Extensive experiments across seven different LLMs and three open-source datasets show that MARA achieves significant improvements in alignment performance while reducing computational costs. The source code and implementation details are publicly available at https://github.com/IAAR-Shanghai/MARA, and the trained models are released at https://huggingface.co/IAAR-Shanghai/MARA_AGENTS.

1 Introduction

The alignment of Large Language Models (LLMs) with human values and preferences has emerged as a crucial challenge in AI development [Wang *et al.*, 2023b]. The alignment is extremely important for LLMs that operate safely and ethically. Among various alignment approaches, Reinforcement Learning from Human Feedback (RLHF) [Ouyang *et al.*, 2022] and Direct Preference Optimization (DPO) [Rafailov *et al.*, 2024b] have emerged as two dominant paradigms. RLHF fine-tunes language models using a reward model trained on human preference datasets, while DPO directly optimizes the models through pairwise comparisons without relying on explicit reward modeling.

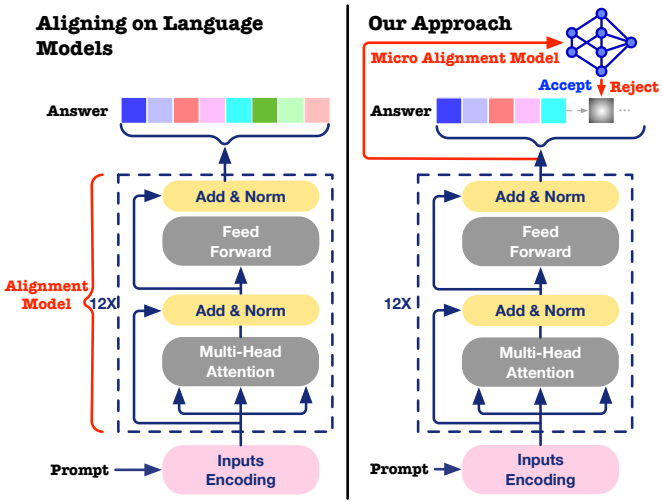


Figure 1: Unlike RLHF or DPO based alignment approach which fine-tunes the language models, the key insight of our approach is simplifying the alignment process into accepting or rejecting a token using a streamlined alignment model.

Although RLHF, DPO and their variants [Perez *et al.*, 2023; Bai *et al.*, 2022b; Lee *et al.*, 2023; Zeng *et al.*, 2024; Azar *et al.*, 2024; Wang *et al.*, 2023a] have demonstrated impressive capabilities in aligning performance, they face a critical challenge: excessive computational resource consumption. These methods require fine-tuning billion or even hundred-billion parameter of language models, typically demanding hundreds of GPU hours and substantial memory resources. Such computational intensity poses a significant barrier for real-world applications [Anwar *et al.*, 2024], particularly in scenarios requiring rapid alignment updates or resource-constrained environments.

To address this challenge, we pose a fundamental question: *Can we develop a micro alignment approach that operates independently of the language model while maintaining excellent alignment performance?*

*Corresponding Author <tangbo@mail.ustc.edu.cn>.

As shown in Figure 1, unlike aligning operated on the language model, our key insight is that the alignment process can be simplified into accepting or rejecting a token using a streamlined alignment model. Specifically, the alignment model is the first to be implemented as a micro fully-connected network. Given the prompt, the partially generated response, and a candidate set of tokens that are sorted by sampling probabilities from a supervised fine-tuned model, our alignment model sequentially determines whether to accept or reject each candidate token during the generation process. This approach transforms the alignment task into a simple binary classification problem, significantly reducing the computational overhead while ensuring effective alignment performance.

A concurrent work called *Aligner* [Ji *et al.*, 2024a] shares a similar motivation of decoupling alignment from the upstream language model. However, our approach differs significantly in both scale and methodology. While *Aligner* employs a seq2seq model with 2B-13B parameters, our method utilizes a micro fully-connected network with merely millions of parameters. Moreover, our execution process is fundamentally simpler, focusing on token-level accept/reject decisions rather than reformulating entire output sentences.

In summary, we propose **M**icro token-level **A**ccept-**R**eject **A**ligning (MARA) approach, which offers three significant advantages:

- Computation-friendly: **Our approach requires only a three-layer fully-connected network as the alignment model, reducing the computation overhead significantly.** Specifically, for aligning an 8B-parameter language model, MARA requires training only a 4M-parameter alignment model, whereas RLHF, DPO, and *Aligner* necessitate training language models with over 20M parameters, even when employing parameter-efficient Low-Rank Adaptation (LoRA).

- Effectiveness: **While operating at a micro level, our approach achieves superior alignment performance**. Specifically, with Llama 3.1-8B as the base model, MARA demonstrates substantial improvements across different benchmark datasets: +31.8% over RLHF, +18.8% over DPO, and +8.8% over *Aligner*.

- Compatibility: The decoupled architecture of MARA-trained alignment models enables seamless integration with various LLMs, i.e., **an alignment model trained on one LLM can be effectively transferred to other LLMs while maintaining strong performance**. For instance, an alignment model trained on Mistral-7B-v0.3 enhances the alignment performance of Llama 3-8B by 25.5% on average across three evaluation datasets.

We prioritize full reproducibility by releasing our complete implementation source code (available in the appendices). Our open-source commitment ensures that **all results presented in this paper can be independently verified and reproduced with minimal effort**, promoting transparency and facilitating future research in the field.

2 Preliminaries

This section introduces the standard RLHF framework, which consists of two primary phases: supervised fine-tuning (SFT) and reinforcement learning-based optimization. We first formalize these phases and then present a token-level decomposition of the alignment process.

2.1 RLHF Framework

SFT Phase: The pre-trained language model is initially fine-tuned on high-quality human demonstrations to generate appropriate responses. This process is optimized through the following objective:

$$J_{SFT} = -\mathbb{E}_{(x,y)\sim\mathcal{D}_{SFT}}\left[\log \pi_{\text{ref}}\left(y|x\right)\right] \tag{1}$$

where x and y denote the input prompt and model response, respectively, sampled from the supervised fine-tuning dataset $\mathcal{D}_{SFT}$. π_{ref} denotes the fine-tuned language model serving as the reference model for subsequent optimization.

RL-based Optimization Phase: Following SFT, the model undergoes reinforcement learning optimization to align with human preferences. The objective function for this phase is:

$$\begin{aligned} J_{RL} =& \mathbb{E}_{x\sim\mathcal{D}_{RL}, y\sim\pi_\theta(\cdot|x)}\left[r\left(x,y\right)\right. \\ & \left. -\lambda D_{KL}\left(\pi_\theta\left(\cdot|x\right) \parallel \pi_{\text{ref}}\left(\cdot|x\right)\right)\right] \end{aligned} \tag{2}$$

where π_θ denotes the model undergoing RL optimization, and $\mathcal{D}_{RL}$ represents the optimization dataset. The coefficient λ controls the KL divergence penalty between the reference model and the aligned model.

The reward model $r(x, y)$ evaluates the alignment between model outputs and human preferences. Given a preference dataset $\mathcal{D}$ containing triples (x, y_w, y_l), where x is the prompt, y_w is the preferred response, and y_l is the less preferred response, the reward model is trained to minimize:

$$\mathcal{L}\left(r_\phi\right) = -\mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}}\left[\log\left(\sigma\left(r_\phi\left(x,y_w\right) - r_\phi\left(x,y_l\right)\right)\right)\right] \tag{3}$$

where σ denotes the Sigmoid function.

2.2 Token-level Decomposition

At its core, the RLHF alignment process can be decomposed into a sequence of token-level decisions. The sentence generation process is formally expressed as:

$$\pi_\theta\left(y|x\right) = \pi_\theta\left(y_1, y_2, \ldots, y_H|x\right) = \prod_{i=1}^{H} \pi_\theta(y_i|x, y_{<i})_{y_i\sim\mathrm{T}_i} \tag{4}$$

where T_i denotes the vocabulary space of available tokens at the i-th position in the output sequence y, and H denotes the sequence length.

This token-level decomposition reveals the possibility of performing alignment at the granularity of individual tokens, enabling more precise control over the generation process. Furthermore, this granular perspective can be refined into explicit accept-or-reject decisions for each token, which motivates our proposed method detailed in the following sections.

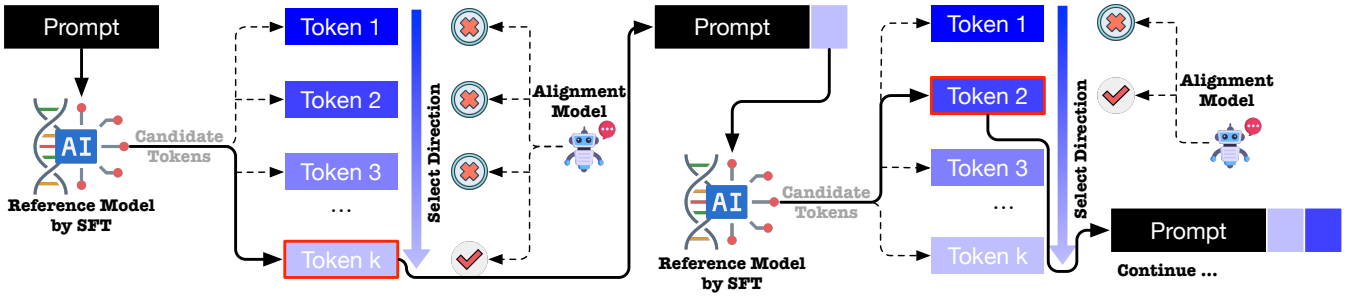


Figure 2: Architecture of MARA: The alignment model performs token selection through accept-reject decisions.

