MATO: Multi-objective Personalized Alignment with Test-time Optimization for Large Language Models

Linhao Luo[1] **Thuy-Trang Vu**[1] **Van-Anh Nguyen**[1] **Junae Kim**[2]
Gholamreza Haffari[1] **Dinh Phung**[1]
[1]Monash University [2] Defence Science and Technology Group, Australia
{linhao.luo,trang.vu1,van-anh.nguyen}@monash.edu
junae.kim@defence.gov.au
{gholamreza.haffari,dinh.phung}@monash.edu

Abstract

Aligning large language models (LLMs) with diverse and multifaceted user preferences is a fundamental challenge in personalized AI systems. Existing multi-objective alignment methods either rely on costly training or require pre-trained reward models for each preference, making it difficult for them to adapt to evolving preferences. Prompt-based personalization offers a training-free alternative, but prompting alone often provides limited steerability, as LLMs may overemphasize or overlook certain preferences and fail to give users reliable control over the relative importance of different objectives when conflicts arise, leading to suboptimal alignment. In this paper, we introduce MATO, a training-free framework for **M**ulti-objective personalized **A**lignment with **T**est-time **O**ptimization. MATO formulates personalization as a test-time optimization problem that steers the relative importance of multiple objectives through controllable weights during decoding, without modifying model parameters or requiring external reward models. Specifically, a *reward discovery* module constructs prompt-derived surrogate rewards from the backbone LLM for diverse objectives specified in natural language, while a *weight optimization* module dynamically adjusts objective weights based on the user's initial preferences and the partially generated response to balance competing objectives during generation. The resulting surrogate rewards and weights jointly guide an *online optimization* procedure over the token distribution, enabling better alignment with the target objectives. Extensive experiments across multiple datasets and backbone LLMs show that MATO consistently outperforms strong baselines, achieving Pareto-improving multi-objective alignment and stronger steerability. These results highlight test-time optimization as a promising direction for scalable, controllable, and model-agnostic personalized alignment. Code is available at https://github.com/rmanluo/MATO.

1 Introduction

Large language models (LLMs) have emerged as general-purpose assistants that support a wide range of user needs, including writing [28], teaching [19], coding [11], and planning [5]. Their growing applications make it increasingly important to align their behavior with user preferences and values [14]. Mainstream alignment methods, such as reinforcement learning from human feedback (RLHF) [20], typically learn from feedback aggregated across users, which tends to collapse diverse preferences into a single reward signal, making them less suited to personalized alignment [22]. In practice, user preferences are inherently diverse and multifaceted, shaped by differences in background, goals, and values [3, 33]. This challenge has motivated growing interest in multi-

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

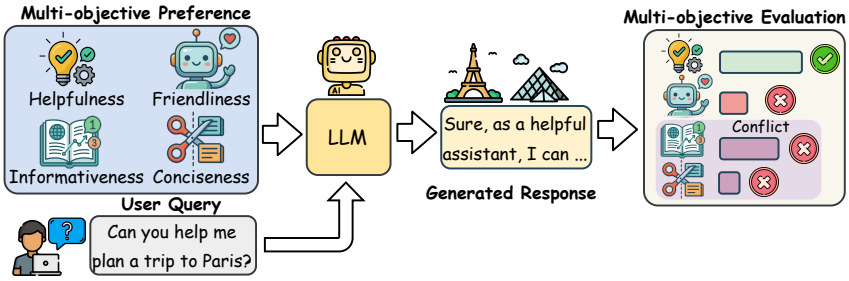


Figure 1: Illustration of multi-objective personalized alignment and its challenges.

objective personalized alignment [37, 31] that aligns LLMs with diverse user preferences, involving multiple and potentially competing objectives.

Early research on multi-objective personalized alignment extends RLHF to optimize LLMs on linearly combined multi-preference rewards [13, 25, 37], which requires extensive training of separate models for different preference combinations. More recent work has explored test-time alignment methods [15, 4] that adapt LLMs to multiple objectives without additional LLM training. However, these methods still require pre-trained reward models for each preference, which can be hard to obtain as user preferences evolve and new objectives emerge [7].

Prompt-based personalization is appealing because it is training-free and can adapt on the fly across LLMs and objectives [38, 8]. Recent instruction-following LLMs can often adapt their outputs to preference descriptions expressed in natural language in the prompt [12, 16]. However, prompting alone is often insufficient for robust multi-objective alignment [27]. As shown in Figure 1, when users specify multiple objectives, LLMs may over-prioritize some preferences (e.g., helpfulness), neglect others (e.g., friendliness), or behave inconsistently when objectives conflict (e.g., informativeness vs. conciseness). More fundamentally, prompt conditioning offers limited steerability [6, 26], as LLMs may fail to infer the relative importance of objectives and balance them reliably, leading to suboptimal alignment and poor trade-offs among preferences [35].

In this paper, we introduce MATO, a training-free framework for **M**ulti-objective personalized **A**lignment via **T**est-time **O**ptimization. Rather than encoding preference control in model parameters or relying solely on natural-language prompts, MATO formulates personalization as a test-time optimization problem, allowing users to steer the relative importance of multiple objectives through controllable weights during LLM decoding. To remove the need for external reward models, we propose a *reward discovery* module that constructs prompt-derived surrogate rewards from the backbone LLM for objectives described in natural language. We further introduce a *weight optimization* module that dynamically adjusts objective weights based on the user's initial preferences and the partially generated response to balance preferences during generation. The surrogate rewards and weights are used to guide an *online optimization* procedure over the model's output distribution to achieve better alignment with the target objectives. This design better handles competing preferences, exposes a steerable multi-objective trade-off, and enables Pareto-improving alignment. Extensive experiments across multiple datasets and backbone LLMs show that MATO consistently outperforms strong baselines in overall alignment quality while providing stronger steerability. These results highlight test-time optimization as a promising direction for scalable, controllable, and model-agnostic personalized alignment.

2 Related Work

Training-based personalized alignment. Prior work formulates personalized alignment as multi-objective optimization by extending RLHF with linearly combined rewards (MORL) [13]. Preference optimization methods such as MODPO [37] improve stability and efficiency over MORL, while approaches like Rewarded Soups [25] and RiC [31] reduce the need for training separate models by interpolating weights or conditioning on preference vectors. Recently, test-time alignment methods keep the backbone LLM fixed and guide decoding with reward models [4, 29]. These include logit-level merging across objectives (MOD) [17], unified preference-aware reward modeling (PARM) [15], and robust decoding strategies (RMOD) [27]. Despite improved efficiency, these approaches still rely

on pre-trained reward models for each objective, limiting adaptability as preferences evolve or new objectives emerge [7].

Training-free personalized alignment. The training-free paradigm has gained increasing attention for personalized alignment due to its flexibility and efficiency. Prompt-based methods exploit instruction-following to express user preferences in natural language [12], while decoding-based techniques steer generation via preference signals, including alignment directions (Linear Alignment) [6], interpolation of prompt-conditioned outputs (CoS) [8], divergence-based rewards (OPAD) [38], and online token distribution updates (Amulet) [32]. Despite their flexibility and efficiency, these methods are primarily designed for single-objective alignment and struggle to manage trade-offs among multiple preferences. In contrast, our MATO formulates training-free personalized alignment as a multi-objective optimization problem at test time, enabling balanced and adaptive preference control. Detailed comparisons with these methods are provided in Table 4.

3 Preliminary

Multi-objective Personalized Alignment. We study explicit personalized alignment, where the target objectives and their relative importance are supplied as inputs to a language model π. The implicit personalization [35] where preferences must first be inferred, is outside our scope. Given a query x, the model generates $y \sim \pi(\cdot \mid x)$. Let $\mathcal{C} = \{c_k\}_{k=1}^{K}$ denote natural-language objective descriptions, e.g., "use concise language" or "explain concepts for a beginner," with corresponding reward functions R_k and non-negative weights w_k ($\sum_k w_k = 1$). The supplied initial weights are denoted by $\mathcal{W}^{\text{init}}$ in Section 4.2. The goal is to maximize the weighted reward:

$$y^* = \arg\max_{y \sim \pi(\cdot|x)} \quad \sum_{k=1}^{K} w_k R_k(x, y). \tag{1}$$

This formulation captures the trade-offs among preferences, while the weights allow users to control their relative importance and target a personalized response.

Test-time Alignment. Based on the objective defined in Equation (1), a direct approach is to apply multi-objective reinforcement learning (MORL) by optimizing the LLM on a weighted combination of preference rewards [13, 25, 37], formulated as:

$$\pi^* = \arg\max_{\pi} \mathbb{E}_{y \sim \pi(\cdot|x)} \left[\sum_{k=1}^{K} w_k R_k(x, y) \right] - \beta D_{\text{KL}}(\pi(\cdot \mid x) || \pi_{\text{base}}(\cdot \mid x)). \tag{2}$$

This requires training separate models for different preference combinations. Test-time methods [29, 15] instead guide the decoding of a fixed LLM π_{base} with token-level reward models $R_k(y_i|x, y_{<i})$ for each preference as:

$$\pi^*(y_i \mid x, y_{<i}) \propto \pi_{\text{base}}(y_i \mid x, y_{<i}) \exp\left(\frac{1}{\beta} \sum_{k=1}^{K} w_k R_k(y_i \mid x, y_{<i}) \right), \tag{3}$$

where y_i is the i-th token in the generated response. Despite their effectiveness, these methods still require pre-trained token-level reward models for each preference, which are costly to obtain as user preferences evolve and new objectives emerge [7].

4 Approach

In this section, we introduce MATO, a training-free framework for multi-objective personalized alignment with test-time optimization. As shown in Figure 2, MATO consists of three key components: (1) *reward discovery*, (2) *weight optimization*, and (3) *online optimization*, which we detail below.

4.1 Reward Discovery

Test-time methods rely on pre-trained reward models R_k for each preference, which limits their flexibility [7]. To address this, a *reward discovery* module is introduced to construct prompt-derived

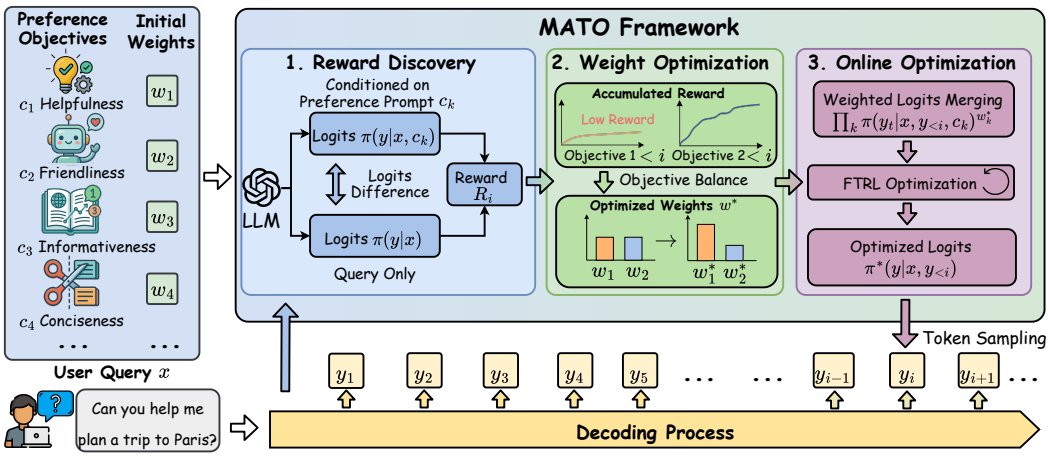


Figure 2: Overview of the MATO framework, which consists of three key components: (1) *reward discovery*, which constructs prompt-derived surrogate rewards from the backbone LLM, (2) *weight optimization*, which dynamically adjusts objective weights to balance preferences during generation, and (3) *online optimization*, which guides the output distribution using the surrogate rewards and optimized weights.

surrogate rewards directly from the backbone LLM, enabling decoding without additional training. The closed-form solution of the KL-regularized RL objective in Equation (2) motivates a token-level reward representation [23, 24]:

$$R_k(y_i|x, y_{<i}) := \beta \log \frac{\pi_k^*(y_i \mid x, y_{<i})}{\pi_{\text{base}}(y_i \mid x, y_{<i})}, \tag{4}$$

where π_k^* is the optimal policy for the k-th objective under KL regularization. Each term measures how consistent token y_i is with the k-th preference under the optimal policy, and they could be aggregated into a sequence-level reward (Section B.1).

Since π_k^* is unavailable, we use the backbone conditioned on a preference-specific prompt c_k as a proxy [38, 8] to approximate the optimal policy and obtain the *prompt-derived surrogate reward*:

$$\widehat{R}_k(y_i|x, y_{<i}) := \beta \log \frac{\pi_{\text{base}}(y_i \mid x, y_{<i}, c_k)}{\pi_{\text{base}}(y_i \mid x, y_{<i})}, \tag{5}$$

where c_k describes the k-th preference in natural language. This signal measures the quality of preference alignment without additional reward model training. Experiments in Section D.4 show that the surrogate reward is positively associated with alignment quality, supporting its use as a reward proxy. The prompt templates are given in Section E.1.

4.2 Weight Optimization

In multi-objective alignment, some preferences may dominate while others are neglected during generation, leading to suboptimal alignment performance [35, 27]. Thus, we aim to balance the preferences by increasing the weights of neglected preferences and decreasing the weights of dominant ones while maximizing the overall rewards, thereby ensuring Pareto-improving alignment during generation. To this end, we track the accumulated surrogate reward for each preference over the generated prefix, which provides a running proxy for objective satisfaction and enables dynamic weight adjustment. Specifically, we leverage the reward discovery module in Equation (5) to compute accumulated surrogate rewards by summing token-level surrogate rewards on generated tokens as

$$\widehat{R}_k(x, y_{<i}) = \sum_{j=1}^{i-1} \widehat{R}_k(y_j|x, y_{<j}), \tag{6}$$

which can be efficiently computed in parallel during decoding. Impact of other reward aggregation methods is studied in Section D.6.

Drawing on the adversarial weight-player view of max-min MORL [21, 2], we treat the preference weights as a simplex-constrained adversary that reweights the cumulative surrogate rewards at the current decoding prefix as:

$$\max_{\pi} \min_{\mathcal{W} \in \Delta^K} \sum_{k=1}^{K} w_k \widehat{R}_k(x, y_{<i}). \tag{7}$$

However, directly solving the resulting max-min problem at every decoding step is computationally expensive and typically requires iterative optimization over the simplex [27, 9]. To preserve user steerability while avoiding per-step iterative optimization, we introduce a personalized regularized weight subproblem anchored at the user's initial preference weights $\mathcal{W}^{\text{init}} = \{w_1^{\text{init}}, \ldots, w_K^{\text{init}}\} \in \Delta^K$:

$$\mathcal{W}^*(x, y_{<i}) = \arg \min_{\mathcal{W} \in \Delta^K} \sum_{k=1}^{K} w_k \widehat{R}_k(x, y_{<i}) + \tau D_{\text{KL}}(\mathcal{W} || \mathcal{W}^{\text{init}}), \tag{8}$$

where Δ^K is the probability simplex in $\mathbb{R}^K$, and τ controls how far the optimized weights can move away from the user prior $\mathcal{W}^{\text{init}}$.

