CIALLO: CORRELATION-AWARE MULTI-OBJECTIVE ALIGNMENT FOR LARGE LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Aligning large language models (LLMs) with diverse human preferences requires balancing multiple objectives that may be compatible or conflicting in different scenarios. However, existing multi-objective alignment methods often optimize different objectives independently, ignoring their correlated preference patterns and conflicting constraints. This oversight prevents the model from reusing shared information across compatible objectives and leveraging contrastive signals from conflicting ones, thereby reducing training efficiency. Motivated by this consideration, we propose CIALLO, a framework that first conducts separated training to learn an initial soft prompt representation for each alignment objective, and then performs joint training under mixed preference weights, where a dynamically tracked Fisher-gradient affinity matrix constrains objective relationships and guides Group Relative Policy Optimization (GRPO) through a Fisher Gram-matching regularizer. Experiments on Reddit Summary with Qwen3-0.6B and Helpful Assistant with Qwen3-4B show that CIALLO consistently achieves macro-averaged gains of 1.409 and 0.699 reward points over REWARD SOUPS.

1 INTRODUCTION

Large Language Models (LLMs) have demonstrated remarkable capabilities across a wide range of complex tasks, including coding and creative generation (Zhao et al., 2023; Bubeck et al., 2023). However, despite their impressive proficiency, LLMs are prone to generating outputs that exhibit hallucinations, biases, or harmful content, posing significant risks to real-world deployment (Ji et al., 2023; Huang et al., 2024). Ensuring that these models align with human values and intentions has become a paramount challenge. To address this, alignment techniques such as Reinforcement Learning from Human Feedback (RLHF) (Ouyang et al., 2022) have been widely adopted. These methods typically optimize the model toward a single, unified reward or preference objective, significantly improving the safety and utility of LLMs.

In practice, LLMs are inherently required to satisfy diverse and often competing values simultaneously. For instance, a model must balance being highly informative with avoiding unsafe advice, or navigate varying user standards regarding verbosity and formatting. To meet these multifaceted demands, there has been growing interest in Multi-Objective Alignment, which aims to equip models with the capability to navigate trade-offs between competing objectives (Li et al., 2020; Rame et al., 2023; Yang et al., 2024). To avoid the high cost of training separate models for every possible preference combination, existing studies predominantly rely on post-hoc or test-time adaptation methods. Prominent approaches include parameter interpolation (e.g., averaging weights of models fine-tuned on different objectives) (Rame et al., 2023; Shi et al., 2024), prompt-conditioned generation (Yang et al., 2024; Wang et al., 2024), and preference-aware reward modeling (Lin et al., 2025).

However, existing methods treat different alignment objectives as independent objectives and optimize them with separate constraints, without explicitly considering the relationships among the objectives. This independence assumption overlooks the fact that alignment objectives are intrinsically related, since they can impose mutually conflicting constraints. When these relationships are ignored, the model is forced to learn each objective from objective-specific signals, limiting its ability to reuse shared information among compatible objectives or leverage contrastive signals from conflicting ones, thereby reducing training and data efficiency. As a result, the learned representations can become fragmented, making it harder to identify consistent alignment directions and to generalize across

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

different preference settings. Therefore, decoupling should not imply complete separation; instead, it should distinguish objective-specific signals while preserving informative dependencies among objectives.

Motivated by this consideration, we propose CIALLO, i.e., Correlation-aware multI-objective ALignment for LLMs via joint Optimization. Specifically, CIALLO first conducts separated training to learn an initial soft prompt representation for each alignment objective. In the second stage, these objective-specific prompts are jointly trained under mixed preference weights, where a dynamically tracked Fisher-gradient affinity matrix is used to constrain the relationships among different objectives, encouraging compatible objectives to share useful information while separating conflicting ones. Our main contributions are summarized as follows:

- We identify an overlooked limitation in existing multi-objective alignment methods: treating different objectives as independent can ignore both compatible and conflicting relationships among them, leading to inefficient use of training data and fragmented alignment representations.
- We propose CIALLO, a two-stage framework that first learns objective-specific soft-prompt controllers and then jointly optimizes them with Group Relative Policy Optimization (GRPO) (Shao et al., 2024) under mixed preference weights, enabling continuous multi-objective control within a single model. Furthermore, CIALLO estimates local inter-objective relationships from objective-specific gradients in the Fisher metric and aligns the controller geometry with these relationships through Fisher Gram matching.
- Extensive experiments demonstrate the effectiveness of our method. Compared with REWARD SOUPS, our approach achieves macro-averaged gains of 1.409 and 0.699 reward points on Reddit Summary and Helpful Assistant, respectively.

2 RELATED WORK

Aligning Large Language Models (LLMs) with human values is a fundamental prerequisite for building safe and reliable AI systems (Ji et al., 2023). The standard paradigm, Reinforcement Learning from Human Feedback (RLHF) (Ouyang et al., 2022), typically involves training a reward model to approximate human preferences and subsequently optimizing the policy via algorithms like PPO (Schulman et al., 2017). Despite its empirical success, standard RLHF is often computationally expensive and unstable. To mitigate these issues, recent advancements have shifted towards Direct Preference Optimization (DPO) (Rafailov et al., 2023), which optimizes the policy directly on preference data without requiring an explicit reward model. Building upon this, researchers have developed various efficient variants to further enhance training stability and performance, such as SimPO (Meng et al., 2024), KTO (Ethayarajh et al., 2024), and ORPO (Hong et al., 2024).