3 Our approach

In this section, we present MARA, a novel token-level accept-reject approach for LLM alignment. Building upon the foundation established by the SFT phase, MARA introduces a language model-independent alignment mechanism, replacing the original language model optimization in RLHF. Specifically, we first formalize the alignment task as a Markov Decision Process (MDP), then detail our accept-reject mechanism, and finally discuss implementation specifics.

3.1 MDP formulation of alignment

Given the reference model π_{ref} from the SFT phase, we formulate the alignment process as a MDP, characterized by the tuple $(\mathcal{S}, \mathcal{A}, \rho, \mathcal{P}, r)$, where:

- State space ($s \in \mathcal{S}$): For the i-th token generation at time step τ, the state $s_\tau = \{x, y_{<i}, t_i^k\}$ comprises the input prompt x, previously generated response $y_{<i}$, and the current candidate token $t_i^k \in \overline{\mathrm{T}}_i$, where $\overline{\mathrm{T}}_i$ denotes a truncated candidate token set (detailed below), and k indexes the candidate token in $\overline{\mathrm{T}}_i$.
- Action space ($a \in \{0, 1\}$): Binary action where $a_{t_i^k} = 1$ indicates accepting the candidate token t_i^k to concatenate with the generated response $y_{<i}$, while $a_{t_i^k} = 0$ indicates rejection. To facilitate understanding, we use $a_{t_i^k}$ instead of a_τ referring to the action on the candidate token.
- Initial state distribution (ρ): Defines the distribution over initial states.
- Reward function ($r : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$) will be detailed in Section 3.3.
- State transition ($\mathcal{P}$): The dynamics follow:

$$s_{\tau+1} = \begin{cases} \{x, y_{<i+1}, t_{i+1}^1\}, & \text{if } a_{t_i^k} = 1; \\ \{x, y_{<i}, t_i^{k+1}\}, & \text{else.} \end{cases} \tag{5}$$

To enhance computational efficiency, we employ a hybrid vocabulary truncation strategy combining two widely-used methods: top-k sampling, which retains only the k most probable tokens, and top-p (nucleus) sampling, which selects the smallest set of tokens whose cumulative probability exceeds p. This combination clips the vocabulary space T_i at each generation step to form a reduced candidate set $\overline{\mathrm{T}}_i$. This hybrid approach ensures both diversity and quality of the candidate tokens while maintaining computational tractability.

3.2 Token-level Accept-Reject Aligning Mechanism

Unlike traditional RLHF that fine-tunes the entire language model, our approach introduces a micro accept-reject model that operates at the token level. This model is implemented as a compact three-layer fully connected neural network. The alignment process follows four key steps:

Step 1: Candidate Generation. Given the prompt x and the previously generated tokens $y_{<i}$, we use the reference model π_{ref} to generate a candidate token set $\overline{\mathrm{T}}_i = \{t_i^1, t_i^2, \ldots, t_i^{|\overline{\mathrm{T}}_i|}\}$ through hybrid sampling that combines top-p and top-k strategies.

Step 2: Probability-based Sorting. Sort the candidate token set $\overline{\mathrm{T}}_i$ in descending order according to their conditional probabilities $\pi_{\text{ref}}(t_i^k | x, y_{<i})$, resulting in an ordered set $\widetilde{\mathrm{T}}_i$.

Step 3: Token Evaluation. For the first token in the $\widetilde{\mathrm{T}}_i$, apply the accept-reject model to evaluate whether to accept or reject the token. The input of the accept-reject model consists of the last hidden state of the token sequence: prompt x, previously generated tokens $y_{<i}$, and the first candidate token in $\widetilde{T}_i$. The output is a binary decision: accept or reject. If accepting the token, add it to the output sentence y, and return to Step 1 until the **EOS** token is selected or the output length reaches its predefined limit. Otherwise, proceed to Step 4.

Step 4: Candidate Update. Upon rejection, remove the token from the candidate token set $\widetilde{\mathrm{T}}_i$, and return to Step 3.

To maintain generation quality, we order candidate tokens by their sampling probabilities in descending order, as higher probability tokens typically better align with the reference model's learned patterns, thus reducing the risk of generating grammatically incorrect or contextually inconsistent content.

For cases when all tokens in the candidate set are rejected by the accept-reject model, we employ a fallback mechanism to ensure generation continuity. Specifically, we enforce the acceptance of the final candidate token, formally expressed as $a_{t_i^k} \in \{1\}$ when $k = \left|\widetilde{\mathrm{T}}_i\right|$. This strategy ensures generation proceeds by selecting lower-probability tokens when necessary, thereby balancing quality and diversity.

According to the above token-level aligning approach, the sampling probability for any token t_i^k, representing the k-th

candidate token in the candidate set $\widetilde{\mathrm{T}}_i$, is formalized as:

$$\begin{aligned}\pi_\theta\left(y_i = t_i^k\right) =& \pi_\theta\left(a_{t_i^k} = 1 | x, y_{<i}\right) \\ & \prod_{k'=1}^{k-1} \pi_\theta\left(a_{t_i^{k'}} = 0 | x, y_{<i}\right)\Big|_{t_i^k \in \widetilde{\mathrm{T}}_i}.\end{aligned} \tag{6}$$

Note that we still use π_θ to denote the accept-reject model parameterized by θ.

From above, the complete aligned response generation process can thus be expressed as:

$$\pi_\theta\left(y|x\right) = \prod_{i=1}^{H} \pi_\theta(y_i = t_i^k | x, y_{<i})\big|_{t \sim \widetilde{\mathrm{T}}_i}. \tag{7}$$

3.3 Implementation

Reward Design

A key challenge in MARA is bridging the gap between sentence-level rewards in standard RLHF and token-level decisions. Our reward function is designed with two objectives: (1) maximizing the reward model score $r(x, y)$ to maintain alignment with human preferences as in RLHF, and (2) preventing reward hacking by constraining the token distribution divergence between the aligned model and the reference model, thereby avoiding unnatural token selections [Eisenstein *et al.*, 2023].

Specifically, for any action $a_{t_i^k}$ at state s_τ, we define the token-level reward as:

$$r_\tau = \begin{cases} -\lambda D_{KL}(\pi_\theta(t_i^k | x, y_{<i}) \parallel \pi_{\text{ref}}(t_i^k | x, y_{<i})), \\ \quad \text{if } s_\tau \text{ is not terminal;} \\ r(x, y) - \lambda D_{KL}(\pi_\theta(t_i^k | x, y_{<i}) \parallel \pi_{\text{ref}}(t_i^k | x, y_{<i})), \\ \quad \text{else.} \end{cases} \tag{8}$$

where $r(x, y)$ denotes the evaluation score by reward model trained by Eq. 3.

This formulation represents a token-level decomposition of the RLHF objective in Eq. 2. We adopt the distributed Soft Actor-Critic (SAC) algorithm [Haarnoja *et al.*, 2018] instead of the conventional Proximal Policy Optimization (PPO) algorithm used in RLHF due to SAC's higher computation efficiency, especially in distributed training environment. PPO requires synchronization between distributed nodes for sample collection and model updates, creating waiting periods. In contrast, SAC enables continuous sample collection across all nodes with real-time parameter updates via a global replay buffer, eliminating waiting periods.

Training Objectives

The alignment model serves as the actor in our framework, with its loss function defined as:

$$\begin{aligned}\mathcal{L}\left(\pi_\theta\right) = -\mathbb{E}_{x \sim \mathcal{D}}[&\alpha_h \mathcal{H}\left(\pi_\theta\left(s\right)\right) \\ &+ \mathbb{E}_{t_k^i \sim \pi_\theta}\left[\min\left(V_1\left(s\right), V_2\left(s\right)\right)\right]]\end{aligned} \tag{9}$$

where $\mathcal{H}$ represents the entropy term of the alignment model that promotes diverse token selection, while $V_1(s)$ and $V_2(s)$ are two critic heads for robust value estimation and overfitting prevention.

The critic network, parameterized by φ, is trained using the following loss:

$$\begin{aligned}\mathcal{L}(\pi_\varphi) = \mathbb{E}_{x \sim \mathcal{D}}\Bigg[\Bigg(&\frac{V_1(s) + V_2(s)}{2} - r(x, \pi_\theta(x)) \\ &- \alpha_h \mathcal{H}(\pi_\theta(s')) \\ &- \gamma \mathbb{E}_{t_k^{i'} \sim \pi_\theta}\left[\min(V_1'(s'), V_2'(s'))\right]\Bigg)^2\Bigg]\end{aligned} \tag{10}$$

where π_φ denotes the critic network with parameters φ. $V_1'(s')$ and $V_2'(s')$ denote the two heads of target critic network.