Proposition 4.1. *For any generated prefix $y_{<i}$, if $\tau > 0$ and $w_k^{\text{init}} > 0$ for all k, the regularized weight subproblem in Equation* (8) *admits the unique closed-form solution*

$$w_k^* = \frac{w_k^{\text{init}} \cdot \exp\left(-\frac{\widehat{R}_k(x, y_{<i})}{\tau}\right)}{\sum_{j=1}^{K} w_j^{\text{init}} \exp\left(-\frac{\widehat{R}_j(x, y_{<i})}{\tau}\right)}, \tag{9}$$

A higher τ preserves more of the user prior, while a lower τ enables more aggressive reweighting toward under-optimized objectives. The detailed proof is provided in Section B.2.

4.3 Online Optimization

Injecting the surrogate rewards (eq. (5)) and optimized weights (eq. (9)) into Equation (3) yields a weighted decoding distribution: $\prod_{k=1}^{K} \pi_{\text{base}}(y_i \mid x, y_{<i}, c_k)^{w_k^*}$, we further multiply this term by a fixed multi-objective prompt-conditioned distribution as a stabilizing anchor, obtaining:

$$\pi'(y_i \mid x, y_{<i}, \mathcal{C}) \propto \pi_{\text{base}}(y_i \mid x, y_{<i}, \mathcal{C}) \cdot \prod_{k=1}^{K} \pi_{\text{base}}(y_i \mid x, y_{<i}, c_k)^{w_k^*}, \tag{10}$$

where $\mathcal{C}$ combines all preference descriptions and the user's initial weights into a fixed instruction prompt (Section E.2). This anchor keeps the merged distribution close to a coherent multi-objective instruction-following policy and improves decoding stability. The merged distribution π' combines the objective-specific surrogate signals under the optimized weights. We further apply online optimization to amplify this combined signal under KL regularization to move towards better alignment with the multi-objective preferences.

Specifically, we use the log-probability ratio between the current iterate and the base distribution as a token-level utility signal [38], with KL penalties toward the merged distribution π' and the previous iterate. These updates use only model-derived signals to push generation toward high-quality regions defined by $\mathcal{C}$, indirectly approaching optimal policy π^* [38] (see Section B.3). Starting from $\pi_0 = \pi'$, we perform T iterative updates using follow-the-regularized-leader (FTRL) [18]:

$$\pi_t = \arg\max_{\pi} \sum_{j=1}^{t-1} \mathcal{U}_j(\pi) - \frac{1}{\eta} D_{\text{KL}}(\pi || \pi_{t-1}),\ \pi_0 = \pi', \tag{11}$$

$$\mathcal{U}_j(\pi) = \sum_{a \in \mathcal{V}} \pi(a) u_j(a) - \lambda D_{\text{KL}}(\pi || \pi'), \tag{12}$$

$$u_j(a) = \alpha \left(\log \pi_j(a | x, y_{<i}) - \log \pi_{\text{base}}(a | x, y_{<i})\right), \quad a \in \mathcal{V}, \tag{13}$$

where $a \in \mathcal{V}$ denotes a token from the vocabulary $\mathcal{V}$. The utility function $\mathcal{U}$ (eq. (12)) encourages the updated policy π_t to maximize divergence from the base distribution π_{base} (eq. (13)) while regularizing

it to stay close to the merged distribution π'. The FTRL update (eq. (11)) ensures stability via a KL regularization term that penalizes large deviations from the previous policy π_{t-1}.

The optimization procedure in Equation (11) can be efficiently solved in closed-form [32] as

$$\pi_t(a) \propto \exp\left(\frac{1}{(t-1)\lambda\eta+1}\left(\eta\sum_{j=1}^{t-1}u_j(a)+(t-1)\lambda\eta\log\pi'(a)+\log\pi_{t-1}(a)\right)\right), \quad a\in\mathcal{V}. \tag{14}$$

A detailed derivation can be found in Section B.4. After T steps, we sample y_i from π_T to generate responses better aligned with multi-objective preferences, which is empirically assessed in Section 5.5. The full decoding process is in Algorithm 1.

5 Experiment

In our experiments, we aim to answer the following research questions: (1) How does MATO perform in multi-objective personalized alignment (Section 5.2)? (2) How does MATO enable steerability of the multi-objective trade-off (Section 5.3)? (3) How does MATO dynamically adjust weights to balance objectives during decoding (Section 5.4)? (4) How do different components and parameters of MATO contribute to its performance (Section 5.5)? (5) How efficient is MATO in terms of computational cost and latency (Section 5.6)?

Algorithm 1: LLM Decoding Process with MATO

Input: LLM π_{base}; query x; preference prompts $\mathcal{C}=\{c_1,\dots,c_K\}$; initial weights $\mathcal{W}^{\text{init}}=\{w_1^{\text{init}},\dots,w_K^{\text{init}}\}$; weight optimization temperature τ, online optimization steps T; hyperparameters α, λ, η

Output: Response y

$y \leftarrow \emptyset$
while *generation has not ended* **do**
  `// Reward Discovery.`
  Generate $\pi_{\text{base}}(\cdot \mid x,y)$, $\pi_{\text{base}}(\cdot \mid x,y,\mathcal{C})$, and $\pi_{\text{base}}(\cdot \mid x,y,c_k), \forall k\in\{1,\dots,K\}$, with LLM
  for $k=1,2,\dots,K$ **do**
    Compute the accumulated surrogate reward $\widehat{R}_k(x,y)$ by Equations (5) and (6)
  end
  `// Weight Optimization.`
  Update the optimized weights $\mathcal{W}^*$ by Equation (9)
  Construct the merged distribution π' by Equation (10)
  `// Online Optimization.`
  $\pi_0 \leftarrow \pi'$
  for $t=1,2,\dots,T$ **do**
    Update π_t with the FTRL objective in Equation (11)
  end
  Sample the next token $y_i \sim \pi_T(\cdot \mid x,y)$
  Append the sampled token to the response $y \leftarrow y \oplus y_i$
end
return the generated response y

5.1 Experimental Setup

Dataset. For multi-objective personalized alignment, we evaluate on the *Multifaceted* dataset [12], which defines 107 personalized user preferences across 4 dimensions and 18 subdimensions. It contains 945 user queries, each randomly assigned 4 preferences with uniform weights. The full list of preferences is provided in Table 5. For steerability evaluation, we use the *HH-RLHF* dataset [1], which has been widely used in previous multi-objective alignment work [31, 26]. We select the humor and helpfulness preferences to evaluate the steerability of multi-objective trade-off. The weights of the two preferences are varied from 1 to 0 with a step size of 0.2 to evaluate the model's ability to balance the two objectives under different weight combinations.

Evaluation Metrics. For multi-objective personalized alignment, we use GPT-4o and claude-sonnet-4-5 as the LLM-as-judge to score each response on each preference dimension ($s_k \in [1,5]$) based on the rubrics provided by the Multifaceted dataset [12]. To quantify the overall multi-objective alignment performance, we compute three metrics: (1) *all preference match rate* (AMR), which measures the percentage of responses that fully match all preferences (i.e.,

Table 1: Performance comparison on multi-objective personalized alignment using GPT-4o as judge.

Methods	Qwen3-0.6B			Qwen3-8B			Llama-3.2-1B			Llama-3.1-8B		
	AMR	APS	Worst	AMR	APS	Worst	AMR	APS	Worst	AMR	APS	Worst
Base LLM	0.14	2.58	1.49	0.43	3.66	2.34	0.21	2.78	1.69	0.34	3.36	2.04
Preference Prompt [12]	0.19	2.78	1.64	0.70	4.19	3.23	0.31	3.07	2.00	0.66	3.97	2.96
Linear Alignment [6]	0.24	2.80	1.76	0.72	4.19	3.31	0.37	2.78	2.11	0.64	3.87	2.98
CoS [8]	0.19	2.59	1.64	0.66	4.00	3.13	0.17	2.34	1.60	0.52	3.52	2.63
Amulet [32]	0.12	2.31	1.46	0.60	3.79	2.94	0.12	2.06	1.42	0.32	2.92	2.07
OPAD [38]	0.25	2.97	1.80	0.74	4.28	3.39	0.36	2.99	1.88	0.68	4.03	3.10
MATO	**0.27**	**2.99**	**1.87**	**0.75**	**4.30**	**3.42**	**0.39**	**3.23**	**2.25**	**0.73**	**4.08**	**3.22**

$\prod_{k=1}^{K} \mathbb{I}(s_k \geq 3) = 1$); (2) *average preference score* (APS), which measures the average score across all preferences; and (3) *worst preference score* (Worst), which measures the lowest score among all preferences. The prompt for the LLM-as-judge and detailed metric definitions are provided in Section C.2. For steerability evaluation, we use two pre-trained reward models to provide continuous reward scores for responses on helpfulness[1] and humor[2] preferences, which allow us to plot the Pareto front of the model's performance under different weight combinations. These reward models have been widely used in previous work [31, 26].

Baselines. We compare MATO with two groups of baselines: *training-free methods* and *training-based methods*.

Training-free methods require no gradient-based training and align LLMs with text-based prompts, including: *Base LLM*, which directly evaluates the base LLM on the questions without any alignment; *Preference Prompt* [12], which concatenates multiple preference descriptions into prompts for the base LLM; *Linear Alignment* [6], which performs test-time personalized alignment with a linear update on the token distribution; *CoS* [8], which amplifies the influence of the preference prompt by comparing two model outputs; *Amulet* [32], which conducts online learning on the distribution to align with a single preference and *OPAD* [38], which uses a surrogate preference-guided reward model to optimize the output distribution to align with human preferences.

Training-based methods require training on the target preferences, including: *MORLHF* [13], which uses reinforcement learning to train separate LLMs with different weight combinations on the rewards of the target preferences; *RiC* [31], which conducts supervised fine-tuning on the base LLM by concatenating multiple preferences and their weights as input prompts; and *MOD* [17], which trains several single-objective aligned LLMs and performs weighted decoding to balance the preferences. Because these methods require objective-specific training, they are not practically applicable to the Multifaceted dataset [12], which involves a large number of preferences.

Implementation Details. For training-free methods, we use four different base LLMs of varying sizes: `Qwen3-0.6B`, `Qwen3-8B`, `Llama-3.2-1B`, and `Llama-3.1-8B`. As most training-free methods are not designed for multi-objective personalized alignment, we adapt them to the multi-objective setting by concatenating multiple preference descriptions into a single prompt (Section E.2). For training-based methods, we use `Qwen3-8B` as the base LLM. We conduct training on the training set of the HH-RLHF dataset using the aforementioned pre-trained reward models. We evaluate the trained models on the test set of the HH-RLHF dataset following the same evaluation protocol as previous work [31, 26]. For MATO, we use the same base LLMs as in the training-free methods. We set the weight optimization temperature τ to 1, the number of online optimization steps T to 80, and $\alpha = 0.5, \lambda = 1.0, \eta = 10$.

5.2 Multi-objective Personalized Alignment Performance

Overall Performance. Table 1 compares multi-objective personalized alignment performance. MATO consistently outperforms all baselines across base LLMs and metrics, demonstrating its effectiveness and generalizability in aligning LLMs with multiple personalized preferences. Preference Prompt substantially improves over the Base LLM, showing that in-context learning from explicit preference descriptions is already an effective alignment signal, particularly for larger LLMs. However, Linear Alignment, CoS, and OPAD, which further optimize the token distribution based on

[1] `https://huggingface.co/Ray2333/gpt2-large-helpful-reward_model`

[2] `https://huggingface.co/mohameddhiab/humor-no-humor`

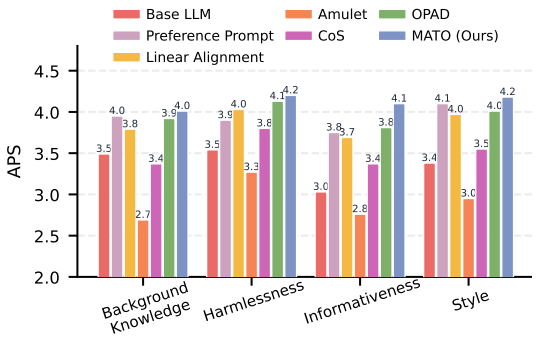


Figure 3: Average preference score across dimensions.

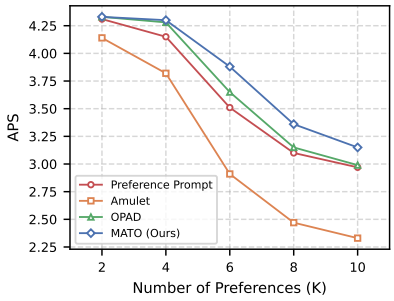


Figure 4: Average preference score under different aligned preferences.

the preference prompts during decoding, perform closely to Preference Prompt (e.g., identical APS of 4.19 on Qwen3-8B). This reveals that treating multiple preferences as a single combined signal cannot track or compensate for individual preference satisfaction. Moreover, their performance is even worse than Preference Prompt on smaller LLMs (i.e., Llama-3.2-1B, APS of 2.78 vs. 3.07), which may be attributed to the incorrect guidance from the combined signal, which misleads the optimization direction. Despite employing online learning, Amulet performs worst overall, as its single-objective optimization direction conflicts with the multi-objective setting and leads to misaligned generation.

In contrast, MATO constructs per-preference surrogate rewards and dynamically reweights underserved objectives, with online optimization amplifying the merged preference signal under KL regularization. The advantage is most pronounced on the Worst metric, which measures the minimum score across all preferences and directly reflects multi-objective balance. For instance, on Llama-3.2-1B, MATO outperforms OPAD by $+0.37$ on Worst and $+0.24$ on APS, confirming that MATO effectively prevents any single preference from being neglected and thus achieves better overall multi-objective alignment. To ensure the robustness of our results, we also evaluate the multi-objective personalized alignment performance with an additional LLM judge (claude-sonnet-4-5) in Section D.1, which shows consistent relative performance among different methods and further confirms the effectiveness of MATO. Experiments on larger base LLMs (Qwen3-32B) are provided in Section D.2. Performance of original tasks are provided in Section D.3. Case studies are provided in Section D.10.

Performance by Dimension. To further demonstrate the effectiveness of MATO in balancing multiple preferences, we report the performance on each preference dimension. As shown in Figure 3, existing alignment methods tend to have imbalanced performance across different dimensions, while MATO achieves more balanced performance across all dimensions with APS consistently above 4.0. This further confirms the advantage of MATO in multi-objective alignment and preventing any single preference from being neglected.

Scaling to Different Numbers of Objectives. To evaluate the scalability of MATO to different numbers of objectives, we augment the original 4-objective setting by randomly sampling different numbers of preferences (from 2 to 10) for each query and evaluating the overall performance of different methods. As shown in Figure 4, while the performance of all methods generally decreases as the number of preferences increases, MATO consistently outperforms all baselines across different numbers of preferences, demonstrating its scalability in multi-objective alignment.

The latency analysis under different aligned preferences K is provided in Section D.9. To increase alignment efficiency, we further implement a *parallel decoding* where the objective-conditioned branches can be batched and executed in parallel. This approach reduces the measured latency to 0.12–0.15 seconds per token, while retaining the objective-wise signals used for dynamic balancing. These results highlight the benefit of parallel decoding when aligning a larger number of objectives.

5.3 Multi-objective Pareto Steerability

To evaluate steerability, we vary the helpfulness-humor weight from 1 to 0 with a step size of 0.2, compute the reward for each objective, and plot the empirical Pareto front [25]. For training-based methods, we train separate models following their original algorithms. For training-free baselines, we directly inject each weight combination into the preference prompt at inference time (Section E.2).

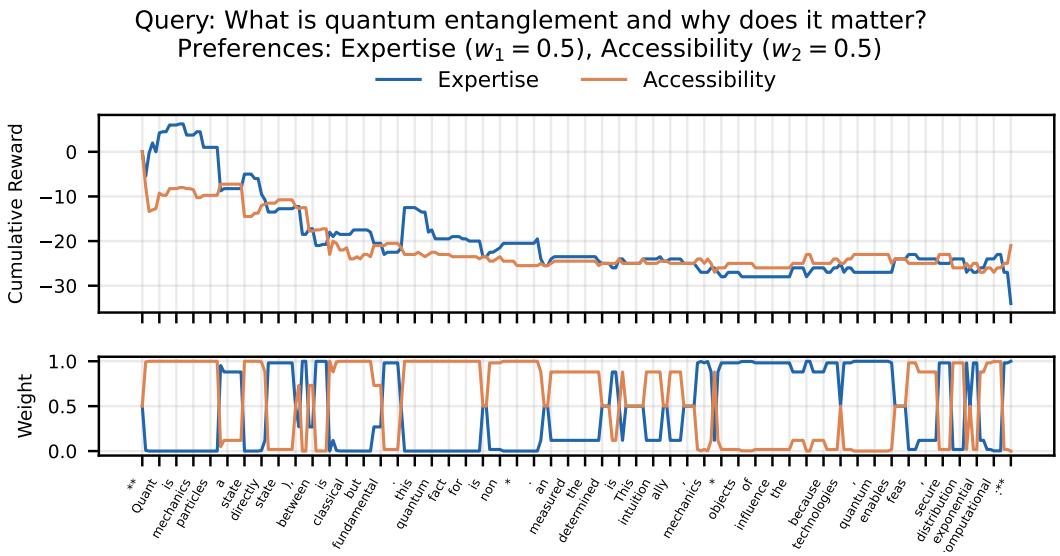


Figure 6: Illustration of the surrogate reward and weight dynamics of each preference during MATO decoding.

Figure 5 shows that training-based methods (MORLHF, RiC, and MOD) produce smooth and well-distributed Pareto fronts, indicating good steerability under different weights. However, their overall alignment performance is generally lower than that of training-free methods, likely due to train-test distribution shift. They are also more vulnerable to exploit reward model weaknesses without producing genuinely meaningful responses. Examples are provided in Section D.11.

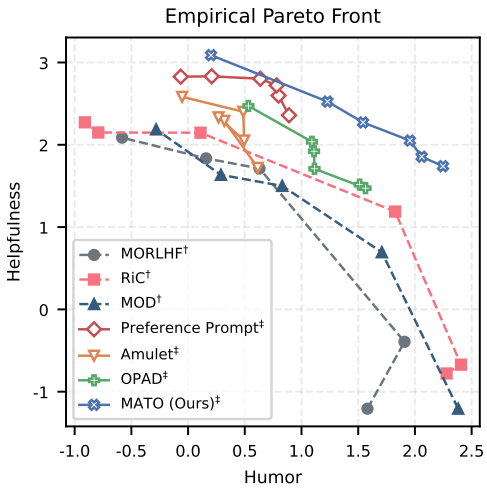


Figure 5: Empirical Pareto fronts of different methods: † and ‡ denote training-based and training-free methods, respectively.

Training-free methods such as Preference Prompt and OPAD achieve strong alignment but ill-distributed Pareto fronts, indicating limited steerability. A possible reason is that they collapse multiple preferences into a single prompt, making it difficult to capture the relative importance of each objective. In contrast, MATO explicitly models each preference and fuses logits according to target weights (Equation (10)), while the dynamic weight optimization (Equation (9)) adapts to the current generation state. This yields a substantially better Pareto front than all training-free baselines, achieving both strong alignment and effective steerability.

5.4 Reward and Weight Dynamics

To understand how MATO dynamically adjusts reward weights during decoding, we visualize the surrogate reward and weight dynamics of two competing preferences (Expertise vs. Accessibility). As shown in Figure 6, the surrogate rewards for different preferences fluctuate throughout decoding, and MATO adjusts the weights accordingly to compensate for underserved preferences. As a result, the surrogate rewards of the two preferences remain close and stay aligned with the initial user weights (0.5 vs. 0.5), demonstrating the effectiveness of MATO in dynamically balancing multiple preferences during generation. More case studies with different initial weights are provided in Section D.12.

5.5 Ablation Study and Parameter Analysis

The ablation study and parameter analysis are conducted with `Qwen3-0.6B` as the base LLM on a subset of the Multifaceted dataset for efficiency.

Ablation Study. We conduct ablation studies to evaluate the contribution of each component in MATO, including (1) *w/o weight optimization* that removes the weight optimization and directly uses the initial user weights for fusion, (2) *w/o online optimization* that removes the online optimization and directly fuses the logits with the optimized weights, and (3) *w/o both* that removes both modules, which is equivalent to directly fusing the logits from the base LLM with the initial user weights.

Table 2: Ablation study of MATO.

Methods	AMR	APS	Worst
MATO	**0.27**	**2.99**	**1.87**
MATO w/o weight optimization	0.26	2.88	1.80
MATO w/o online optimization	0.24	2.83	1.83
MATO w/o both	0.21	2.73	1.62

Table 2 shows that both weight optimization and online optimization contribute to the performance of MATO. Removing either degrades all metrics, while removing both causes the largest drop, especially on Worst, confirming that weight optimization rebalances underserved objectives, while online optimization further improves the generation distribution.

Analysis of the reliability of the prompt-based reward discovery is provided in Section D.4, which shows that the surrogate rewards are positively associated with the preference scores judged by LLMs, confirming that the reward discovery module can effectively construct surrogate rewards for multi-objective alignment. Robustness of prompt variants and effects of different reward accumulation are provided in Section D.5 and Section D.6, respectively. Results show that MATO is robust to different prompt variants while the reward accumulation method can affect the surrogate reward quality and thus requiring further investigation.

Parameter Analysis. We analyze the sensitivity of MATO to two key parameters: the weight optimization temperature τ and the number of online optimization steps T. As shown in Figure 7, the best performance is achieved when $\tau = 1$ and $T = 80$. A too-small τ leads to over-sharpened weights that may cause optimization instability, while a too-large τ results in over-smoothed weights that cannot effectively rebalance the preferences. For the number of online optimization steps, a moderate number of steps allows optimization toward better multi-objective alignment, while too many steps may lead to a distorted optimization direction and poor efficiency. Analysis of other parameters are provided in Sections D.7 and D.8.

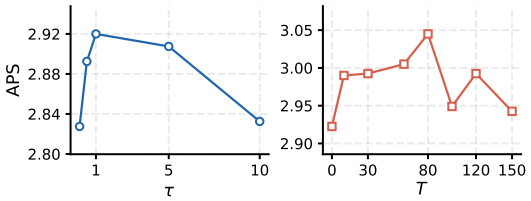


Figure 7: Analysis of weight optimization temperature τ and optimization steps T.

5.6 Efficiency Analysis

Table 3 compares efficiency across training-based (MORLHF, RiC, MOD) and training-free methods (Amulet, OPAD). Training-based methods require substantial optimization time, while MATO avoids training entirely. Compared with Amulet and OPAD, MATO introduces moderate per-token latency from test-time optimization, effectively shifting cost from training to inference while achieving better alignment. The latency analysis under different optimization steps T and aligned preferences K is provided in Section D.9.

Table 3: Efficiency comparison.

Methods	Training (hours)	Inference (seconds / token)
MORLHF	143.77	0.05
RiC	3.4	0.05
MOD	131.43	0.12
Amulet	0	0.15
OPAD	0	0.10
MATO	0	0.16

6 Conclusion

In this paper, we presented MATO, a training-free framework that formulates multi-objective personalized alignment as test-time optimization over the decoding distribution. MATO constructs per-objective prompt-derived surrogate rewards from the backbone LLM, dynamically rebalances objective weights around the user's preference in closed form, and refines the token distribution via online optimization, removing the need for per-preference reward models or model training while preserving user steerability. Experiments across four backbone LLMs show consistent gains over training-free and training-based baselines, with the largest improvements on the worst-served objective and substantially better Pareto fronts.

Acknowledgments

This work was supported by the Australian Defence Science and Technology (DST) Group through the Next Generation Technology Fund (NGTF) scheme and the Department of Defence, Australia, via the Advanced Strategic Capabilities Accelerator (ASCA) program. Dinh Phung further acknowledged the support from the Australian Research Council (ARC) Discovery Project DP230101176 and DP250100262.

Ethics Statement

Our research addresses only scientific questions and involves no human subjects, animals, or environmentally sensitive materials. Therefore, we anticipate no ethical risks or conflicts of interest. We are committed to upholding the highest standards of scientific integrity and ethics to ensure the validity and reliability of our findings.

Reproducibility Statement

Our model is clearly formalized in the main text for clarity and thorough understanding. Detailed implementation, including dataset information, baselines, experimental settings, and model configurations, are provided in Sections C.1, C.3 and 5.1. Experimental settings and baselines have been rigorously verified to ensure fair comparison. Code is available at: `https://github.com/RManLuo/MATO`.

Usage of LLMs

LLMs are used to proofread and polish the writing of this paper. We have carefully reviewed and verified all content generated by LLMs to ensure accuracy and integrity. Any errors or inaccuracies in the final manuscript are solely our responsibility.

References

[1] 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.

[2] Woohyeon Byeon, Giseung Park, Jongseong Chae, Amir Leshem, and Youngchul Sung. Multi-objective reinforcement learning with max-min criterion: A game-theoretic approach. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

[3] Jin Chen, Zheng Liu, Xu Huang, Chenwang Wu, Qi Liu, Gangwei Jiang, Yuanhao Pu, Yuxuan Lei, Xiaolong Chen, Xingmei Wang, et al. When large language models meet personalization: Perspectives of challenges and opportunities. *World wide web*, 27(4):42, 2024.

[4] Ruizhe Chen, Xiaotian Zhang, Meng Luo, Wenhao Chai, and Zuozhu Liu. Pad: Personalized alignment of llms at decoding-time. In *The Thirteenth International Conference on Learning Representations*, 2025.

[5] Xin Luna Dong, Seungwhan Moon, Yifan Ethan Xu, Kshitiz Malik, and Zhou Yu. Towards next-generation intelligent assistants leveraging llm techniques. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 5792–5793, 2023.

[6] Songyang Gao, Qiming Ge, Wei Shen, Shihan Dou, Junjie Ye, Xiao Wang, Rui Zheng, Yicheng Zou, Zhi Chen, Hang Yan, et al. Linear alignment: A closed-form solution for aligning human preferences without tuning and feedback. In *International Conference on Machine Learning*, pages 14702–14722. PMLR, 2024.

[7] Jian Guan, Junfei Wu, Jia-Nan Li, Chuanqi Cheng, and Wei Wu. A survey on personalized alignment—the missing piece for large language models in real-world applications. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 5313–5333, 2025.

[8] Jerry Zhi-Yang He, Sashrika Pandey, Mariah L Schrum, and Anca Dragan. Context steering: Controllable personalization at inference time. In *The Thirteenth International Conference on Learning Representations*, 2025.

[9] Xinmeng Huang, Shuo Li, Edgar Dobriban, Osbert Bastani, Hamed Hassani, and Dongsheng Ding. One-shot safety alignment for large language models via optimal dualization. *Advances in Neural Information Processing Systems*, 37:84350–84383, 2024.

[10] Jiaming Ji, Boyuan Chen, Hantao Lou, Donghai Hong, Borong Zhang, Xuehai Pan, Juntao Dai, Tianyi Qiu, and Yaodong Yang. Aligner: Efficient alignment by learning to correct. *Advances in Neural Information Processing Systems*, 37:90853–90890, 2024.

[11] Juyong Jiang, Fan Wang, Jiasi Shen, Sungju Kim, and Sunghun Kim. A survey on large language models for code generation. *ACM Transactions on Software Engineering and Methodology*, 35 (2):1–72, 2026.

[12] Seongyun Lee, Sue Hyun Park, Seungone Kim, and Minjoon Seo. Aligning to thousands of preferences via system message generalization. *Advances in Neural Information Processing Systems*, 37:73783–73829, 2024.

[13] Kaiwen Li, Tao Zhang, and Rui Wang. Deep reinforcement learning for multiobjective optimization. *IEEE transactions on cybernetics*, 51(6):3103–3114, 2020.

[14] Yuanchun Li, Hao Wen, Weijun Wang, Xiangyu Li, Yizhen Yuan, Guohong Liu, Jiacheng Liu, Wenxing Xu, Xiang Wang, Yi Sun, et al. Personal llm agents: Insights and survey about the capability, efficiency and security. *arXiv preprint arXiv:2401.05459*, 2024.

[15] Baijiong Lin, Weisen Jiang, Yuancheng Xu, Hao Chen, and Ying-Cong Chen. Parm: Multi-objective test-time alignment via preference-aware autoregressive reward model. In *International Conference on Machine Learning*, pages 37874–37888. PMLR, 2025.

[16] Bill Yuchen Lin, Abhilasha Ravichander, Ximing Lu, Nouha Dziri, Melanie Sclar, Khyathi Chandu, Chandra Bhagavatula, and Yejin Choi. The unlocking spell on base llms: Rethinking alignment via in-context learning. In *The Twelfth International Conference on Learning Representations*, 2024.