The entropy coefficient α_h is automatically adjusted according to:

$$\mathcal{L}(\alpha_h) = \mathbb{E}_{a_h \sim \pi_\theta}\left[-\alpha_h \pi_\theta\left(a_\tau | s_\tau\right) - \alpha_\tau \overline{\mathcal{H}}\right] \tag{11}$$

where $\overline{\mathcal{H}} = 2$ is the minimum entropy threshold based on the action space dimension.

4 Experiments

We conduct comprehensive experiments to evaluate MARA's effectiveness across diverse datasets, reward models, and base LLMs.

4.1 Experimental Setup

Datasets. Our experiments leverage three comprehensive evaluation benchmarks for utility, safety and ethical assessment: PKU-SafeRLHF (SafeRLHF) [Ji *et al.*, 2024b], which serves as our primary training dataset and contains 83.4K preference pairs with safety meta-labels and human preferences for helpfulness and harmlessness; BeaverTails [Ji *et al.*, 2024c], covering 14 safety categories including sensitive topics like abuse and political discussions; and HarmfulQA [Bhardwaj and Poria, 2023], comprising 10 themes of ChatGPT-generated conversations for evaluating responses in potentially harmful scenarios.

Upstream LLMs. We evaluate our alignment model on two widely-adopted open-source LLM families with various model scales and versions. Specifically, the Llama family includes Llama-3-8B, Llama-3.1-8B, Llama-3.2-3B, and Llama-3.2-1B [AI@Meta, 2024], while the Mistral family comprises Mistral-7B-v0.1, Mistral-7B-v0.2, and Mistral-7B-v0.3 [Jiang *et al.*, 2023]. Due to computational constraints, we do not include larger models such as Llama-3-70B in our experiments. For brevity, we omit the suffix 'Instruct' from all model names.

Evaluation Metrics. Following the methodology of *aligner* [Ji *et al.*, 2024a], we adopt the preference rate metric to evaluate model performance across two critical dimensions: helpfulness and harmlessness, which respectively assess the utility and safety aspects of generated responses. The preference rate is defined as:

$$w = \frac{N_w - N_l}{N_w + N_e + N_l} \times 100\% \tag{12}$$

where N_w, N_e, and N_l denote the numbers of wins, ties, and loses respectively in pairwise comparisons between different

Table 1: Performance improvements of MARA across PKU-SafeRLHF, BeaverTails, and HarmfulQA datasets. Each entry shows the percentage improvement in preference rate achieved by applying MARA compared to using the original LLM alone.

	Upstream LLM + MARA *vs.* Upstream LLM		
Upstream LLM	SafeRLHF	BeaverTails	HarmfulQA
Llama 3-8B	+33.67%	+36.86%	+52.05%
Llama 3.1-8B	+39.70%	+35.43%	+62.09%
Llama 3.2-1B	+45.23%	+31.71%	+59.43%
Llama 3.2-3B	+32.66%	+29.43%	+35.45%
Mistral-7B-v0.1	+11.06%	+17.43%	+10.45%
Mistral-7B-v0.2	+4.52%	+14.29%	+4.92%
Mistral-7B-v0.3	+17.09%	+16.00%	+11.07%
Average	+26.28%	+25.88%	+33.64%

alignment approaches. A comprehensive description of the evaluation metrics is provided in AppendixA.2.

Reward Models. Our alignment training utilizes a combination of two specialized models: beaver-7b-v1.0-reward for utility evaluation and beaver-7b-v1.0-cost [Dai *et al.*, 2024] for safety evaluation (where a negative cost score indicates a safe response). The final reward $r(x, y)$ is computed as:

$$r(x, y) = \alpha_r R(x, y) - \alpha_c C(x, y) \tag{13}$$

where $R(x, y)$ and $C(x, y)$ denote the evaluation scores from the reward and cost models respectively, and α_r and α_c are their corresponding weights.

For the Llama family of models, we employ a balanced weighting scheme with $\alpha_r = \alpha_c = 1$. For the Mistral family, we adopt an asymmetric weighting ($\alpha_r = 2$, $\alpha_c = 1$) to counteract their inherent bias towards safety over utility, thereby achieving a better utility-safety trade-off.

Computing Resources. Our experiments are performed on one Nvidia H800-80GB GPU. The machine is equipped with 192 Intel(R) Xeon(R) Platinum 8468v processors and has a CPU memory of 1584 GB.

4.2 Experiment Results

Performance on Different Evaluation Datasets.

Table 1 shows that MARA significantly improves the alignment performance of various upstream LLMs with different scales and versions in all three challenging datasets. For the Llama family, MARA achieves remarkable improvements, with gains of up to +45.23% on SafeRLHF, +36.86% on BeaverTails, and +62.09% on HarmfulQA. The improvements are particularly pronounced for Llama 3.1-8B, which shows the strongest overall performance gains. For the Mistral family, while the improvements are more modest, MARA still demonstrates consistent positive impacts, with average gains of +10.89%, +15.91%, and +8.81% across the three datasets respectively. On average, MARA yields substantial improvements of +26.28%, +25.88%, and +33.64% across SafeRLHF, BeaverTails, and HarmfulQA respectively, indicating its robust generalization capability across different evaluation scenarios. More detailed experimental results, including the win/tie/lose statistics across different models and datasets, are presented in Appendix B.1.

Comparison with Different Baselines.

Table 2 presents a comprehensive comparison between MARA and three representative baselines: RLHF, DPO, and *Aligner*. For the implementation of RLHF and DPO, we utilize the source code from [Zheng *et al.*, 2024]; For the implementation of *Aligner*, we utilize the aligner-7b-v1.0 alignment model from [Aligners, 2024]. Note that we don't include recent variants of RLHF and DPO (e.g., RLAIF, RLHAIF, TDPO, IPO) as baselines since they mainly improve specific aspects (e.g., feedback sources, overfitting) rather than proposing fundamentally different alignment mechanisms. The experimental results demonstrate that MARA consistently outperforms RLHF and DPO across all three datasets, achieving average improvements of +18.65% and +12.59% respectively.

Compared to *Aligner*, MARA shows marginally lower performance on certain Llama-based models, likely because *Aligner* was trained using Llama models as base models. Nevertheless, MARA demonstrates competitive advantages on SafeRLHF (+5.60%) and BeaverTails (+4.37%) evaluation datasets, while showing stronger ablation effects on HH-RLHF and Ultra-Feedback datasets (Table 4). More importantly, MARA shows significant computational advantages: it requires only a micro three-layer alignment model (4M parameters) compared to *Aligner*'s 7B parameters, achieving higher inference speed (31.41 *vs.* 20.63 tokens/s).

Compatibility Analysis.

To verify the compatibility of our approach, we conduct comprehensive experiments examining how alignment models trained on one LLM generalize to other inference LLMs. Table 3 presents the results for two alignment models, trained on Llama-3.1-8B and Mistral-7B-v0.3 respectively, when applied to various inference LLMs. The results show that the alignment model trained on Llama-3.1-8B demonstrates strong generalization capability, achieving substantial improvements across all evaluation datasets. Even when applied to different model families, it maintains robust performance, with improvements ranging from +4.02% to +28.14% across different Mistral versions. Similarly, the Mistral-7B-v0.3-trained alignment model shows consistent improvements across both model families, with particularly strong performance when applied to Llama 3-8B (SafeRLHF: +18.09%, BeaverTails: +20.71%, HarmfulQA: +37.70%). Overall, our approach demonstrates strong cross-model compatibility with average improvements of +13.82%, +13.91%, and +17.87% across SafeRLHF, BeaverTails, and HarmfulQA respectively.

4.3 Ablation Studies

Ablation on Training Datasets

The experiments introduced before utilized PKU-SafeRLHF datasets for training the alignment model. To verify MARA's effectiveness across different training datasets, we evaluate its performance against SFT, RLHF, DPO and *Aligner* on two additional datasets: HH-RLHF [Bai *et al.*, 2022a] and Ultra-Feedback [Cui *et al.*, 2023]. Table 4 presents the results using Llama 3.1-8B as the upstream LLM. The experimental results demonstrate MARA's consistent superiority, with substantial average improvements over all baselines: +23.25% over SFT, +31.75% over RLHF,

Table 2: Performance comparison of MARA against RLHF, DPO, and *Aligner* measured by percentage improvements of preference rate.

	MARA *vs.* RLHF			MARA *vs.* DPO			MARA *vs. Aligner*		
Upstream LLM	SafeRLHF	BeaverTails	HarmfulQA	SafeRLHF	BeaverTails	HarmfulQA	SafeRLHF	BeaverTails	HarmfulQA
Llama 3-8B	+29.65%	+22.14%	+45.08%	+15.08%	+14.86%	+28.28%	+5.53%	+2.71%	+5.74%
Llama 3.1-8B	+21.61%	+28.00%	+5.74%	+0.00%	+14.57%	+1.64%	+0.50%	-3.14%	-11.89%
Llama 3.2-1B	+39.20%	+22.29%	+38.93%	+0.50%	+0.14%	+14.96%	-6.03%	-12.0%	-9.63%
Llama 3.2-3B	+8.54%	+7.43%	-1.02%	+1.01%	+2.71%	-4.92%	-2.01%	-6.71%	-12.91%
Mistral-7B-v0.1	+8.04%	-0.14%	+0.00%	+9.55%	+17.29%	+6.15%	+10.55%	+17.0%	+0.61%
Mistral-7B-v0.2	+24.62%	+16.86%	+17.01%	+22.61%	+27.0%	+21.93%	+14.57%	+13.43%	+4.30%
Mistral-7B-v0.3	+14.07%	+16.86%	+26.64%	+17.09%	+29.86%	+23.98%	+16.08%	+19.29%	+11.48%
Average	+20.82%	+16.21%	+18.91%	+9.41%	+15.20%	+13.15%	+5.60%	+4.37%	-1.76%

Table 3: Compatibility analysis for our approach, that an alignment model trained with a LLM to be aggregate with other inference LLM. The value of each cell represents the percentage improvement in preference rate of our algorithm over the upstream model, *i.e.,* inference model.

		MARA *vs.* Inference LLM		
Training LLM	Inference LLM	SafeRLHF	BeaverTails	HarmfulQA
Llama-3.1-8B	Llama 3.1-8B	+39.70%	+35.43%	+62.09%
	Llama 3-8B	+27.14%	+28.14%	+45.29%
	Mistral-7B-v0.3	+14.07%	+5.00%	+6.15%
	Mistral-7B-v0.2	+4.02%	+7.00%	+7.17%
	Mistral-7B-v0.1	+17.09%	+12.57%	+1.64%
Mistral-7B-v0.3	Mistral-7B-v0.3	+9.05%	+9.14%	+6.97%
	Mistral-7B-v0.2	+3.02%	+1.71%	+1.43%
	Mistral-7B-v0.1	+4.52%	+6.86%	+4.30%
	Llama 3.1-8B	+1.51%	+12.57%	+5.94%
	Llama 3-8B	+18.09%	+20.71%	+37.70%
Average		+13.82%	+13.91%	+17.87%

Table 4: Performance comparison of MARA against baseline approaches (SFT, RLHF, DPO, and *Aligner*) on various datasets, reported as percentage improvements in preference rate. All experiments use Llama 3.1-8B as the upstream model.

Dataset	*vs.* SFT	*vs.* RLHF	*vs.* DPO	*vs. Aligner*
HH-RLHF	+24.00%	+43.00%	+28.50%	+8.50%
Ultra-Feedback	+22.50%	+20.50%	+9.00%	+9.00%
Average	+23.25%	+31.75%	+18.75%	+8.75%

+18.75% over DPO, and +8.75% over *Aligner*. Notably, MARA achieves particularly strong performance on HH-RLHF, with improvements of up to +43.00% over RLHF, while maintaining robust gains on Ultra-Feedback across all baseline comparisons.

Ablation on the Reward Signal Distribution

To analyze the impact of different reward signal distributions, we conduct experiments with various ratios between reward and cost model signals (α_r:α_c). As shown in Table 5, we observe distinct trade-offs between helpfulness and harmlessness across different configurations. When solely using the reward model (α_r:α_c=1:0), the model shows improved helpfulness (up to +37.57%) but decreased harmlessness. Conversely, using only the cost model (α_r:α_c=0:1) leads to significant gains in harmlessness (up to +65.83%) but often at the expense of helpfulness. A balanced ratio of α_r:α_c=2:1 achieves the best overall performance across datasets, with preference rate improvements reaching +17.09% on SafeRLHF, +16.00% on BeaverTails, and +11.07% on HarmfulQA using Mistral-7B-v0.3. This suggests that while both reward signals are important, slightly emphasizing the reward model over the cost model leads to optimal balance between helpfulness and safety. Detailed ablation studies on the impact of candidate token set size and the sensitivity to KL divergence coefficients are presented in Appendix B.2 and Appendix B.3, respectively.

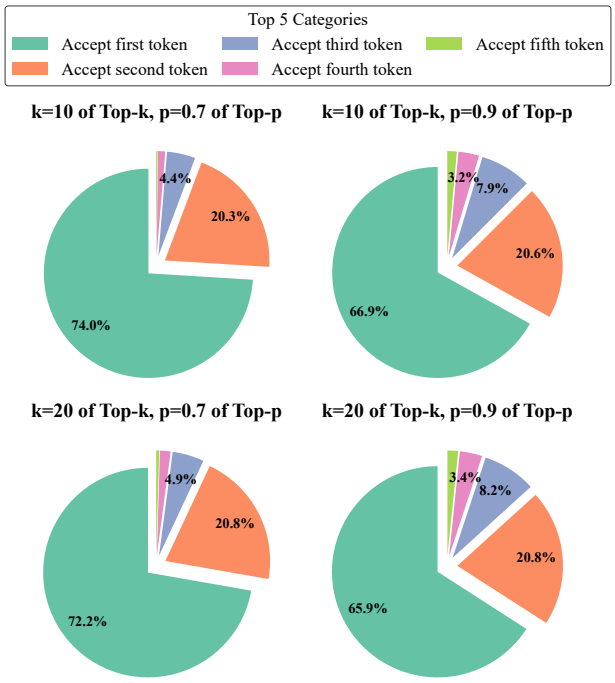


Figure 3: Token acceptance distribution under various Top-k and Top-p sampling configurations using Llama 3.2-1B on the PKU-SafeRLHF dataset. The plots show the first five most frequently accepted token positions, with acceptance rates below 2% omitted.

4.4 Visualisation of Accepted Tokens Distribution

To visualize the accept or rejection process for candidate tokens under our approach, and evaluate the impact of sampling parameters on token selection, we present an instance of the experiment performed on PKU-SafeRLHF dataset with Llama-3.2-1B as the upstream LLM. More experiments are provided in the Appendix B.4.

Figure 3 shows the distribution of accepted tokens across the candidate token set under different sampling parameter

Upstream LLM	Reward distribution	SafeRLHF			BeaverTails			HarmfulQA		
		Helpful(↑)	Harmless(↑)	Perference(↑)	Helpful(↑)	Harmless(↑)	Perference(↑)	Helpful(↑)	Harmless(↑)	Perference(↑)
Mistral-7B-v0.1	α_r:α_c=1:0	**+29.65%**	-26.13%	+2.01%	**+37.29%**	-22.57%	+7.29%	**+21.93%**	-18.44%	+1.43%
	α_r:α_c=0:1	-55.78%	**+63.82%**	+4.02%	-34.29%	**+51.57%**	+8.57%	-49.39%	**+53.89%**	+2.46%
	α_r:α_c=1:1	-37.49%	+49.25%	+6.03%	-19.71%	+48.71%	+14.43%	-40.78%	+51.84%	+5.53%
	α_r:α_c=2:1	-16.58%	+28.69%	**+11.06%**	-2.71%	+37.21%	**+17.43%**	-22.75%	+43.44%	**+10.45%**
Mistral-7B-v0.2	α_r:α_c=1:0	**+24.62%**	-20.60%	+2.01%	**+31.71%**	-21.14%	+5.29%	**+27.05%**	-22.95%	+2.05%
	α_r:α_c=0:1	-91.96%	**+40.70%**	-1.01%	-30.00%	+33.71%	+1.86%	-52.87%	**+52.46%**	-0.20%
	α_r:α_c=1:1	-22.61%	+35.68%	**+6.53%**	-15.43%	**+35.14%**	+9.71%	-41.19%	+50.41%	+4.51%
	α_r:α_c=2:1	-20.6%	+30.15%	+4.52%	+9.43%	+19.29%	**+14.29%**	-31.56%	+41.6%	**+4.92%**
Mistral-7B-v0.3	α_r:α_c=1:0	**+29.65%**	-9.05%	+10.55%	**+37.57%**	-11.29%	+13.00%	**+11.07%**	-9.84%	+0.61%
	α_r:α_c=0:1	-39.70%	**+65.83%**	+13.07%	-37.71%	**+45.71%**	+4.00%	-56.97%	+60.25%	+1.64%
	α_r:α_c=1:1	-31.66%	+54.27%	+11.56%	-17.14%	+43.14%	+13.00%	-50.41%	**+69.06%**	+9.22%
	α_r:α_c=2:1	-17.39%	+50.75%	**+17.09%**	-2.29%	+34.71%	**+16.00%**	-42.62%	+64.75%	**+11.07%**

Table 5: Impact of reward signal distribution on model performance in terms of helpful, harmless, and preference rate. α_r:α_c represents the ratio between reward model (beaver-7b-v1.0-reward) and cost model (beaver-7b-v1.0-cost) signals in the reward of $r(x, y)$. **Bold** numbers indicate the best performance under each metric (helpful/harmless/preference) for each model version.

configurations. The analysis reveals three key findings:

1) Dominance of first token: Across all parameter settings, the first candidate token demonstrates strong dominance, accounting for 65.9-74.0% of all acceptances. This suggests the model maintains high confidence in its primary predictions with the greatest sampling probability.

2) Parameter sensitivity: Increasing the Top-p value and Top-k value results in a notable decrease in first token acceptance, while simultaneously increasing the acceptance rates of second and third tokens. This indicates that higher sampling thresholds promote more diverse token selection.