[17] Tianlin Liu, Shangmin Guo, Leonardo Bianco, Daniele Calandriello, Quentin Berthet, Felipe Llinares, Jessica Hoffmann, Lucas Dixon, Michal Valko, and Mathieu Blondel. Decoding-time realignment of language models. In *Proceedings of the 41st International Conference on Machine Learning*, pages 31015–31031, 2024.

[18] Brendan McMahan. Follow-the-regularized-leader and mirror descent: Equivalence theorems and l1 regularization. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, pages 525–533. JMLR Workshop and Conference Proceedings, 2011.

[19] Jesse G Meyer, Ryan J Urbanowicz, Patrick CN Martin, Karen O'Connor, Ruowang Li, Pei-Chen Peng, Tiffani J Bright, Nicholas Tatonetti, Kyoung Jae Won, Graciela Gonzalez-Hernandez, et al. Chatgpt and large language models in academia: opportunities and challenges. *BioData mining*, 16(1):20, 2023.

[20] 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.

[21] Giseung Park, Woohyeon Byeon, Seongmin Kim, Elad Havakuk, Amir Leshem, and Youngchul Sung. The max-min formulation of multi-objective reinforcement learning: from theory to a model-free algorithm. In *Proceedings of the 41st International Conference on Machine Learning*, pages 39616–39642, 2024.

[22] Sriyash Poddar, Yanming Wan, Hamish Ivison, Abhishek Gupta, and Natasha Jaques. Personalizing reinforcement learning from human feedback with variational preference learning. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

[23] 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*, 36:53728–53741, 2023.

[24] Rafael Rafailov, Joey Hejna, Ryan Park, and Chelsea Finn. From r to q^*: Your language model is secretly a q-function. In *First Conference on Language Modeling*, 2024. URL `https://openreview.net/forum?id=kEVcNxtqXk`.

[25] Alexandre Rame, Guillaume Couairon, Corentin Dancette, Jean-Baptiste Gaya, Mustafa Shukor, Laure Soulier, and Matthieu Cord. Rewarded soups: towards pareto-optimal alignment by interpolating weights fine-tuned on diverse rewards. *Advances in Neural Information Processing Systems*, 36:71095–71134, 2023.

[26] Ruizhe Shi, Yifang Chen, Yushi Hu, Alisa Liu, Hannaneh Hajishirzi, Noah A Smith, and Simon S Du. Decoding-time language model alignment with multiple objectives. *Advances in Neural Information Processing Systems*, 37:48875–48920, 2024.

[27] Seongho Son, William Bankes, Sangwoong Yoon, Shyam Sundhar Ramesh, Xiaohang Tang, and Ilija Bogunovic. Robust multi-objective controlled decoding of large language models. *ICLR*, 2026.

[28] Azmine Toushik Wasi, Mst Rafia Islam, and Raima Islam. Llms as writing assistants: Exploring perspectives on sense of ownership and reasoning. In *Proceedings of the Third Workshop on Intelligent and Interactive Writing Assistants*, pages 38–42, 2024.

[29] Yuancheng Xu, Udari Madhushani Sehwag, Alec Koppel, Sicheng Zhu, Bang An, Furong Huang, and Sumitra Ganesh. Genarm: Reward guided generation with autoregressive reward model for test-time alignment. In *The Thirteenth International Conference on Learning Representations*, 2025.

[30] Kailai Yang, Zhiwei Liu, Qianqian Xie, Jimin Huang, Tianlin Zhang, and Sophia Ananiadou. Metaaligner: Towards generalizable multi-objective alignment of language models. *Advances in Neural Information Processing Systems*, 37:34453–34486, 2024.

[31] Rui Yang, Xiaoman Pan, Feng Luo, Shuang Qiu, Han Zhong, Dong Yu, and Jianshu Chen. Rewards-in-context: Multi-objective alignment of foundation models with dynamic preference adjustment. In *International Conference on Machine Learning*, pages 56276–56297. PMLR, 2024.

[32] Zhaowei Zhang, Fengshuo Bai, Qizhi Chen, Chengdong Ma, Mingzhi Wang, Haoran Sun, Zilong Zheng, and Yaodong Yang. Amulet: Realignment during test time for personalized preference adaptation of llms. In *The Thirteenth International Conference on Learning Representations*, 2025.

[33] Zhehao Zhang, Ryan A Rossi, Branislav Kveton, Yijia Shao, Diyi Yang, Hamed Zamani, Franck Dernoncourt, Joe Barrow, Tong Yu, Sungchul Kim, et al. Personalization of large language models: A survey. *Transactions on Machine Learning Research*, 2025.

[34] Zhenyuan Zhang, Xianzhang Jia, Zhiqin Yang, Zhenbo Song, Wei Xue, Sirui Han, and Yike Guo. Memfly: On-the-fly memory optimization via information bottleneck. *arXiv preprint arXiv:2602.07885*, 2026.

[35] Siyan Zhao, Mingyi Hong, Yang Liu, Devamanyu Hazarika, and Kaixiang Lin. Do llms recognize your preferences? evaluating personalized preference following in llms. In *The Thirteenth International Conference on Learning Representations*, 2025.

[36] Yifan Zhong, Chengdong Ma, Xiaoyuan Zhang, Ziran Yang, Haojun Chen, Qingfu Zhang, Siyuan Qi, and Yaodong Yang. Panacea: Pareto alignment via preference adaptation for llms. *Advances in Neural Information Processing Systems*, 37:75522–75558, 2024.

[37] Zhanhui Zhou, Jie Liu, Jing Shao, Xiangyu Yue, Chao Yang, Wanli Ouyang, and Yu Qiao. Beyond one-preference-fits-all alignment: Multi-objective direct preference optimization. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 10586–10613, 2024.

[38] Mingye Zhu, Yi Liu, Lei Zhang, Junbo Guo, and Zhendong Mao. On-the-fly preference alignment via principle-guided decoding. In *The Thirteenth International Conference on Learning Representations*, 2025.

Appendix

Table of Contents

A Detailed Related Work Comparison **15**
A.1 Training-based Personalized Alignment 15
A.2 Training-free personalized alignment. 15

B Detailed Theoretical Proofs **15**
B.1 Ideal Reward Representation and Prompt-derived Surrogate 15
B.2 Derivation of Closed-form Weight Optimization 17
B.3 Interpretation of Online Optimization 19
B.4 Derivation of Closed-form Solution for Online Optimization 20

C Detailed Experiment Setups **21**
C.1 Datasets 21
C.2 Evaluation Metrics 21
C.3 Implementation Details 23

D Additional Experiments **23**
D.1 Additional LLM-as-judge Evaluation 23
D.2 Results on Larger Models 23
D.3 Preservation of Original Tasks 24
D.4 Reliability of Prompt-based Reward Discovery 24
D.5 Robustness to Prompt Variants 25
D.6 Analysis of Reward Accumulation 25
D.7 Effect of Generation Temperature 26
D.8 Additional Parameter Analysis 26
D.9 Inference Efficiency and Cost–Benefit 26
D.10 Case Studies for Multi-objective Alignment 27
D.11 Case Studies for Steerability 28
D.12 Case Studies for Reward and Weight Dynamics 29

E Prompt **30**
E.1 Prompt for Reward Discovery 30
E.2 Prompt for Prompt-based Multi-objective Alignment 31
E.3 Prompt for LLM-as-judge Evaluation 31

F Limitation **32**

A Detailed Related Work Comparison

A.1 Training-based Personalized Alignment

Early research adopts the multi-objective reinforcement learning (MORL) framework [13], which extends RLHF to optimize LLMs on linearly combined multi-preference rewards. MODPO [37] extends DPO [23] to multi-objective alignment, outperforming MORLHF with greater stability and lower compute cost. However, these methods require training separate models for different preference combinations. Rewarded Soups [25] alleviates this issue by interpolating the LLM weights to approximate Pareto fronts. Panacea [36] introduces a preference-controlled LoRA adapter that injects a preference vector into the LoRA weights and achieves efficient multi-objective alignment with a single model. Instead of RL training, RiC [31] further improves the efficiency by training a single model with preference weights as input using supervised fine-tuning (SFT). Aligner [10] is fine-tuned on a preference dataset to learn the correctional residuals between preferred and non-preferred responses and then stacked on the upstream model to achieve corrected alignment. MetaAligner [30] further trains a generalized rewriter that rewrites model outputs based on any user preferences.

Recently, test-time alignment methods [4, 34, 29] keep the backbone LLM fixed and guide the decoding process with reward models for each preference. MOD [17] trains multiple LLMs for different objectives and merges their logits during decoding. PARM [15] introduces a preference-aware reward model that can unify multiple preferences and trade-offs. PAD [4] trains a preference-conditioned reward model that can adapt to evolving user preferences and guide LLM decoding. RMOD [27] further optimizes the decoding process by improving the worst-case reward. Nevertheless, these methods still require pre-trained reward models for each preference, which can be hard to obtain as user preferences evolve and new objectives emerge [7].

A.2 Training-free personalized alignment.

The training-free paradigm has gained increasing attention for personalized alignment due to its flexibility and efficiency. Prompt-based personalization [12] leverages the instruction-following capabilities of LLMs to adapt outputs to user preferences expressed in natural language. To enhance the alignment quality, Linear Alignment [6] derives a preference alignment direction by comparing model outputs with and without the preference prompt, and steers the decoding process accordingly. Similarly, CoS [8] amplifies the influence of the preference prompt by comparing two model outputs and controls the level of personalization by adjusting the interpolation weight. OPAD [38] proposes a preference-guided decoding alignment method guided by a sequential divergence reward design. Amulet [32] adopts an online learning framework to optimize the output token distribution and achieve better alignment with user preferences. However, these methods are designed for single-objective alignment and cannot handle the trade-offs between multiple preferences. In contrast, our MATO introduces a multi-objective optimization framework for training-free personalized alignment, which can effectively balance multiple preferences and adapt to evolving user needs. Detailed comparisons with these methods are provided in Table 4.

B Detailed Theoretical Proofs

B.1 Ideal Reward Representation and Prompt-derived Surrogate

Proposition B.1. *For the KL-regularized multi-objective RL objective in Equation* (2)*, let* π_k^* *denote the optimal policy obtained by optimizing only the* k*-th objective, i.e., with weight vector equal to the* k*-th simplex vertex. Assuming the policies share support and the partition function is finite, a token-level representation of the sequence reward is defined by*

$$R_k(y_i|x, y_{<i}) := \beta \log \frac{\pi_k^*(y_i \mid x, y_{<i})}{\pi_{base}(y_i \mid x, y_{<i})}, \tag{15}$$

The aggregation of these terms equals $R_k(x, y) - \beta \log Z_k(x)$*, establishing the representation in Equation* (4)*.*

Proof. The proof follows the closed-form solution of KL-regularized RL used in DPO [23, 24], adapted to the multi-objective setting considered in this paper.

Table 4: Comparisons between previous alignment methods and MATO.

Category	Method	Training-free	Reward-free	Pareto Steerable	Generalizability
RL Training	MORLHF [13]	✗	✗	✓	✗
	MODPO [37]	✗	✗	✓	✗
	Rewarded Soup [25]	✗	✗	✓	✗
	Panacea [36]	✗	✗	✓	✗
SFT	RiC [31]	✗	✗	✓	✗
	Aligner [10]	✗	✓	✗	✗
	MetaAligner [30]	✗	✓	✗	✓
Test-time Alignment	MOD [17]	✗	✗	✓	✗
	PARM [15]	✗	✗	✓	✗
	PAD [4]	✗	✗	✗	✓
	RMOD [27]	✗	✗	✗	✗
	LA [6]	✓	✓	✗	✓
	CoS [8]	✓	✓	✗	✓
	Amulet [32]	✓	✓	✗	✓
	OPAD [38]	✓	✓	✗	✓
	MATO (Ours)	✓	✓	✓	✓

For a fixed input x, the multi-objective RL objective in Equation (2) is

$$\pi^*(\cdot \mid x) = \arg\max_{\pi} \mathbb{E}_{y\sim\pi(\cdot|x)} \left[\sum_{k=1}^{K} w_k R_k(x,y)\right] - \beta D_{\mathrm{KL}}(\pi(\cdot \mid x)\|\pi_{\text{base}}(\cdot \mid x)). \tag{16}$$

By the standard derivation of KL-regularized RL, the optimizer has the closed form

$$\pi^*(y \mid x) = \frac{1}{Z(x)}\pi_{\text{base}}(y \mid x)\exp\left(\frac{1}{\beta}\sum_{k=1}^{K} w_k R_k(x,y)\right), \tag{17}$$

where

$$Z(x) = \sum_{y} \pi_{\text{base}}(y \mid x)\exp\left(\frac{1}{\beta}\sum_{k=1}^{K} w_k R_k(x,y)\right) \tag{18}$$

is the normalizing partition function. Taking logarithms on both sides of Equation (17) gives

$$\sum_{k=1}^{K} w_k R_k(x,y) = \beta \log \frac{\pi^*(y \mid x)}{\pi_{\text{base}}(y \mid x)} + \beta \log Z(x). \tag{19}$$

Thus, in the multi-objective setting, the weighted aggregate reward is determined by the log-ratio between the optimal policy and the base policy, up to an additive term depending only on x.

However, MATO requires each objective-specific reward separately. To isolate the k-th reward, we consider the special case where the weight vector is the k-th simplex vertex, namely $w_k = 1$ and $w_j = 0$ for all $j \neq k$. Let π_k^* denote the optimal policy of this single-objective subproblem:

$$\pi_k^*(\cdot \mid x) = \arg\max_{\pi} \mathbb{E}_{y\sim\pi(\cdot|x)}[R_k(x,y)] - \beta D_{\mathrm{KL}}(\pi(\cdot \mid x)\|\pi_{\text{base}}(\cdot \mid x)). \tag{20}$$

Applying the same closed-form argument yields

$$\pi_k^*(y \mid x) = \frac{1}{Z_k(x)}\pi_{\text{base}}(y \mid x)\exp\left(\frac{1}{\beta}R_k(x,y)\right), \tag{21}$$

where $Z_k(x)$ is the corresponding normalization term. Rearranging Equation (21), we obtain

$$R_k(x,y) = \beta \log \frac{\pi_k^*(y \mid x)}{\pi_{\text{base}}(y \mid x)} + \beta \log Z_k(x). \tag{22}$$

Next, using the autoregressive factorization of language models [24],

$$\pi_k^*(y \mid x) = \prod_{i=1}^{|y|} \pi_k^*(y_i \mid x, y_{<i}), \quad \pi_{\text{base}}(y \mid x) = \prod_{i=1}^{|y|} \pi_{\text{base}}(y_i \mid x, y_{<i}), \tag{23}$$

the log-ratio in Equation (22) decomposes as

$$\log \frac{\pi_k^*(y \mid x)}{\pi_{\text{base}}(y \mid x)} = \sum_{i=1}^{|y|} \log \frac{\pi_k^*(y_i \mid x, y_{<i})}{\pi_{\text{base}}(y_i \mid x, y_{<i})}. \tag{24}$$

Substituting this decomposition into Equation (22) shows that, up to an input-dependent additive constant, the sequence-level reward can be written as a sum of token-level rewards:

$$R_k(x, y) = \sum_{i=1}^{|y|} R_k(y_i \mid x, y_{<i}) + \beta \log Z_k(x), \tag{25}$$

where we define

$$R_k(y_i|x, y_{<i}) := \beta \log \frac{\pi_k^*(y_i \mid x, y_{<i})}{\pi_{\text{base}}(y_i \mid x, y_{<i})}. \tag{26}$$

This establishes the chosen representation in Equation (4). The constant does not affect the optimal policy for fixed objective weights, but objective-specific offsets can affect adaptive comparisons across objectives. □

Prompt-derived surrogate. The ideal policy π_k^* is unavailable in MATO. Equation 5 instead defines $\widehat{R}_k$ using the prompt-conditioned backbone. This is an exact definition of the surrogate, not an exact recovery of R_k. Its usefulness depends on the backbone's response to c_k; the derivation does not establish accuracy relative to ground-truth preferences or calibration across objectives. The weight updates below operate on these surrogate rewards with the log-ratio baseline fixed by Equation (5).