3) Optimal decision window: The acceptance distribution demonstrates MARA has learned to concentrate its safety-alignment decisions within a compact token space, where over 95% of all acceptances occur within the top three positions. Such a focused decision mechanism helps maintain generation efficiency while ensuring safety controls.

5 Related Works

Prior approaches to language model alignment can be categorized along two primary dimensions: the utilization of RL and the granularity of alignment.

5.1 Alignment Methodology

RL-based Alignment: RL has demonstrated remarkable effectiveness across various language tasks, including question-answering, machine translation, and text summarization [Kang *et al.*, 2020; Ziegler *et al.*, 2019; Stiennon *et al.*, 2020; Nakano *et al.*, 2021]. Recent years have witnessed its successful application in aligning language models with human values and preferences. State-of-the-art language models, notably InstructGPT [Ouyang *et al.*, 2022] and Llama2 [Touvron *et al.*, 2023], leverage RL-based fine-tuning. A prominent paradigm in this domain is RLHF, which constructs reward models from human preference data to guide model fine-tuning. Building upon RLHF, RLxF extends this framework to incorporate diverse feedback sources beyond human responses. The variable x in RLxF encompasses AI feedback (RLAIF) [Bai *et al.*, 2022b; Lee *et al.*, 2023] and hybrid human-AI feedback (RLHAIF) [Wu *et al.*, 2021; Saunders *et al.*, 2022; Perez *et al.*, 2023].

Non-RL Alignment: Given the implementation complexity and computational demands of RL-based approaches, alternative methodologies have emerged. RAFT [Dong *et al.*, 2023] and RRHF [Yuan *et al.*, 2023] employ selective fine-tuning on high-quality samples. Rain [Li *et al.*, 2024] and *Aligner* [Ji *et al.*, 2024a] adopt output rectification strategies, with Rain utilizing an evaluation-rewind mechanism and *Aligner* implementing a supervised, decoupled alignment model. DPO [Rafailov *et al.*, 2024b] establishes a theoretical mapping between optimal policies and reward functions, eliminating explicit reward modeling. Recent extensions—IPO [Azar *et al.*, 2024], Token-DPO [Zeng *et al.*, 2024], and DPO-f [Wang *et al.*, 2023a]—address challenges in overfitting, fine-grained alignment, and reverse KL regularization constraints, respectively. Our approach maintains alignment quality comparable to RL-based methods while achieving computational efficiency superior to existing non-RL techniques.

5.2 Alignment Granularity

Sentence-Level Alignment: In most alignment research, the alignment environment is modeled as a bandit environment, treating complete sentences as atomic actions. This paradigm encompasses RLHF [Ouyang *et al.*, 2022] and its variants (RLAIF [Bai *et al.*, 2022b; Lee *et al.*, 2023], RLHAIF [Wu *et al.*, 2021; Saunders *et al.*, 2022; Perez *et al.*, 2023]), as well as DPO-based methods [Rafailov *et al.*, 2024b; Azar *et al.*, 2024; Wang *et al.*, 2023a]. Additional approaches in this category include Rain and *Aligner*'s comprehensive sentence correction mechanisms, and RAFT and RRHF's high-quality sentence filtering strategies.

Token-Level Alignment: Recently, there has been increased interest in token-level alignment, decomposing the task into sequential token generation decisions for enhanced granularity and control. Both RLHF [Zhong *et al.*, 2024] and DPO [Zeng *et al.*, 2024; Rafailov *et al.*, 2024a] have evolved to support token-level alignment through MDP formulations, enabling more precise intervention at each generation step. MARA uniquely decouples the alignment model from the language model while operating at the token level. This novel approach significantly reduces computational costs compared

to existing coupled alignment methods, making MARA the first efficient decoupled token-level alignment framework.

6 Conclusion

This paper proposes MARA, a micro alignment approach that enhances LLMs' adherence to human preferences through token-level control. Our key innovation lies in introducing a micro alignment model that operates independently from the base language model, making **Accept/Reject** decisions for candidate tokens to achieve fine-grained alignment. Implemented as a compact three-layer fully-connected network, MARA significantly reduces computational overhead compared to existing SOTA approaches while maintaining superior alignment performance. Extensive experiments across multiple evaluation datasets and LLM architectures demonstrate MARA's effectiveness.

Ethics Statement

Our research on MARA presents a novel approach to language model alignment that significantly reduces computational requirements while maintaining alignment effectiveness. Our experimental validation utilizes publicly available datasets like PKU-SafeRLHF, BeaverTails and HarmfulQA, which are designed to promote helpful, harmless, and honest AI responses. While our efficient alignment approach could theoretically be misused for harmful purposes, we strongly oppose any malicious applications and advocate for the responsible development of alignment techniques.

To promote transparency and reproducibility, we release our complete implementation code under open-source licensing. To maintain anonymity during the review process, we temporarily withhold our trained models and training logs to avoid disclosing author information. Upon acceptance, we will release all materials on the Hugging Face platform. We encourage the AI community to build upon our work while maintaining strict ethical standards, with the goal of making AI alignment more accessible while serving human values and social good.

Acknowledgments

This work was supported by the Innovation and Technology Commission - Mainland-Hong Kong Joint Funding Scheme (Grant No. MHP/038/23).

Contribution Statement

Yang Zhang and Yu Yu contribute to this work equally.

References

[AI@Meta, 2024] AI@Meta. Llama 3 model card. 2024.

[Aligners, 2024] Aligners. aligner/aligner-7b-v1.0 · Hugging Face — huggingface.co. https://huggingface.co/aligner/aligner-7b-v1.0, 2024. [Accessed 24-01-2025].

[Anwar *et al.*, 2024] Usman Anwar, Abulhair Saparov, Javier Rando, Daniel Paleka, Miles Turpin, Peter Hase, Ekdeep Singh Lubana, Erik Jenner, Stephen Casper, Oliver Sourbut, et al. Foundational challenges in assuring alignment and safety of large language models. *arXiv preprint arXiv:2404.09932*, 2024.

[Azar *et al.*, 2024] Mohammad Gheshlaghi Azar, Zhaohan Daniel Guo, Bilal Piot, Remi Munos, Mark Rowland, Michal Valko, and Daniele Calandriello. A general theoretical paradigm to understand learning from human preferences. In *International Conference on Artificial Intelligence and Statistics*, pages 4447–4455. PMLR, 2024.

[Bai *et al.*, 2022a] 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.

[Bai *et al.*, 2022b] Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, et al. Constitutional ai: Harmlessness from ai feedback. *arXiv preprint arXiv:2212.08073*, 2022.

[Bhardwaj and Poria, 2023] Rishabh Bhardwaj and Soujanya Poria. Red-teaming large language models using chain of utterances for safety-alignment. *arXiv preprint arXiv:2308.09662*, 2023.

[Cui *et al.*, 2023] Ganqu Cui, Lifan Yuan, Ning Ding, Guanming Yao, Wei Zhu, Yuan Ni, Guotong Xie, Zhiyuan Liu, and Maosong Sun. Ultrafeedback: Boosting language models with high-quality feedback. *arXiv e-prints*, pages arXiv–2310, 2023.

[Dai *et al.*, 2024] 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.

[Dong *et al.*, 2023] Hanze Dong, Wei Xiong, Deepanshu Goyal, Yihan Zhang, Winnie Chow, Rui Pan, Shizhe Diao, Jipeng Zhang, SHUM KaShun, and Tong Zhang. Raft: Reward ranked finetuning for generative foundation model alignment. *Transactions on Machine Learning Research*, 2023.

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

[Haarnoja *et al.*, 2018] Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *International conference on machine learning*, pages 1861–1870. PMLR, 2018.

[Ji *et al.*, 2024a] Jiaming Ji, Boyuan Chen, Hantao Lou, Donghai Hong, Borong Zhang, Xuehai Pan, Tianyi Qiu, Juntao Dai, and Yaodong Yang. Aligner: Efficient alignment by learning to correct. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

[Ji *et al.*, 2024b] Jiaming Ji, Donghai Hong, Borong Zhang, Boyuan Chen, Josef Dai, Boren Zheng, Tianyi Qiu, Boxun Li, and Yaodong Yang. Pku-saferlhf: Towards multi-level safety alignment for llms with human preference. *arXiv preprint arXiv:2406.15513*, 2024.

[Ji *et al.*, 2024c] Jiaming Ji, Mickel Liu, Josef Dai, Xuehai Pan, Chi Zhang, Ce Bian, Boyuan Chen, Ruiyang Sun, Yizhou Wang, and Yaodong Yang. Beavertails: Towards improved safety alignment of llm via a human-preference dataset. *Advances in Neural Information Processing Systems*, 36, 2024.

[Jiang *et al.*, 2023] 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*, 2023.

[Kang *et al.*, 2020] Xiaomian Kang, Yang Zhao, Jiajun Zhang, and Chengqing Zong. Dynamic context selection for document-level neural machine translation via reinforcement learning. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2242–2254, 2020.

[Lee *et al.*, 2023] Harrison Lee, Samrat Phatale, Hassan Mansoor, Thomas Mesnard, Johan Ferret, Kellie Lu, Colton Bishop, Ethan Hall, Victor Carbune, Abhinav Rastogi, et al. Rlaif: Scaling reinforcement learning from human feedback with ai feedback. *arXiv preprint arXiv:2309.00267*, 2023.

[Li *et al.*, 2024] Yuhui Li, Fangyun Wei, Jinjing Zhao, Chao Zhang, and Hongyang Zhang. Rain: Your language models can align themselves without finetuning. In *The Twelfth International Conference on Learning Representations*, 2024.

[Nakano *et al.*, 2021] Reiichiro Nakano, Jacob Hilton, Suchir Balaji, Jeff Wu, Long Ouyang, Christina Kim, Christopher Hesse, Shantanu Jain, Vineet Kosaraju, William Saunders, et al. Webgpt: Browser-assisted question-answering with human feedback. *arXiv preprint arXiv:2112.09332*, 2021.

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

[Perez *et al.*, 2023] Ethan Perez, Sam Ringer, Kamilė Lukošiūtė, Karina Nguyen, Edwin Chen, Scott Heiner, Craig Pettit, Catherine Olsson, Sandipan Kundu, Saurav Kadavath, et al. Discovering language model behaviors with model-written evaluations. In *61st Annual Meeting of the Association for Computational Linguistics, ACL 2023*, pages 13387–13434. Association for Computational Linguistics (ACL), 2023.

[Rafailov *et al.*, 2024a] Rafael Rafailov, Joey Hejna, Ryan Park, and Chelsea Finn. From r to q^*: Your language model is secretly a q-function. *arXiv preprint arXiv:2404.12358*, 2024.

[Rafailov *et al.*, 2024b] 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. *Advances in Neural Information Processing Systems*, 36, 2024.

[Saunders *et al.*, 2022] William Saunders, Catherine Yeh, Jeff Wu, Steven Bills, Long Ouyang, Jonathan Ward, and Jan Leike. Self-critiquing models for assisting human evaluators. *arXiv preprint arXiv:2206.05802*, 2022.

[Stiennon *et al.*, 2020] 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. *Advances in Neural Information Processing Systems*, 33:3008–3021, 2020.

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

[Wang *et al.*, 2023a] Chaoqi Wang, Yibo Jiang, Chenghao Yang, Han Liu, and Yuxin Chen. Beyond reverse kl: Generalizing direct preference optimization with diverse divergence constraints. *arXiv preprint arXiv:2309.16240*, 2023.

[Wang *et al.*, 2023b] Yufei Wang, Wanjun Zhong, Liangyou Li, Fei Mi, Xingshan Zeng, Wenyong Huang, Lifeng Shang, Xin Jiang, and Qun Liu. Aligning large language models with human: A survey. *arXiv preprint arXiv:2307.12966*, 2023.

[Wu *et al.*, 2021] Jeff Wu, Long Ouyang, Daniel M Ziegler, Nisan Stiennon, Ryan Lowe, Jan Leike, and Paul Christiano. Recursively summarizing books with human feedback. *arXiv preprint arXiv:2109.10862*, 2021.

[Yuan *et al.*, 2023] Zheng Yuan, Hongyi Yuan, Chuanqi Tan, Wei Wang, Songfang Huang, and Fei Huang. Rrhf: Rank responses to align language models with human feedback without tears. *arXiv preprint arXiv:2304.05302*, 2023.

[Zeng *et al.*, 2024] Yongcheng Zeng, Guoqing Liu, Weiyu Ma, Ning Yang, Haifeng Zhang, and Jun Wang. Token-level direct preference optimization. In *Forty-first International Conference on Machine Learning*, 2024.

[Zheng *et al.*, 2024] Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, Zheyan Luo, Zhangchi Feng, and Yongqiang Ma. Llamafactory: Unified efficient fine-tuning of 100+ language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, Bangkok, Thailand, 2024. Association for Computational Linguistics.

[Zhong *et al.*, 2024] Han Zhong, Guhao Feng, Wei Xiong, Li Zhao, Di He, Jiang Bian, and Liwei Wang. Dpo meets ppo: Reinforced token optimization for rlhf. *arXiv preprint arXiv:2404.18922*, 2024.

[Ziegler *et al.*, 2019] 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*, 2019.

Appendices

A Experiments Setup

A.1 Parameter Setting

The parameter settings for the presented algorithms are presented in Tables 6.

Table 6: Parameter Settings for MARA

Parameters	Values
Training episodes	20000
Number of trajectories collection workers	7
Batch size	1024
Learning rate of actor network	0.0003
Learning rate of critic network	0.0003
Learning rate of entropy coefficient	0.0003
KL divergence coefficient λ	0.1
Initial entropy coefficient	0.8
Discount factor	0.99
Buffer capacity	1e6
Network layers	3
Hidden size	[4096, 1024, 256]
Target network update rate	0.005
Max response length limit	512
p of Top-p	0.95
k of top-k	50

A.2 Evaluation Metrics

To systematically compare different alignment approaches, we utilize the scores from reward models: beaver-7b-v1.0-reward for helpfulness evaluation and beaver-7b-v1.0-cost for harmlessness assessment. The comparison methodology operates as follows:

- A win (N_w +1) is recorded when an alignment approach achieves higher scores in both helpfulness and harmlessness dimensions
- A lose (N_l +1) is registered when an approach scores lower in both dimensions
- A tie (N_e +1) is counted in all other scenarios where the superiority is not conclusive

The rationale behind this comparison methodology stems from our observation that many aligned models simply refuse to respond to adversarial prompts. While such behavior ensures safety, it compromises utility. Therefore, we propose a more rigorous metric where a response is considered superior only when it demonstrates both enhanced helpfulness and harmlessness simultaneously.

B More Experiment Results

B.1 Detailed Results about Table 1 - 2

Table 7 presents the detailed results about Table 1 in terms of the win, tie, and lose number when applying MARA compared to using the original LLM alone. For evaluation, each entry presents three numbers: the number of times MARA outperforms the original LLM (win), the number of times they perform equally well (tie), and the number of times MARA underperforms (lose). For example, when applying MARA to Llama 3-8B on the SafeRLHF dataset, it achieves better performance in 93 cases, equal performance in 80 cases, and worse performance in 26 cases compared to using Llama 3-8B alone.

Similarly, Table 8 - 10 present the detailed results about Table 2 in terms of the win, tie, and lose number when comparing MARA to other alignment approaches in PKU-SafeRLHF, BeaverTails, and HarmfulQA datasets, respectively. For instance, when comparing MARA with RLHF on the BeaverTails dataset using Llama 3-8B, MARA achieves better performance in 251 cases, equal performance in 353 cases, and worse performance in 96 cases. These comprehensive statistics across different datasets and baseline methods demonstrate the consistent effectiveness of our proposed MARA approach across various model scales and alignment baselines.

B.2 The Ablation on the Size of Candidate Token Set.

To investigate how the candidate token set size influences model performance, we conducted ablation experiments with different Top-p and Top-k configurations using Llama-3.1-8B and Llama-3.2-1B. Table 11 shows that increasing p and k values leads to larger candidate token sets, with lengths growing from 6 to 27 tokens for Llama-3.1-8B and 6 to 30 tokens for Llama-3.2-1B at k=40. Notably, larger candidate sets generally yield better performance: Llama-3.1-8B's preference rate improvement increases from +26.63% (k=10, p=0.7) to +44.22% (k=40, p=0.9), while Llama-3.2-1B's improvement rises from +21.11% (k=10, p=0.7) to +33.67% (k=40, p=0.9) as the candidate set expands. This enhancement can be attributed to the increased diversity in token selection enabled by larger candidate sets.

B.3 The Ablation on KL Divergence Coefficients

To investigate the influence of KL divergence coefficient λ on model performance, we conduct experiments with different λ values (0.01, 0.1, 0.5, and 1.0) using Llama-3.2-1B and Mistral-7B-v0.3 models. As shown in Figure 4, we track both reward model scores and cost model scores during training. For Llama-3.2-1B (Figure 4a), while different λ values lead to varying training dynamics initially, all configurations eventually converge to similar reward scores around 5 and cost scores around -10. Similarly, Mistral-7B-v0.3 (Figure 4b) exhibits convergence behavior where different λ values ultimately reach comparable performance levels, with reward scores stabilizing around 8 and cost scores around -12. This convergence phenomenon suggests that the final model performance is relatively robust to the choice of λ, though the training stability and convergence speed differ. Based on the training dynamics and stability considerations, we choose $\lambda = 0.1$ as our default setting, which offers a good balance between convergence speed and training stability.

B.4 Visualisation Resuls

Figures 5 - 7 present the complete visualization results of the distribution of accepted tokens across the candidate token set

Table 7: Performance improvements of MARA across PKU-SafeRLHF, BeaverTails, and HarmfulQA datasets. Each entry shows the win, tie, or lose number when applying MARA compared to using the original LLM alone.