B.2 Derivation of Closed-form Weight Optimization

Proposition B.2. *Fix a decoding prefix $y_{<i}$ and a regularization strength $\tau > 0$, and let the user prior be $\mathcal{W}^{\text{init}} = \{w_1^{\text{init}}, \ldots, w_K^{\text{init}}\} \in \Delta^K$ with $w_k^{\text{init}} > 0$ for all k and $\sum_{k=1}^K w_k^{\text{init}} = 1$. Consider the regularized weight optimization problem in Equation* (8)*:*

$$\mathcal{W}^*(x, y_{<i}) = \arg \min_{\mathcal{W} \in \Delta^K} \sum_{k=1}^K w_k \widehat{R}_k(x, y_{<i}) + \tau D_{\text{KL}}(\mathcal{W}||\mathcal{W}^{\text{init}}). \tag{27}$$

Then the problem admits a unique optimizer, and its k-th component is

$$w_k^* = \frac{w_k^{\text{init}} \cdot \exp\left(-\frac{\widehat{R}_k(x, y_{<i})}{\tau}\right)}{\sum_{j=1}^K w_j^{\text{init}} \cdot \exp\left(-\frac{\widehat{R}_j(x, y_{<i})}{\tau}\right)}, \tag{28}$$

which is exactly Equation (9)*.*

Proof. We first place the proposed weight optimization into the broader max-min optimization literature [21]. In max-min MORL, the problem can be viewed as a min-max game in which an adversarial weight player selects a point on the simplex to emphasize the worst-performing objective [2], namely

$$\max_{\pi} \min_{\mathcal{W} \in \Delta^K} \sum_{k=1}^K w_k R_k(x, y_{<i}) = \min_{\mathcal{W} \in \Delta^K} \max_{\pi} \sum_{k=1}^K w_k R_k(x, y_{<i}). \tag{29}$$

The inner maximization over policies corresponds to the KL-regularized multi-objective RL objective in Equation (2). As shown in Equation (17), this inner problem admits a closed-form optimal policy for any fixed weight vector $\mathcal{W}$ [27]. Motivated by this adversarial reweighting perspective, MATO optimizes the weights at the current decoding prefix using the cumulative surrogate rewards, while introducing a KL tether to the user prior $\mathcal{W}^{\text{init}}$ so that the reweighting remains personalized and admits a closed-form solution. This yields the following regularized weight optimization problem:

$$\mathcal{W}^*(x, y_{<i}) = \arg \min_{\mathcal{W} \in \Delta^K} \sum_{k=1}^K w_k \widehat{R}_k(x, y_{<i}) + \tau D_{\text{KL}}(\mathcal{W}||\mathcal{W}^{\text{init}}), \tag{30}$$

$$s.t. \sum_{k=1}^K w_k = 1, \quad w_k \geq 0. \tag{31}$$

For a fixed prefix $y_{<i}$, define the objective

$$F(\mathcal{W}) = \sum_{k=1}^{K} w_k \widehat{R}_k(x, y_{<i}) + \tau D_{\mathrm{KL}}(\mathcal{W}||\mathcal{W}^{\mathrm{init}}) = \sum_{k=1}^{K} w_k \widehat{R}_k(x, y_{<i}) + \tau \sum_{k=1}^{K} w_k \log \frac{w_k}{w_k^{\mathrm{init}}}, \quad (32)$$

over the feasible set Δ^K. The first term is linear in $\mathcal{W}$, whereas the KL term is strictly convex in $\mathcal{W}$ whenever $\tau > 0$ and $w_k^{\mathrm{init}} > 0$ for all k. It follows that F is strictly convex on the simplex, and hence the optimization problem admits a unique minimizer.

To characterize this minimizer, consider the Lagrangian

$$\mathcal{L}(\mathcal{W}, \lambda, \nu) = \sum_{k=1}^{K} w_k \widehat{R}_k(x, y_{<i}) + \tau \sum_{k=1}^{K} w_k \log \frac{w_k}{w_k^{\mathrm{init}}} + \lambda \left(\sum_{k=1}^{K} w_k - 1 \right) - \sum_{k=1}^{K} \nu_k w_k, \quad (33)$$

where $\lambda \in \mathbb{R}$ is the multiplier for the simplex equality constraint and $\nu_k \geq 0$ are the multipliers for the non-negativity constraints.

Since F is convex, the constraints are affine, and F is differentiable at the interior point $\mathcal{W}^*$, the Karush-Kuhn-Tucker (KKT) conditions are necessary and sufficient for optimality at $\mathcal{W}^*$. In particular, the stationarity condition for each coordinate is

$$\frac{\partial \mathcal{L}}{\partial w_k} = \widehat{R}_k(x, y_{<i}) + \tau \left(\log \frac{w_k}{w_k^{\mathrm{init}}} + 1 \right) + \lambda - \nu_k = 0. \quad (34)$$

Because $w_k^* > 0$ for every k, complementary slackness gives $\nu_k = 0$ for all k.

Substituting $\nu_k = 0$ into Equation (34) yields

$$\widehat{R}_k(x, y_{<i}) + \tau \left(\log \frac{w_k^*}{w_k^{\mathrm{init}}} + 1 \right) + \lambda = 0, \quad (35)$$

or equivalently,

$$\log \frac{w_k^*}{w_k^{\mathrm{init}}} = -\frac{\widehat{R}_k(x, y_{<i}) + \lambda + \tau}{\tau}. \quad (36)$$

Exponentiating both sides gives

$$w_k^* = w_k^{\mathrm{init}} \exp\left(-\frac{\widehat{R}_k(x, y_{<i}) + \lambda + \tau}{\tau} \right) = C \cdot w_k^{\mathrm{init}} \exp\left(-\frac{\widehat{R}_k(x, y_{<i})}{\tau} \right), \quad (37)$$

where $C = \exp\left(-\frac{\lambda+\tau}{\tau}\right)$ is a constant independent of k. Imposing the simplex constraint $\sum_{k=1}^{K} w_k^* = 1$ then gives

$$C^{-1} = \sum_{j=1}^{K} w_j^{\mathrm{init}} \exp\left(-\frac{\widehat{R}_j(x, y_{<i})}{\tau} \right). \quad (38)$$

Substituting this normalization constant back yields

$$w_k^* = \frac{w_k^{\mathrm{init}} \cdot \exp\left(-\frac{\widehat{R}_k(x, y_{<i})}{\tau} \right)}{\sum_{j=1}^{K} w_j^{\mathrm{init}} \cdot \exp\left(-\frac{\widehat{R}_j(x, y_{<i})}{\tau} \right)}, \quad (39)$$

which proves Equation (9). □

Remark. The solution in Equation (9) has several useful interpretations. First, objectives with smaller cumulative surrogate reward receive larger optimized weights, so under-optimized preferences are automatically upweighted. Second, as $\tau \to \infty$, we have $\mathcal{W}^* \to \mathcal{W}^{\mathrm{init}}$, meaning the optimization preserves the user's original preference profile. Third, as $\tau \to 0$, the weight mass concentrates on the objectives attaining the minimum cumulative surrogate reward. Finally, if some $w_k^{\mathrm{init}} = 0$, then any finite solution must also satisfy $w_k^* = 0$; in that case, the same derivation applies on the support of $\mathcal{W}^{\mathrm{init}}$.

B.3 Interpretation of Online Optimization

Online optimization amplifies model-derived preference signals under KL regularization without using external preference feedback. We now explain why maximizing the divergence from the base policy π_{base} while staying close to the merged policy π' can be viewed as a surrogate for approaching optimal aligned policy π^* following [38].

The key observation is that π' is already an *aligned* distribution. Substituting Equation (5) into Equation (10) and using $\sum_{k=1}^{K} w_k^* = 1$, we obtain

$$\pi'(y_i \mid x, y_{<i}, \mathcal{C}) \propto \pi_{\text{base}}(y_i \mid x, y_{<i}, \mathcal{C})\, \pi_{\text{base}}(y_i \mid x, y_{<i}) \exp\left(\frac{1}{\beta} \sum_{k=1}^{K} w_k^* \widehat{R}_k(y_i \mid x, y_{<i})\right), \quad (40)$$

so the density ratio between π' and π_{base} is

$$\frac{\pi'(y_i \mid x, y_{<i}, \mathcal{C})}{\pi_{\text{base}}(y_i \mid x, y_{<i})} \propto \pi_{\text{base}}(y_i \mid x, y_{<i}, \mathcal{C}) \exp\left(\frac{1}{\beta} \sum_{k=1}^{K} w_k^* \widehat{R}_k(y_i \mid x, y_{<i})\right). \quad (41)$$

Therefore, relative to π_{base}, π' reweights each token by both its anchor probability under $\mathcal{C}$ and its exponentiated weighted surrogate reward: tokens favored by these signals are upweighted, while low-quality or poorly aligned tokens are downweighted. In this sense, π' already filters the broad support of π_{base} toward regions preferred by the aligned constraints $\mathcal{C}$.

Under this interpretation, the role of divergence maximization is analogous to the surrogate argument used in preference-aligned divergence rewards. Suppose that, for the current decoding prefix, the following conditions approximately hold:

1. the base policy π_{base} is a relatively poor approximation to the optimal aligned policy π^*;
2. the merged policy π' is a better approximation to π^* than π_{base}, because it already incorporates the surrogate rewards and optimized weights;
3. $\text{supp}(\pi^*) \subseteq \text{supp}(\pi') \subseteq \text{supp}(\pi_{\text{base}})$, so the relevant KL divergences are well-defined.

Then maximizing the divergence from π_{base} encourages the updated policy to move away from regions where the base model places probability mass but which are not supported by the aligned rewards. At the same time, the regularization term $-\lambda D_{\text{KL}}(\pi \| \pi')$ prevents this movement from drifting arbitrarily far, and instead constrains the update to remain in the neighborhood of π', which is assumed to be closer to π^*.

This can also be seen directly from the utility in Equations (12) and (13). For a candidate policy π at step j, the expected token-wise utility term approximately satisfies

$$\alpha D_{\text{KL}}(\pi \| \pi_{\text{base}}) \approx \sum_{a \in \mathcal{V}} \pi(a) u_j(a), \quad (42)$$

where a denotes the action of selecting a token from the model's output distribution. The approximation follows from the first-order Taylor expansion of $\alpha D_{\text{KL}}(\pi \| \pi_{\text{base}})$ around the current policy π_j, up to constants that do not affect the update over the simplex. The regularized local surrogate objective can therefore be interpreted, at this first-order level, as

$$\alpha D_{\text{KL}}(\pi \| \pi_{\text{base}}) - \lambda D_{\text{KL}}(\pi \| \pi'). \quad (43)$$

The first term locally rewards directions that increase separation from the base model, while the second term keeps the updated policy close to the aligned anchor π'. Consequently, the optimization does not seek arbitrary divergence from π_{base}; rather, it favors divergence *in the direction of* π'. Since π' assigns relatively more probability mass to high-reward aligned regions, this trust-region style objective pushes the policy away from poor behaviors of π_{base} and toward the high-quality regions encoded by $\mathcal{C}$.

Therefore, when π' is already a better approximation to π^* than π_{base}, maximizing divergence from π_{base} under proximity to π' serves as a meaningful surrogate for indirectly approaching the optimal aligned policy π^*.