	Upstream LLM + MARA *vs.* Upstream LLM								
	SafeRLHF			BeaverTails			HarmfulQA		
Upstream LLM	Win	Tie	Lose	Win	Tie	Lose	Win	Tie	Lose
Llama 3-8B	93	80	26	355	248	97	296	150	42
Llama 3.1-8B	98	82	19	335	278	87	331	129	28
Llama 3.2-1B	104	81	14	334	254	112	329	120	39
Llama 3.2-3B	98	68	33	294	318	88	236	189	63
Mistral-7B-v0.1	44	133	22	200	422	78	93	353	42
Mistral-7B-v0.2	43	122	34	200	400	100	100	312	76
Mistral-7B-v0.3	59	115	25	206	400	94	97	348	43

Table 8: Performance comparison of MARA against RLHF, DPO, and *Aligner* on PKU-SafeRLHF dateset. Each entry shows the win, tie, or lose number when applying MARA compared to other alignment methods.

	MARA *vs.* RLHF			MARA *vs.* DPO			MARA *vs. Aligner*		
Upstream LLM	Win	Tie	Lose	Win	Tie	Lose	Win	Tie	Lose
Llama 3-8B	81	96	22	65	99	35	52	106	41
Llama 3.1-8B	68	106	25	43	113	43	38	124	37
Llama 3.2-1B	89	99	11	47	106	46	27	133	39
Llama 3.2-3B	50	116	33	54	93	52	37	121	41
Mistral-7B-v0.1	36	143	20	49	120	30	45	130	24
Mistral-7B-v0.2	66	116	17	68	108	23	56	116	27
Mistral-7B-v0.3	49	129	21	60	113	26	53	125	21

on PKU-SafeRLHF dataset with Llama-3.2-1B, Llama-3.1-8B, and Mistral-7B-v0.3, respectively. From the complete results, the three same findings derived by Figure 3 (dominance of first token, parameter sensitivity, and optimal decision window) can still be observed from Figures 5 - 7.

These patterns remain consistent across different model architectures and scales, suggesting they are inherent characteristics of the MARA approach rather than model-specific phenomena.

Table 9: Performance comparison of MARA against RLHF, DPO, and *Aligner* on BeaverTails dateset. Each entry shows the win, tie, or lose number when applying MARA compared to other alignment methods.

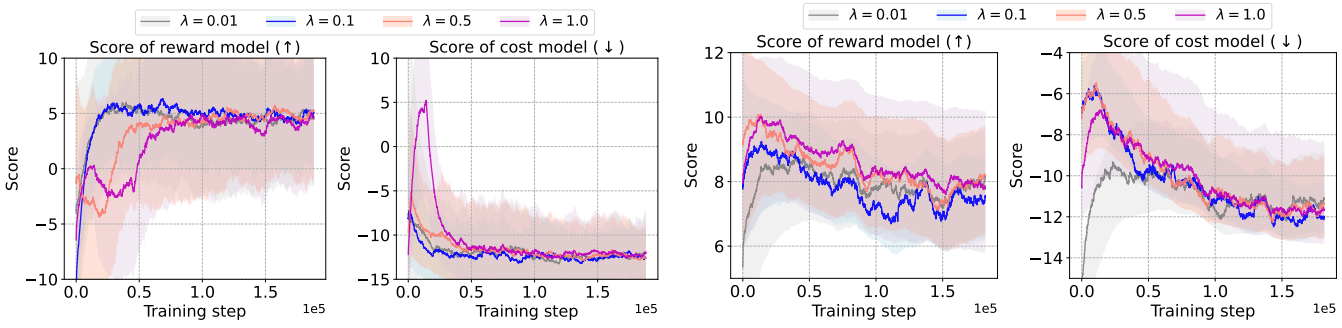
	MARA *vs.* RLHF			MARA *vs.* DPO			MARA *vs. Aligner*		
Upstream LLM	Win	Tie	Lose	Win	Tie	Lose	Win	Tie	Lose
Llama 3-8B	251	353	96	254	296	150	172	375	153
Llama 3.1-8B	259	378	63	202	398	100	120	438	142
Llama 3.2-1B	228	400	72	181	339	180	97	422	181
Llama 3.2-3B	189	374	137	179	361	160	120	413	167
Mistral-7B-v0.1	127	445	128	221	379	100	195	429	76
Mistral-7B-v0.2	191	436	73	263	363	74	197	400	103
Mistral-7B-v0.3	191	436	73	278	353	69	226	383	91

Table 10: Performance comparison of MARA against RLHF, DPO, and *Aligner* on HarmfulQA dateset. Each entry shows the win, tie, or lose number when applying MARA compared to other alignment methods.

	MARA *vs.* RLHF			MARA *vs.* DPO			MARA *vs. Aligner*		
Upstream LLM	Win	Tie	Lose	Win	Tie	Lose	Win	Tie	Lose
Llama 3-8B	238	232	18	219	188	81	105	306	77
Llama 3.1-8B	121	274	93	113	270	105	76	278	134
Llama 3.2-1B	233	212	43	173	215	100	91	259	138
Llama 3.2-3B	93	297	98	110	244	134	73	279	136
Mistral-7B-v0.1	58	372	58	96	326	66	66	359	63
Mistral-7B-v0.2	113	345	30	156	283	49	110	289	89
Mistral-7B-v0.3	144	330	14	163	279	46	108	328	52

Table 11: Experimental analysis of how token candidate length affects model alignment performance through Top-k and Top-p sampling strategies on PKU-SafeRLHF dataset. The results measure the performance improvements (in percentage) over the upstream model in terms of preference rate.

		Top k=10			Top k=20			Top k=40		
Upstream LLM	Metrics	p=0.7	p=0.8	p=0.9	p=0.7	p=0.8	p=0.9	p=0.7	p=0.8	p=0.9
Llama-3.1-8B	Length	6	7	8	10	12	14	14	19	27
	Preference rate	+26.63%	+28.64%	+40.70%	+28.64%	+28.64%	+45.23%	+28.64%	+30.65%	+44.22%
Llama-3.2-1B	Length	6	7	9	10	13	16	16	20	30
	Preference rate	+21.11%	+19.60%	+34.17%	+25.13%	+30.15%	+37.69%	+20.60%	+34.17%	+33.67%



(a) Llama-3.2-1B

(b) Mistral-7B-v0.3

Figure 4: Ablation experiments for the selection of KL divergence coefficient λ on the algorithm performance. The experiments are performed on PKU-SafeRLHF dataset with different upstream models.

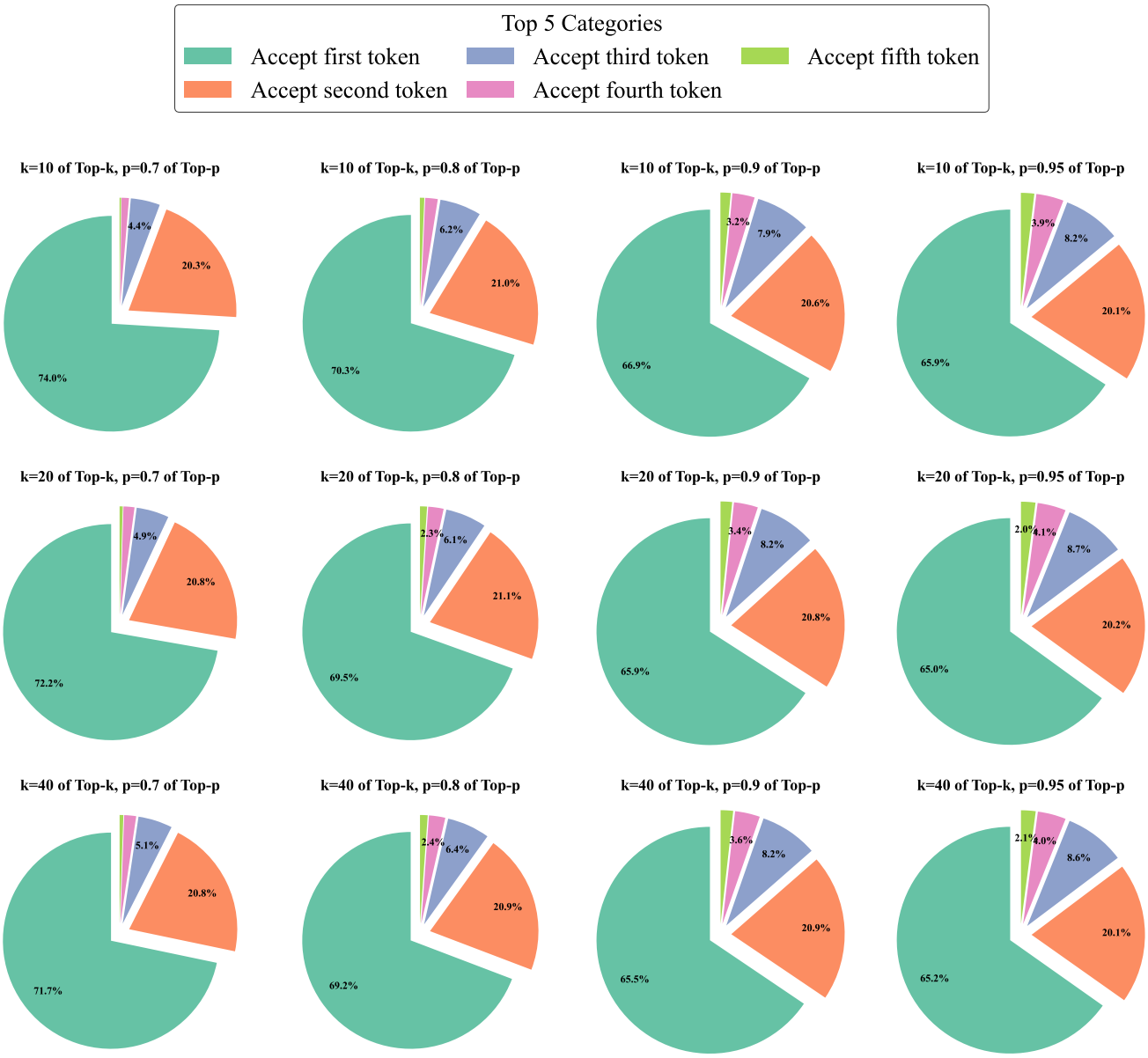


Figure 5: Distribution of accepted tokens across the candidate token set on PKU-SafeRLHF dataset with Llama-3.2-1B as the upstream LLM. Token positions with acceptance probabilities below 1% are omitted for clarity.

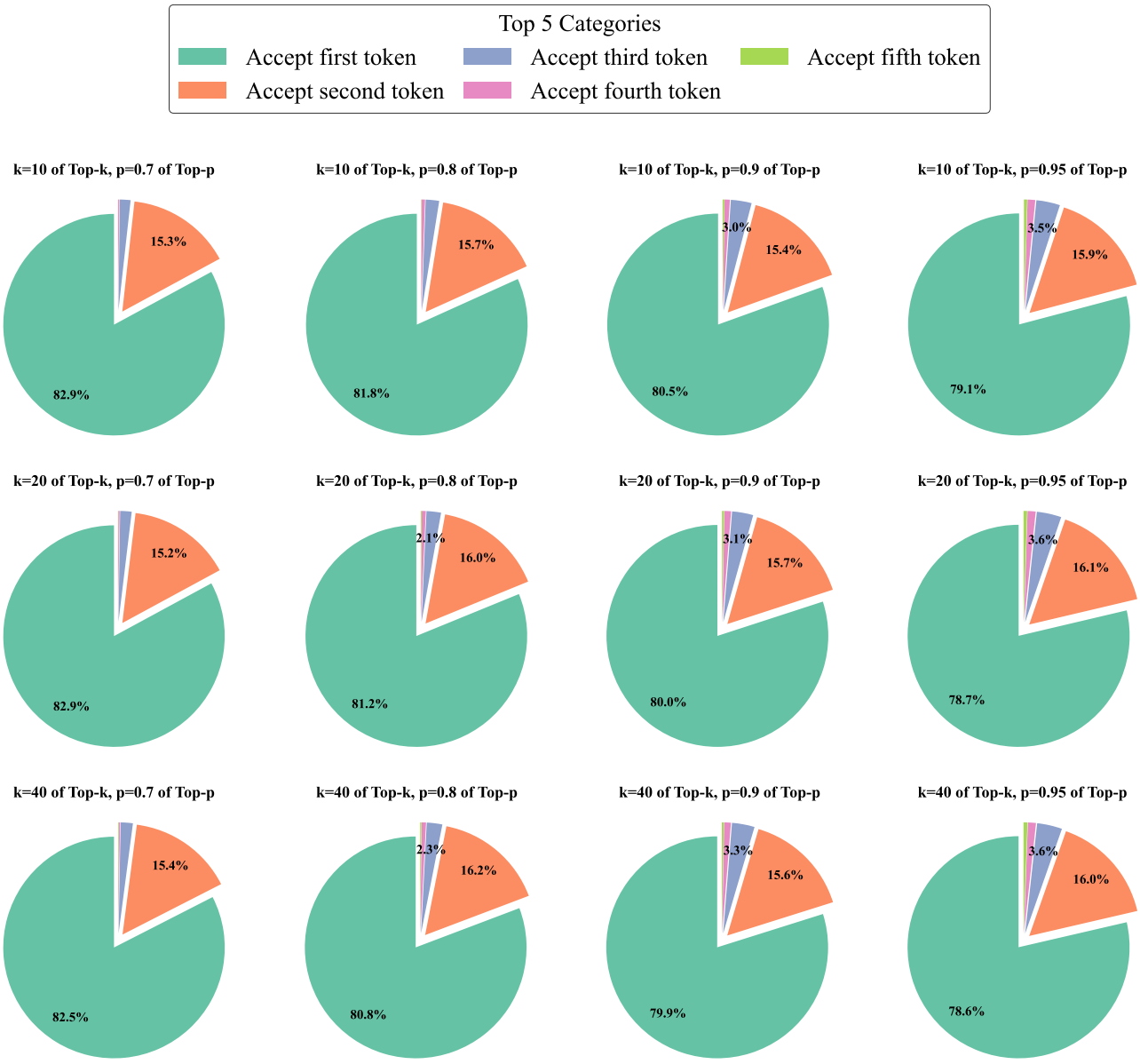


Figure 6: Distribution of accepted tokens across the candidate token set on PKU-SafeRLHF dataset with Llama-3.1-8B as the upstream LLM. Token positions with acceptance probabilities below 1% are omitted for clarity.

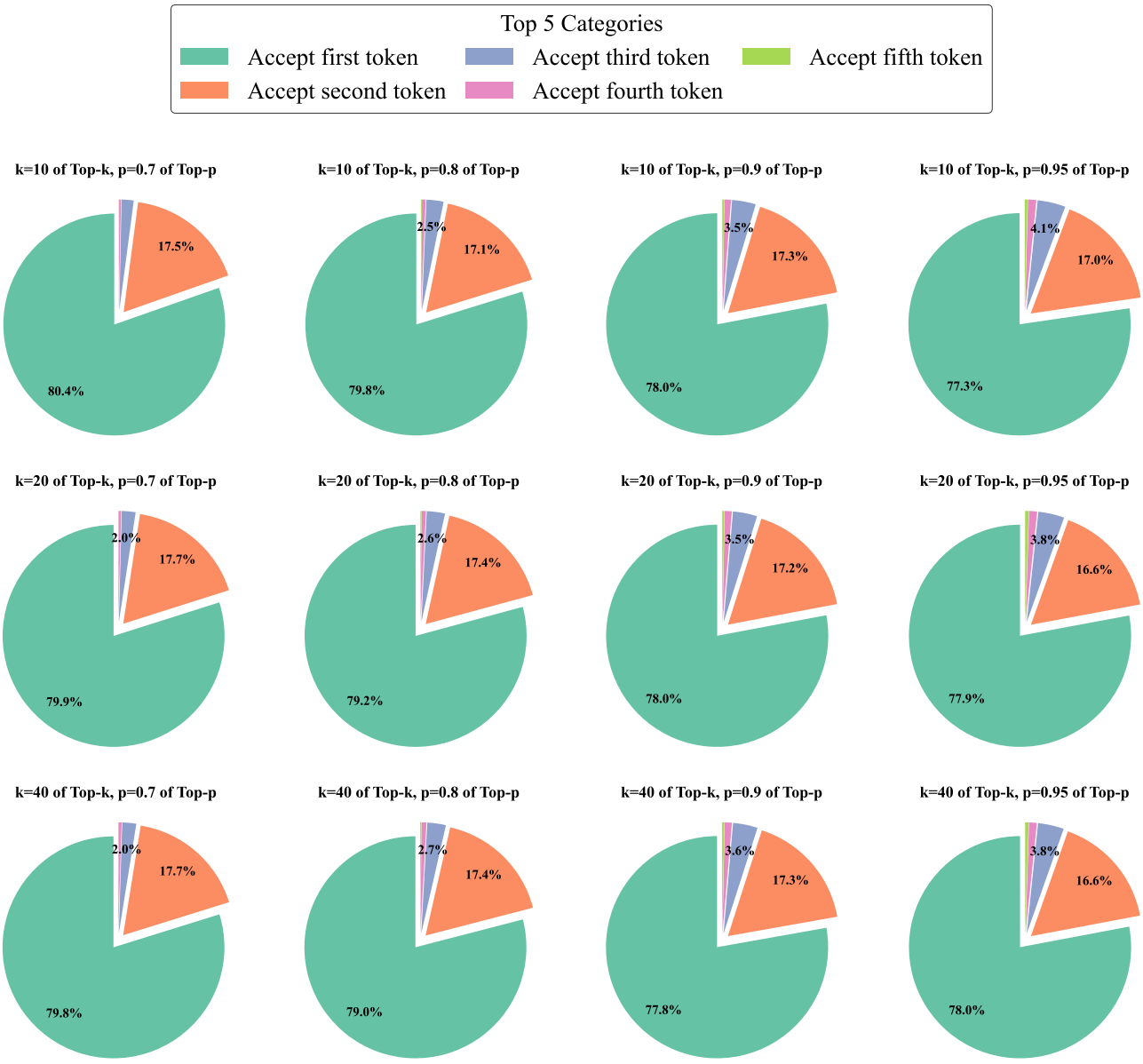


Figure 7: Distribution of accepted tokens across the candidate token set on PKU-SafeRLHF dataset with Mistral-7B-v0.3 as the upstream LLM. Token positions with acceptance probabilities below 1% are omitted for clarity.