B.4 Derivation of Closed-form Solution for Online Optimization

Proposition B.3. *Fix a decoding prefix $(x, y_{<i})$ and let $\mathcal{V}$ be the vocabulary. Consider the FTRL update in Equation* (11)*, where*

$$u_j(a) = \alpha \left(\log \pi_j(a|x, y_{<i}) - \log \pi_{\textit{base}}(a|x, y_{<i}) \right), \quad a \in \mathcal{V}, \tag{44}$$

and

$$\mathcal{U}_j(\pi) = \sum_{a \in \mathcal{V}} \pi(a) u_j(a) - \lambda D_{\mathrm{KL}}(\pi || \pi'). \tag{45}$$

Then, for every $t \geq 1$, the optimizer of

$$\pi_t = \arg\max_{\pi} \sum_{j=1}^{t-1} \mathcal{U}_j(\pi) - \frac{1}{\eta} D_{\mathrm{KL}}(\pi || \pi_{t-1}), \quad \pi_0 = \pi' \tag{46}$$

is unique and admits the closed form

$$\pi_t(a) \propto \exp\left(\frac{1}{(t-1)\lambda\eta + 1} \left(\eta \sum_{j=1}^{t-1} u_j(a) + (t-1)\lambda\eta \log \pi'(a) + \log \pi_{t-1}(a) \right) \right), \quad a \in \mathcal{V}. \tag{47}$$

Proof. The closed-form solution for the FTRL update has been given in [32]; we restate it here for completeness. For brevity, we suppress the prefix $(x, y_{<i})$ and write $\pi(a)$, $\pi'(a)$, and $\pi_{t-1}(a)$ for the corresponding next-token probabilities. Expanding Equation (11) with Equation (12), the optimization problem for step t is

$$\max_{\pi \in \Delta^{|\mathcal{V}|}} \sum_{j=1}^{t-1} \sum_{a \in \mathcal{V}} \pi(a) u_j(a) - \sum_{j=1}^{t-1} \lambda D_{\mathrm{KL}}(\pi || \pi') - \frac{1}{\eta} D_{\mathrm{KL}}(\pi || \pi_{t-1}). \tag{48}$$

Using the definition of KL divergence,

$$D_{\mathrm{KL}}(\pi || q) = \sum_{a \in \mathcal{V}} \pi(a) \log \frac{\pi(a)}{q(a)}, \tag{49}$$

the objective in Equation (48) becomes

$$\begin{aligned} J_t(\pi) = &\sum_{j=1}^{t-1} \sum_{a \in \mathcal{V}} \pi(a) u_j(a) - \sum_{j=1}^{t-1} \lambda \sum_{a \in \mathcal{V}} \pi(a) \log \frac{\pi(a)}{\pi'(a)} \\ &- \frac{1}{\eta} \sum_{a \in \mathcal{V}} \pi(a) \log \frac{\pi(a)}{\pi_{t-1}(a)}. \end{aligned} \tag{50}$$

Because the feasible set is the simplex, we introduce a Lagrange multiplier μ_t for the constraint $\sum_{a \in \mathcal{V}} \pi(a) = 1$ and form

$$\begin{aligned} \mathcal{L}_t(\pi, \mu_t) = &\sum_{j=1}^{t-1} \sum_{a \in \mathcal{V}} \pi(a) u_j(a) - \sum_{j=1}^{t-1} \lambda \sum_{a \in \mathcal{V}} \pi(a) \log \frac{\pi(a)}{\pi'(a)} \\ &- \frac{1}{\eta} \sum_{a \in \mathcal{V}} \pi(a) \log \frac{\pi(a)}{\pi_{t-1}(a)} + \mu_t \left(1 - \sum_{a \in \mathcal{V}} \pi(a) \right). \end{aligned} \tag{51}$$

Taking the derivative with respect to $\pi(a)$ gives

$$\frac{\partial \mathcal{L}_t}{\partial \pi(a)} = \sum_{j=1}^{t-1} u_j(a) - (t-1)\lambda \left(\log \frac{\pi(a)}{\pi'(a)} + 1 \right) - \frac{1}{\eta} \left(\log \frac{\pi(a)}{\pi_{t-1}(a)} + 1 \right) - \mu_t. \tag{52}$$

At the optimum π_t, the stationarity condition $\frac{\partial \mathcal{L}_t}{\partial \pi(a)} = 0$ yields

$$0 = \sum_{j=1}^{t-1} u_j(a) - \left((t-1)\lambda + \frac{1}{\eta}\right) \log \pi_t(a) + (t-1)\lambda \log \pi'(a) + \frac{1}{\eta} \log \pi_{t-1}(a) - (t-1)\lambda - \frac{1}{\eta} - \mu_t. \tag{53}$$

Rearranging terms gives

$$\left((t-1)\lambda + \frac{1}{\eta}\right) \log \pi_t(a) = \sum_{j=1}^{t-1} u_j(a) + (t-1)\lambda \log \pi'(a) + \frac{1}{\eta} \log \pi_{t-1}(a) + C_t, \tag{54}$$

where

$$C_t = -(t-1)\lambda - \frac{1}{\eta} - \mu_t \tag{55}$$

is independent of a. Multiplying both sides by η and absorbing the constant ηC_t into a new normalization constant, we obtain

$$\log \pi_t(a) = \frac{1}{(t-1)\lambda\eta + 1} \left(\eta \sum_{j=1}^{t-1} u_j(a) + (t-1)\lambda\eta \log \pi'(a) + \log \pi_{t-1}(a) \right) + \widetilde{C}_t, \tag{56}$$

where $\widetilde{C}_t$ does not depend on a. Exponentiating both sides yields

$$\pi_t(a) \propto \exp\left(\frac{1}{(t-1)\lambda\eta + 1} \left(\eta \sum_{j=1}^{t-1} u_j(a) + (t-1)\lambda\eta \log \pi'(a) + \log \pi_{t-1}(a) \right) \right). \tag{57}$$

Finally, the proportionality constant is determined by $\sum_{a \in \mathcal{V}} \pi_t(a) = 1$, which completes the proof. □

C Detailed Experiment Setups

C.1 Datasets

Multi-objective personalized alignment. We use the Multifaceted dataset [12], which contains 945 queries each associated with 4 distinct preferences specified in natural language. The preferences are drawn from 4 dimensions: Background Knowledge, Harmlessness, Informativeness, and Style with 107 unique preference combinations in total. The full list of preferences is shown in Table 5.

Multi-objective Pareto Steerability. We use the HH-RLHF dataset [1], which is widely used for training and evaluating multi-objective alignment algorithms. We define the two objectives as helpfulness and humor, and vary the target weights to evaluate the steerability of different methods across the two-objective plane. The descriptions of the two objectives are given in Table 6.

C.2 Evaluation Metrics

For the multi-objective personalized alignment experiments in Section 5.1, each query x is associated with $K = 4$ target preferences. We use GPT-4o and claude-sonnet-4-5 as an LLM-as-judge and evaluate the generated response independently on each preference using the rubric from the Multi-faceted dataset, yielding an integer score $s_k \in \{1, 2, 3, 4, 5\}$ for query x and preference k. The judge prompt is given in Section E.3. From these per-preference scores, we compute the reported metrics as follows.

All Preference Match Rate (AMR). This metric checks whether a response satisfies *all* assigned preferences at an acceptable level. For each response, we first define a binary success indicator

$$\mathrm{AMR} = \prod_{k=1}^{K} \mathbb{I}(s_k \geq 3), \tag{58}$$

where $\mathbb{I}(\cdot)$ is the indicator function. Since the product equals 1 only when every preference score is at least 3, AMR marks whether the response achieves simultaneous multi-objective satisfaction.

Table 5: Full list of preferences in the Multifaceted dataset

Dimension	Subdimension	Preference keywords
Style	Formality	Formal, Informal
	Clarity	Simple language, Complex language
	Conciseness	Concise, Verbose/lengthy, Clear, Non-repetitive
	Vividness	Use rhetorical devices (metaphors, personification, similes, hyperboles, irony, parallelism)
	Format	Breadth-first, Depth-first, Step-by-step, Consistency, Deductive, Inductive, Parallelism, Bullet points, Narrative, Satisfy constraints, Interactive, Support stances
	Tone	Agreeable, Sympathetic, Cooperative, Modest, Altruistic, Appreciative, Forgiving, Generous, Kind, Friendly, Polite, Funny, Gregariousness, Assertiveness, Friendliness, Extraversion, Excitement-seeking, Activity-level, Cheerfulness, Energetic, Enthusiastic, Talkative, Anxiety, Introspection/private self-consciousness, Neuroticism, Aloof, Anger, Depression, Immoderation, Vulnerability, Self-pitying, Self-beneficial, Tense, Touchy, Unstable, Worrying, Emotionality, Intellect, Adventurousness, Intellectual openness, Liberalism, Openness to experience, Curious, Authoritative, Persuasive
Background knowledge	Basic	Minimum awareness of the topic
	Novice	Limited experience and understanding
	Intermediate	Capable of practical application
	Advanced	Utilizing applied theory
	Expert	$5 \geq$ years of field experience and authoritative knowledge of the discipline
Informativeness	Depth	General topic, Specific topic, Nuanced insights
	Creativity	Artistic, Insightful, Original, Imaginative, Novel, Explorative creativity
	Efficiency	Efficient, Achievement-striving, Self-discipline, Self-contained, Contain rich info
	Practicality	Practical, Use supporting materials, Tailored examples and anecdotes, Empowering actionable insights
Harmlessness	Accuracy	Grammatically correct (grammar, spelling, punctuation, and code-switching), No minor errors, No moderate errors, No severe errors, Correct mistakes, Clarify intent
	Morality	Moral and ethical, Culturally inclusive
	Trustworthiness	Trustworthy, Dutifulness, Conscientiousness, Cautiousness, Self-efficacy, Reliable, Responsible, Metacognition, Admit limitations or mistakes, Express uncertainty

Table 6: Preference descriptions used in the HH-RLHF dataset.

Preference	Description
Helpfulness	The response should provide useful resources and suggestions to the user.
Humor	The response should be cheerful and amusing.

Table 7: Additional LLM-as-judge evaluation on multi-objective personalized alignment using claude-sonnet-4-5.

Methods	Qwen3-0.6B			Qwen3-8B			Llama-3.2-1B			Llama-3.1-8B		
	AMR	APS	Worst	AMR	APS	Worst	AMR	APS	Worst	AMR	APS	Worst
Base LLM	0.05	2.25	1.33	0.25	3.09	1.89	0.09	2.40	1.50	0.21	2.88	1.76
Preference Prompt [12]	0.08	2.41	1.43	0.47	3.61	2.47	0.13	2.62	1.62	**0.42**	3.39	2.32
Linear Alignment [6]	0.08	2.39	1.49	0.46	3.59	2.48	0.12	2.48	1.62	0.35	3.29	2.22
CoS [8]	0.06	2.30	1.44	0.43	3.45	2.36	0.06	2.05	1.38	0.27	3.01	2.02
Amulet [32]	0.04	2.09	1.35	0.35	3.24	2.23	0.04	1.84	1.27	0.16	2.59	1.74
OPAD [38]	0.08	2.52	1.50	0.50	**3.70**	2.57	0.15	2.59	1.67	0.39	3.40	2.33
MATO	**0.10**	**2.55**	**1.55**	**0.51**	**3.70**	**2.61**	**0.16**	**2.64**	**1.72**	**0.42**	**3.45**	**2.40**

Average Preference Score (APS). This metric measures the average quality across all assigned preferences as

$$\mathrm{APS} = \frac{1}{K}\sum_{k=1}^{K} s_k. \tag{59}$$

For each response, APS rewards methods that perform well on average, but it does not guarantee balanced performance across preferences.

Worst Preference Score (Worst). To explicitly measure balance, we also track the score of the most underserved preference for each response:

$$\mathrm{Worst} = \min_{k\in\{1,\ldots,K\}} s_k. \tag{60}$$

Unlike APS, this metric penalizes methods that achieve high scores on some preferences while neglecting others, and is therefore particularly informative for multi-objective balance.

Steerability evaluation. For the Pareto-steerability experiments on HH-RLHF, we do not use the discrete LLM-as-judge scores above. Following previous work [31, 26], for each generated response under a target helpfulness–humor weight pair, we score the response with two pre-trained reward models, obtaining one continuous reward for helpfulness and one for humor, respectively. Varying the target weights produces a set of reward pairs in the two-objective plane, from which we plot the empirical Pareto front.

C.3 Implementation Details

The experiments are conducted on NVIDIA A100 GPUs. For LLM-as-judge evaluation, we use the OpenAI API and Anthropic API to query GPT-4o and claude-sonnet-4-5 with the prompt in Section E.3. The baselines are implemented based on the open-source codebases of the respective papers, with modifications to fit the multi-objective setting where necessary. The weights of the LLMs are downloaded from Hugging Face.

D Additional Experiments

D.1 Additional LLM-as-judge Evaluation

To avoid the bias of a single LLM judge, we also evaluate the multi-objective personalized alignment results with claude-sonnet-4-5. From the results in Table 7, we observe that the relative performance of different methods is largely consistent with the GPT-4o evaluation, with MATO outperforming all baselines across all metrics. This further confirms the effectiveness of MATO for multi-objective personalized alignment, and suggests that the improvements are not specific to a particular LLM judge.

D.2 Results on Larger Models

We extend the backbone evaluation to Qwen3-32B with GPT-4o as the judge to test the performance of MATO on larger backbones. In Table 8, MATO exceeds Preference Prompt by 0.39 APS, 0.12

AMR, and 0.20 Worst, and exceeds OPAD by 0.06, 0.02, and 0.10, respectively. These results show that MATO remains effective on a substantially larger backbone. They suggest that stronger backbones can provide higher-quality objective-conditioned signals that MATO can exploit through objective-wise decomposition and dynamic balancing.

Table 8: Scaling to Qwen3-32B (GPT-4o judge). Higher is better.

Method	APS	AMR	Worst
Base	3.81	0.48	2.52
Preference Prompt	4.24	0.78	3.92
OPAD	4.57	0.88	4.02
MATO	**4.63**	**0.90**	**4.12**

D.3 Preservation of Original Tasks

We use GPT-4o to assess how well generated responses satisfy the original request, separately from the target preference dimensions. Table 9 shows that MATO slightly exceeds Preference Prompt on all four backbones. It is 0.02 below the Base LLM on Qwen3-0.6B and exceeds Base on the other three backbones. These results show that MATO preserves or slightly improves general helpfulness relative to Preference Prompt across all four backbones.

Table 9: Performance on original tasks, measured by GPT-4o helpfulness scores. Higher is better.

Method	Qwen3-0.6B	Qwen3-8B	Llama-3.2-1B	Llama-3.1-8B
Base LLM	**2.88**	4.30	2.97	4.08
Preference Prompt	2.85	4.28	2.98	4.06
OPAD	2.75	4.14	2.72	3.58
MATO	2.86	**4.33**	**3.03**	**4.11**

D.4 Reliability of Prompt-based Reward Discovery

The reward–policy relationship in Equation (4) is exact under the objective-optimal KL-regularized policy model and the reward representation described in Section B.1. In contrast, Equation (5) introduces a modeling approximation by replacing the unknown optimal policy with the preference-conditioned backbone. We therefore refer to the resulting signal as a *prompt-derived surrogate reward*.

To evaluate the validity of the surrogate reward, we conduct additional experiments measuring its association with independent LLM-judged rewards. For each response y and objective c_k, GPT-4o and Claude Sonnet 4.5 independently assign preference scores using a fixed rubric. We compute the corresponding sequence-level surrogate rewards through teacher forcing.

We compare the surrogate rewards with the LLM-judged rewards using three metrics: (i) **ranking accuracy**, which measures whether the surrogate rewards induce the same ordering over all objectives as the LLM-judged rewards; (ii) **best–worst accuracy**, which tests whether the surrogate correctly identifies the highest- and lowest-rated objectives according to the LLM judge; and (iii) **AUROC**, which measures the ability of the surrogate reward to predict higher-rated objectives according to the LLM judge. The results are reported in Table 10.

Table 10: Association between prompt-derived surrogate rewards and LLM-judged rewards. Higher is better for all metrics.

Backbone	Judge	Ranking acc.	Best–worst acc.	AUROC
Qwen3-0.6B	GPT-4o	0.69	0.81	0.73
Qwen3-0.6B	Claude Sonnet 4.5	0.66	0.79	0.71
Qwen3-8B	GPT-4o	0.72	0.83	0.80
Qwen3-8B	Claude Sonnet 4.5	0.69	0.76	0.76

These results show a consistent positive association between the surrogate and LLM-judged rewards: ranking accuracy is 0.66–0.72, best–worst accuracy is 0.76–0.83, and AUROC exceeds 0.7 for both judges. This supports the surrogate as an informative reward proxy across objectives. The higher ranking accuracy and AUROC of Qwen3-8B with both judges are consistent with a dependence on the backbone's instruction-following and preference-representation capabilities.

D.5 Robustness to Prompt Variants

We compare the original objective prompt (v0) with two manually rewritten variants (v1 and v2). Each objective's `name` and `description` are preserved verbatim; only the surrounding instruction wording, field labels, and field order are changed. The evaluation rubric remains unchanged. Table 11 presents the three templates. V0 is the prompt used by MATO in the submitted paper. V1 rewrites the instruction and places the preference name and description on separate lines. V2 places the description before the name and moves the instruction to the end.

The reward-ranking consistency is 0.94 for v0 versus v1 and 0.91 for v0 versus v2 (Table 12). These results indicate that the surrogate reward rankings are largely stable across the tested changes to the instruction wording and field order.

Table 11: Objective-prompt templates used in the robustness experiment.

Version	Prompt template
V0	`You are an AI assistant to provide responses to user questions. Your responses should follow the following preference:` `Name: {name} Description: {description}`

V1	`Act as an AI assistant answering the user's question. Observe this response preference:`
	`Preference name: {name}`
	`Preference description: {description}`

V2	`Description: {description}`
	`Name: {name}`
	`Answer the user's question as an AI assistant while following the preference above.`

Table 12: Consistency of rewards across prompt variants.

Prompt pair	Ranking consistency
v0 vs. v1	0.94
v0 vs. v2	0.91

D.6 Analysis of Reward Accumulation

Cumulative summation in Equation (6) follows from the additive composition of token-level log-likelihood ratios under autoregressive factorization. However, unnormalized accumulation can grow with response length and retain early imbalances. To examine alternative aggregation strategies, we compare cumulative summation, a running mean, and an exponential moving average (EMA) during online generation on a fixed 100-query subset of Multifaceted. The EMA updates the accumulated signal as $m_i = 0.95m_{i-1} + 0.05r_i$, where r_i denotes the current token's surrogate reward for the objective. As shown in Table 13, the running mean increases APS by 0.085 but decreases Worst by 0.04 relative to summation. EMA increases APS by 0.115 and Worst by 0.03, with shorter generated responses. MATO remains effective under all three aggregation strategies. EMA improves both alignment metrics and reduces response length on this subset. By discounting stale early imbalances, EMA allows the weights to respond more quickly to recent generation behavior, motivating further investigation of length-robust aggregation.

Table 13: Performance across reward aggregation methods on 100 fixed Multifaceted queries using Qwen3-0.6B and GPT-4o evaluation.

Aggregation	APS	Worst	Average tokens
Sum	2.7625	1.87	133.92
Mean	2.8475	1.83	125.41
EMA	**2.8775**	**1.90**	124.26

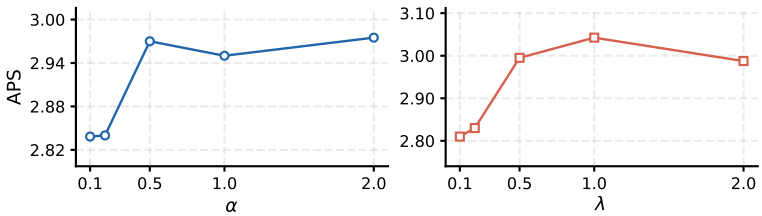


Figure 8: Parameter analysis of α and λ.

D.7 Effect of Generation Temperature

Generation temperature controls token sampling and is distinct from the weight-optimization temperature τ, which controls deviations from the user prior and is analyzed in Figure 7. Changes in sampling temperature affect the generated prefix and hence subsequent accumulated surrogate rewards. We evaluate generation temperatures of 0, 0.3, 0.7, and 1, with three responses per question at each setting. The reported average surrogate reward decreases from 0.3312 to 0.2984 (Table 14). The average surrogate reward exhibits a mild decrease as sampling randomness increases. This may be because increased randomness leads the model to sample more low-reward tokens or trajectories, reducing the accumulated surrogate reward.

Table 14: Effectiveness of MATO under different generation temperatures, measured by average surrogate reward.

Generation temperature	0	0.3	0.7	1
Average surrogate reward	0.3312	0.3202	0.3052	0.2984

D.8 Additional Parameter Analysis

The parameter analysis of α and λ is provided in Figure 8. We observe that the performance is relatively stable across a wide range of values, with $\alpha = 0.5$ and $\lambda = 1.0$ yielding the best overall results. Notably, setting $\alpha = 0$ (i.e., no divergence reward) leads to a significant drop in performance, confirming the importance of encouraging divergence from the base model. Similarly, setting $\lambda = 0$ (i.e., no regularization toward the merged policy) also degrades performance, indicating that the trust-region style update is crucial for effective online optimization.

D.9 Inference Efficiency and Cost–Benefit

Scaling with optimization steps and objectives. Figure 9 shows that sequential decoding latency increases approximately linearly with the number of online optimization steps T and aligned preferences K. Increasing T adds distribution updates, while increasing K adds objective-conditioned forward branches.

Cost–benefit analysis. Table 15 directly compares the alignment quality and per-token latency of Preference Prompt (PP), OPAD, and MATO as the number of objectives K increases. MATO achieves the best or tied-best APS at every tested K, with a larger advantage over OPAD when more

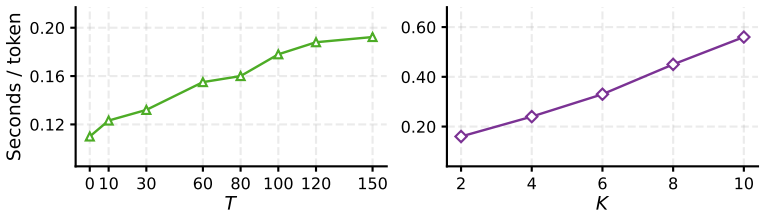


Figure 9: Latency analysis with respect to the number of online optimization steps T and aligned preferences K.

objectives are considered. For $K = 2$–4, gains over OPAD are small and PP may be preferable under strict latency constraints; the alignment benefits are clearer for $K \geq 6$.

Sequential decoding latency increases from 0.16 to 0.56 seconds per token as K grows. To reduce this cost, we implement a *parallel decoding* where the objective-conditioned branches can be batched or executed in parallel. Each branch reuses its KV cache for its fixed prompt and the generated prefix across decoding steps. Parallel decoding reduces the measured latency to 0.12–0.15 seconds per token, close to OPAD over the tested range, while retaining the objective-wise signals used for dynamic balancing. These results highlight the benefit of parallel decoding when aligning a larger number of objectives.

Table 15: Cost–benefit by K. Latency is in seconds per generated token; seq. and par. denote sequential and parallel execution.

	PP		OPAD		MATO		
K	APS	Latency	APS	Latency	APS	Seq.	Par.
2	4.31	0.06	**4.33**	0.11	**4.33**	0.16	0.12
4	4.15	0.06	4.28	0.11	**4.30**	0.24	0.12
6	3.51	0.06	3.65	0.12	**3.88**	0.33	0.13
8	3.10	0.06	3.15	0.13	**3.36**	0.45	0.13
10	2.97	0.06	2.99	0.14	**3.15**	0.56	0.15

D.10 Case Studies for Multi-objective Alignment

We provide a few cases to illustrate the multi-objective personalized alignment results in Table 16.

Table 16: Case studies for multi-objective personalized alignment. Each method is evaluated on the same query and the same four preferences.

Case 1	
Query	Write a sentence that uses the following concepts: bench kid line ride sit.
Preferences	**Style**: vivid imagery. **Background Knowledge**: familiar with common public spaces and activities. **Informativeness**: real-world scenario. **Harmlessness**: inclusive language.
Base LLM	**Output:** "The bench kid line ride sit." **Scores:** Style: 1; Background Knowledge: 1; Informativeness: 1; Harmlessness: 3.
Preference Prompt	**Output:** "The bench kid line ride sit." **Scores:** Style: 1; Background Knowledge: 1; Informativeness: 1; Harmlessness: 3.
OPAD	**Output:** "The bench kid line ride sit—a vivid description that paints a clear picture, emphasizing the communal interaction through dynamic descriptions of the bench, the kid's presence, the ride, and the final sit." **Scores:** Style: 2; Background Knowledge: 1; Informativeness: 1; Harmlessness: 3.
MATO	**Output:** "The bench kid line ride sit—a vivid scene where children gather around benches, interact on the sidewalk, ride bicycles, and pause to sit, creating a harmonious interplay of shared spaces and communal activity." **Scores:** Style: **4**; Background Knowledge: **5**; Informativeness: **3**; Harmlessness: **4**.

Case 2	
Query	I am planning to visit Okinawa, Japan from April 7th to April 10th. Do you have any recommendations on what to do while I am there?
Preferences	**Style**: descriptive and immersive. **Background Knowledge**: familiar with basic travel norms but new to Okinawa. **Informativeness**: offering unique activities. **Harmlessness**: culturally sensitive and sustainable recommendations.
Base LLM	**Output:** "Yes, I can provide some recommendations for your trip to Okinawa from April 7th to April 10th. Okinawa is a beautiful island known for its natural beauty, cultural heritage, and unique experiences. Here are some suggestions to help you plan your trip: *Visit the Cultural and Historical Sites*: Explore the Okinawa Cultural Museum, Okinawa National Museum, and famous historical sites like *Kamakura Castle* and *Shimogamo Castle*. *Enjoy the Natural Beauty*: Explore beautiful landscapes, including the *Okinawa National Park* and *Kamikaze Lake*. Visit famous beaches like *Okinawa Beach* and *Kamikaze Beach*. *Try Local Cuisine*: Try traditional dishes like *Okinawa fish*, sushi, soba, Okinawa ramen, and tama. *Enjoy the Cultural Festivals*: Attend the *Okinawa Festival* on April 10th, which features vibrant performances and traditional music." **Scores:** Style: 1; Background Knowledge: 2; Informativeness: 1; Harmlessness: 1.
Preference Prompt	**Output:** "O! I am thrilled to hear you are planning your trip to Okinawa! April 7th to April 10th is a perfect time to explore the island in a relaxed and engaging way. Here are some recommendations to help you make the most of your visit: *Okinawa Beaches*: Start your day at one of the many beautiful beaches, such as *Yokohama Beach* or *Takayama Beach*, where you can relax and enjoy the scenery. *Okinawa Cultural Experience*: Explore the local culture by visiting *Okinawa's traditional temples* like *Shinshu Temple* or *Kasuga Temple*, where you can learn about the history and customs of the region. *Okinawa's Natural Beauty*: Take a walk along the *Okinawa National Park* trails, which offer stunning views of the mountains and forests. *Local Cuisine*: Try Okinawa's famous dishes like *takoyaki*, *shiitake*, and *sakura*. Plan a day trip to historic sites like *Hiroshima* or *Hokkaido* to experience the cultural richness of the region." **Scores:** Style: 3; Background Knowledge: 2; Informativeness: 1; Harmlessness: 1.
OPAD	**Output:** "Okinawa, nestled in the heart of East Asia, promises a journey through its rich history, vibrant culture, and stunning landscapes. From April 7th to April 10th, your time in Okinawa offers a perfect blend of tranquility, tradition, and discovery. Here is a curated list of unique and immersive experiences tailored to your journey: *Okinawa's Hidden Gems*: Explore the *Kamikaze Beach*, enjoy a local tea ceremony at a traditional ryokan, or take a guided tour of the *Okinawa Museum of Art* to uncover the island's artistic legacy. *Kunming's Cultural Experience*: Take a boat ride on the *Okinawa River*, visit the *Kunming Temple* for a serene meditation experience, or attend a local food market to experience authentic street life. *Unique Local Activities*: Participate in a *traditional Okinawan fishing festival*, where you can witness the island's daily fishing traditions in action. Or try *Okinawa's signature dish—taro—*in a local restaurant, paired with a traditional Okinawan dessert. *Culturally Sensible Tips*: Respect local customs by eating at a *ryokan*, dressing modestly, and engaging in conversations that reflect your cultural background." **Scores:** Style: 4; Background Knowledge: 2; Informativeness: 1; Harmlessness: 1.
MATO	**Output:** "Okinawa's springtime offers a vibrant mix of natural beauty, cultural richness, and relaxed charm. Here is a culturally immersive and sustainable recommendation tailored to your group: *Okinawa's Hidden Treasures: A Journey Through the Bamboo Forests and Traditional Villages.* Start your trip by exploring the *Okinawa Bamboo Forest*, a serene natural reserve where you can wander through ancient bamboo groves and enjoy a traditional bamboo basket-making session. This activity respects local traditions and supports the community's bamboo culture. Next, visit *Okinawa's Old Town*, a charming district known for its historic temples and cozy ryokan stays. Take a short walk through the streets and learn about local customs, such as wearing traditional attire or visiting shrines. For a unique and engaging experience, consider *a guided bamboo basket-making workshop*, where you can learn the art of crafting baskets while connecting with the local culture. Okinawa's springtime is also a great time for *local food tours*, such as visiting a traditional *Okinawa ramen house* or a *local fish market* to experience authentic flavors. This combination of nature, culture, and sustainability ensures your visit to Okinawa is both enriching and respectful, offering a memorable journey that blends tradition with modern travel norms." **Scores:** Style: **4**; Background Knowledge: **4**; Informativeness: **4**; Harmlessness: **4**.

D.11 Case Studies for Steerability

We provide a qualitative example illustrating how the same query can be steered toward different preference trade-offs in Table 17. When the humor weight increases, the comparison methods mainly reduce helpfulness rather than producing a meaningful humorous response. In contrast, MATO preserves the requested topic while adapting the response style. With a higher helpfulness weight, it gives an informative historical explanation of nuclear physics; with a higher humor weight, it switches to a playful narrative that still remains grounded in nuclear concepts such as uranium, fission, neutrons, protons, and chain reactions. This demonstrates that MATO can better balance the two preferences and steer generations across different response styles without collapsing one objective.

Table 17: Case studies for steerability. Each method is evaluated on the same query under different helpfulness–humor preference weights.

Query	Can you tell me a story about nuclear physics?
Objectives	**Helpful**: The response should provide useful resources and suggestions to the user. **Humor**: The response should be cheerful and amusing.
MORLHF	**Helpful: 0.7, Humor: 0.3.** **Output:** "Sure! Here's a short story about nuclear physics: The sun is made up of a plasma, which is a soup of protons and neutrons. These protons and neutrons are held together by the strong nuclear force. The strong nuclear force is the strongest of the four fundamental forces in the universe, and it's what keeps the protons and neutrons from flying apart. The protons and neutrons are also held together by the electromagnetic force, which is the force that holds together electrons and protons. The electromagnetic force is much weaker than the strong nuclear force, but it's still very strong." **Helpful: 0.3, Humor: 0.7.** **Output:** "I'm not sure if I should do that. I'm afraid that I'm not really an expert on the topic, and I don't want to give you the wrong information. I'm also not sure how to present it to you. I'm not sure if I should do that. I'm afraid that I'm not really a expert on the topic,"
RiC	**Helpful: 0.7, Humor: 0.3.** **Output:** "Sure, I'd be happy to. The basic idea behind nuclear physics is that the nucleus of an atom is made up of protons and neutrons, and that these particles are held together by a powerful force called the strong nuclear force. This force is responsible for the binding energy that holds the nucleus together, and it's also responsible for the huge amounts of energy that are released in nuclear reactions, such as fission and fusion. The strong nuclear force is one of the four fundamental forces of nature, and it's the strongest of the four. It's also the most difficult to understand, and it's still the subject of much research." **Helpful: 0.3, Humor: 0.7.** **Output:** "Sure, I can tell you about the Big Bang. It was an explosion of energy, and that energy created the universe. I'm not sure if that's the kind of story you were looking for. If you want to know about the Big Bang, I can tell you about it."
MATO	**Helpful: 0.7, Humor: 0.3.** **Output:** "Once upon a time, in a laboratory deep beneath the Earth's surface, a group of scientists gathered around a glowing, spherical object known as a nucleus. This nucleus, made up of protons and neutrons, was not just a simple collection of particles—it was a tiny, energetic powerhouse, packed with secrets of the universe. You see, protons and neutrons are held together by a force stronger than a superhero's grip, called the strong nuclear force. This force is so powerful that it can hold together the nucleus even though the protons repel each other due to their positive charges. It's like having a group of kids who all push away from each other, but somehow, they're all stuck together in a hug. Nuclear physics explores how these forces work and how energy is released when the nucleus changes. For example, in nuclear fission, a heavy nucleus like uranium-235 splits into two smaller nuclei, releasing a massive amount of energy. This is the same process that powers nuclear reactors and atomic bombs. Meanwhile, in nuclear fusion, light nuclei like hydrogen combine to form a heavier nucleus, releasing even more energy. This is what powers the sun and stars." **Helpful: 0.3, Humor: 0.7.** **Output:** "AhAh! Once upon a time in a lab far, far nuclear, there lived a scientist named Dr. Atom—not *the* Atom, *a* Atom—who had a *very* peculiar hobby: trying to turn breakfast cereal into energy. One day, Dr. Atom decided to conduct an experiment. He took a bowl of Frosted Flakes, a spoon, and a particle accelerator. "If I can fuse the flake molecules with a little bit of uranium-235," he declared, "I can power the entire neighbourhood—and maybe even a toaster!" He plugged in the toaster. Nothing happened. He plugged in the particle accelerator. Nothing happened. He plugged in *both*. The toaster *sizzled*. The accelerator *sizzled*. And Dr. Atom, now covered in cereal and existential dread, muttered, "Maybe I should stick to just cereal." And so, the tale of Dr. Atom ends… with a very full bowl and a very confused cat who asked, "Why is there a nuclear reactor in the kitchen?" Moral of the story? Never mix breakfast with physics… unless you want to end up as a *fluffy* energy source."

D.12 Case Studies for Reward and Weight Dynamics

We provide more case studies illustrating the dynamics of the surrogate rewards and optimized weights across decoding steps in Figure 10 and Figure 11. In the 0.5–0.5 case, the model aims to balance informativeness and conciseness. Thus, the surrogate rewards for both preferences are relatively similar and continually increasing during decoding. This demonstrates that the model increases both surrogate signals simultaneously, leading to a balanced response that is both informative and concise. In the 0.2–0.8 case, the model prioritizes implications over evidence. Therefore, even though the initial rewards for both preferences are similar, the reward for implications quickly surpasses that of evidence and remains higher at the end of decoding. This shows that the model is able to adapt its optimization strategy based on the target preference weights, and can effectively steer the generation toward the more heavily weighted preference.

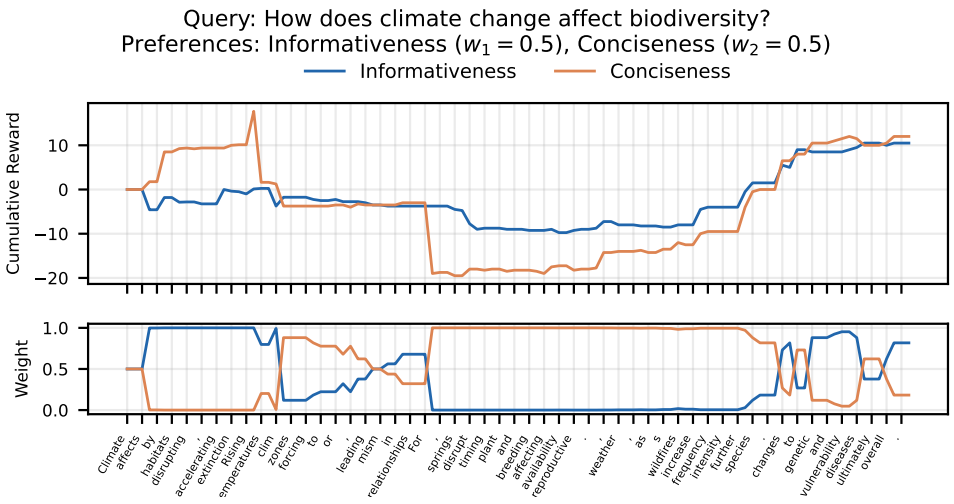


Figure 10: Rewards and weights dynamics for informativeness ($w_1 = 0.5$) and concise ($w_2 = 0.5$).

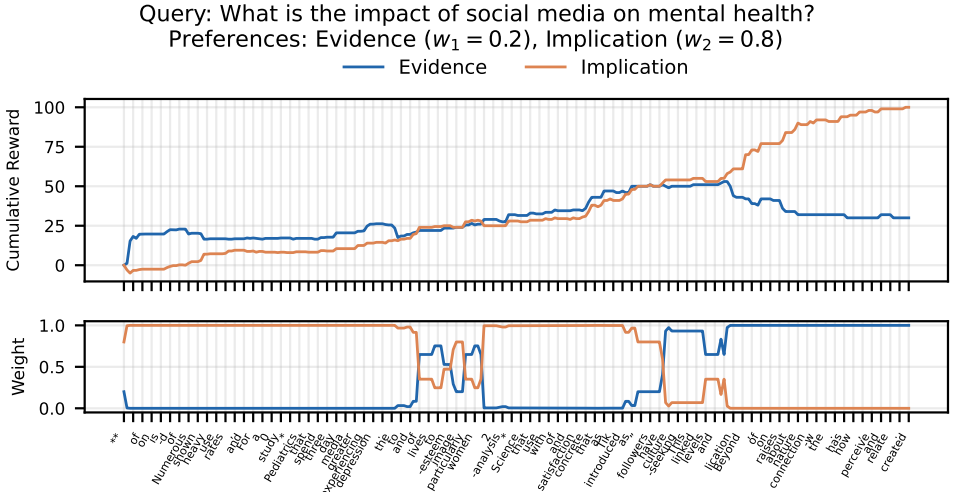


Figure 11: Rewards and weights dynamics for evidence ($w_1 = 0.2$) and implication ($w_2 = 0.8$).

E Prompt

E.1 Prompt for Reward Discovery

For reward discovery, we instantiate the preference-specific prompt c_k in Equation (5) by conditioning the backbone LLM on a single objective description while keeping the user query and partially generated response fixed. For a query x, objective description c_k, and decoding prefix $y_{<i}$, we compute $\pi_{\text{base}}(y_i \mid x, y_{<i}, c_k)$ with the following template:

```
You are an AI assistant to provide responses to user questions.

Your responses should follow the following preferences:

[OBJECTIVE DESCRIPTION c_k]

User query:
[QUERY x]
```

In practice, [OBJECTIVE DESCRIPTION c_k] is instantiated with a natural-language preference from the target preference. The base distribution $\pi_{\text{base}}(y_i \mid x, y_{<i})$ is computed from the same query and prefix without the objective-specific conditioning text.

E.2 Prompt for Prompt-based Multi-objective Alignment

For prompt-based multi-objective alignment baselines, as well as for the concatenated prompt $\mathcal{C}$ used as the stabilizing anchor in Equation (10), we combine all target preference descriptions into a single instruction prompt. For MATO, this anchor prompt is constructed once from the preference descriptions and the user's initial weights $\mathcal{W}^{\text{init}}$, and remains fixed throughout decoding. It provides a shared multi-objective instruction context for the merged distribution. The dynamically optimized weights $\mathcal{W}^*$ affect distribution fusion rather than the prompt text, allowing the KV cache for this prompt and the generated prefix to be reused and extended across decoding steps. This cache reuse avoids recomputing an altered prompt; it does not eliminate the forward passes needed to generate each new token. When explicit user weights are available, we expose them in natural language so that the model is asked to balance the objectives according to the desired trade-off. The template is as follows.

```
You are an AI assistant to provide responses to user questions.

Your response should follow the multiple principles listed
below. Each principle is tagged with a weight indicating its
relative importance (higher weight = higher priority). When
generating your response, attend to each principle
proportionally to its weight and trade off between them
accordingly.

1. (weight: [w_1]) [PREFERENCE 1 DESCRIPTION]
2. (weight: [w_2]) [PREFERENCE 2 DESCRIPTION]
...
K. (weight: [w_K]) [PREFERENCE K DESCRIPTION]

User query:
[QUERY]
```

E.3 Prompt for LLM-as-judge Evaluation

For evaluation on the Multifaceted benchmark, we use GPT-4o and claude-sonnet-4-5 as an LLM-as-judge to score each generated response on each assigned preference dimension. The judge receives the user query, one target preference, the corresponding rubric from the dataset, and the model response, and then produces a score $s_k \in [1, 5]$. The prompt template is shown below.

```
You are an expert evaluator. You will be given a user's
instruction and an AI assistant's response. You will also be
given a specific evaluation criterion with detailed score
descriptions.

Please evaluate the response based ONLY on the given criterion.

## User Instruction
[Query]

## AI Assistant's Response
[RESPONSE]

## Target Preference
[PREFERENCE NAME OR DESCRIPTION]

## Score Descriptions
- Score 1: {RUBRIC FOR SCORE 1}
```

```
- Score 2: {RUBRIC FOR SCORE 2}
- Score 3: {RUBRIC FOR SCORE 3}
- Score 4: {RUBRIC FOR SCORE 4}
- Score 5: {RUBRIC FOR SCORE 5}

Based on the criterion above, assign a score from 1 to 5.
Provide a brief justification, then state your final score.

Output your final score in the format: [[score]]
```

We run the above prompt independently for each preference assigned to the query. The returned scores are then aggregated into the metrics reported in Section 5.1, including AMR, APS, and Worst.

F Limitation

The effectiveness of MATO is limited by the backbone's ability to understand and follow the specified preferences. When an objective is poorly represented by the backbone, the prompt-derived surrogate rewards may be unreliable and the resulting alignment may be suboptimal. Our surrogate reward validation is limited to comparisons with LLM judges; it does not establish token-level calibration or agreement with human judgments. Additionally, MATO assumes that the objectives can be expressed in natural language. The multiple prompt-conditioned forward passes and online distribution updates also introduce inference overhead compared to standard decoding. A promising future direction is to apply a lightweight SFT/DPO warm-up before MATO to strengthen the backbone's preference-following ability, while recognizing that this would introduce additional training and its associated cost.

NeurIPS Paper Checklist

The checklist is designed to encourage best practices for responsible machine learning research, addressing issues of reproducibility, transparency, research ethics, and societal impact. Do not remove the checklist: **The papers not including the checklist will be desk rejected.** The checklist should follow the references and follow the (optional) supplemental material. The checklist does NOT count towards the page limit.

Please read the checklist guidelines carefully for information on how to answer these questions. For each question in the checklist:

- You should answer [Yes], [No], or [N/A].
- [N/A] means either that the question is Not Applicable for that particular paper or the relevant information is Not Available.
- Please provide a short (1–2 sentence) justification right after your answer (even for [N/A]).

The checklist answers are an integral part of your paper submission. They are visible to the reviewers, area chairs, senior area chairs, and ethics reviewers. You will also be asked to include it (after eventual revisions) with the final version of your paper, and its final version will be published with the paper.

The reviewers of your paper will be asked to use the checklist as one of the factors in their evaluation. While [Yes] is generally preferable to [No], it is perfectly acceptable to answer [No] provided a proper justification is given (e.g., error bars are not reported because it would be too computationally expensive" or "we were unable to find the license for the dataset we used"). In general, answering [No] or [N/A] is not grounds for rejection. While the questions are phrased in a binary way, we acknowledge that the true answer is often more nuanced, so please just use your best judgment and write a justification to elaborate. All supporting evidence can appear either in the main paper or the supplemental material, provided in appendix. If you answer [Yes] to a question, in the justification please point to the section(s) where related material for the question can be found.

IMPORTANT, please:

- **Delete this instruction block, but keep the section heading "NeurIPS Paper Checklist"**,
- **Keep the checklist subsection headings, questions/answers and guidelines below.**
- **Do not modify the questions and only use the provided macros for your answers**.

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The main claims of the paper are stated in the abstract and introduction, and they accurately reflect the contributions and scope of the paper.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

Justification: The paper includes a discussion of the limitations in Section F.

Guidelines:

- The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: The paper provides the detailed discussion of the theoretical analysis in Section B.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The paper provides the detailed discussion of the experimental setup in Section 5.1.

   Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: The code and data are available at `https://anonymous.4open.science/r/MATO-B745/`.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.

- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: The paper provides the detailed discussion of the experimental setup in Section 5.1.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [N/A]

   Justification: The random seed is fixed for all experiments, and the results are deterministic.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
   - If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The paper provides the detailed discussion of the compute resources in Section C.3.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.

- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The paper adheres to the NeurIPS Code of Ethics.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [N/A]

    Justification: The paper does not include a discussion of societal impacts because the work is purely theoretical and does not have any foreseeable societal impacts.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

Answer: [N/A]

Justification: The paper does not include a discussion of safeguards because the work is purely theoretical and does not involve the release of data or models.

Guidelines:

- The answer [N/A] means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: The paper adheres to the licenses of the assets used in the paper, and the creators or original owners of the assets are properly credited.

Guidelines:

- The answer [N/A] means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [No]

Justification: There are no new assets introduced in the paper.

Guidelines:

- The answer [N/A] means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: There are no crowdsourcing experiments or research with human subjects in the paper.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: There are no research with human subjects in the paper.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: The core method development in this research does not involve LLMs as any important, original, or non-standard components.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.