Despite the significant efficiency improvements brought by these approaches, they predominantly model human preference as a single scalar reward. This simplification fundamentally overlooks the multifaceted complexity of real-world applications (Vamplew et al., 2018). In practice, LLMs are frequently required to simultaneously satisfy diverse and inherently conflicting objectives, such as balancing helpfulness with harmlessness, or navigating varying stylistic constraints (Bai et al., 2022). Optimizing toward a single unified objective often induces an "alignment tax", where forcing improvements in one dimension inevitably degrades performance in another. Consequently, a single-objective approach is insufficient to capture the intricate Pareto frontier of human values, necessitating alignment paradigms that can explicitly manage the trade-offs between competing objectives (Mukherjee et al., 2024).

To address the limitations of single-objective alignment, a growing body of literature has focused on multi-objective alignment. For training-time optimization, methods like MODPO (Zhou et al., 2024) and CPO (Guo et al., 2024b) extend direct preference learning to multi-objective scenarios by modifying the loss functions to explicitly balance multiple reward signals. To accommodate diverse user preferences dynamically at test-time without exhaustive retraining, various adaptation strategies have been proposed. For instance, Reward Soups (Rame et al., 2023) interpolates the network parameters of multiple expert models fine-tuned on distinct objectives, while MOD (Shi et al., 2024) dynamically fuses their output logits during the decoding phase. Other approaches, such as RIC (Yang et al., 2024) and DPA (Wang et al., 2024), explicitly inject preference weightings into the input prompts to steer the language model's generation space. More recently, advanced

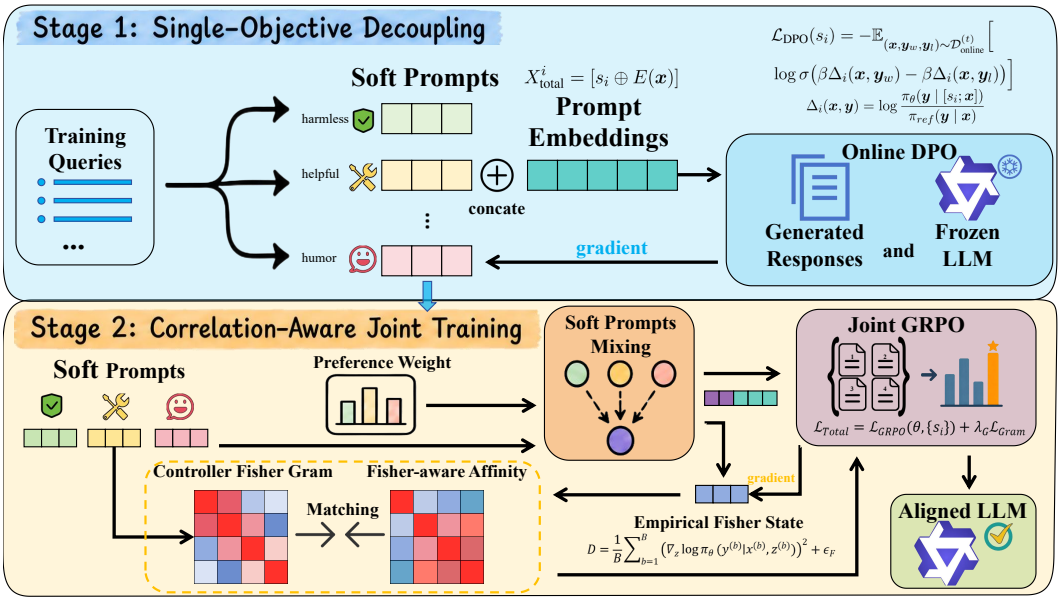


Figure 1: Illustration of the two-stage correlation-aware framework in CIALLO

test-time adaptation methods have emerged: PARM (Lin et al., 2025) trains a preference-aware autoregressive reward model to dynamically evaluate and guide the generation trajectory, whereas HoE (Li et al., 2025) utilizes a hierarchical Mixture-of-Experts architecture to seamlessly route and combine task-specific LoRA adapters based on spatial preference regions.

Our work is also closely related to recent studies on multi-objective LLM alignment, including **?** and **?**. While these works investigate prompt-aware user preference and dynamic preference adaptation, respectively, our work focuses on explicitly modeling inter-objective correlation.

3 PRELIMINARY

We first establish the formal notation and definitions for the task of aligning LLMs with human preferences. Let $\mathcal{V}$ be the vocabulary of a language model. The generation process is characterized by a policy $\pi : \mathcal{X} \to \mathcal{Y}$, which generates a response sequence $\boldsymbol{y} \in \mathcal{Y}$ for a given input query $\boldsymbol{x} \in \mathcal{X}$. Specifically, the input and output are represented as token sequences $\boldsymbol{x} = [x^1, x^2, \ldots, x^m]$ and $\boldsymbol{y} = [y^1, y^2, \ldots, y^n]$, where m and n denote their respective sequence lengths. The fundamental goal of alignment is to optimize the policy π such that the generated output $\boldsymbol{y}$ is maximally consistent with human values and specific constraints encoded within the context $\boldsymbol{x}$.

In multi-objective alignment scenarios, the model is required to satisfy K distinct and potentially competing objectives (e.g., helpfulness, safety, or verbosity) simultaneously. Each objective $k \in \{1, \ldots, K\}$ is associated with a specific reward model $r_k : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$. For any input-response pair $(\boldsymbol{x}, \boldsymbol{y})$, the multi-objective performance is represented by a reward vector $r(\boldsymbol{x}, \boldsymbol{y}) = [r_1(\boldsymbol{x}, \boldsymbol{y}), \ldots, r_K(\boldsymbol{x}, \boldsymbol{y})]^\top$. To navigate the trade-offs between these objectives, we introduce a preference weight vector $\boldsymbol{w} = [w_1, \ldots, w_K]^\top \in \Delta^{K-1}$, where $\sum_{k=1}^K w_k = 1$ and $w_k \geq 0$. The overall scalar reward for a given preference configuration is then formulated as the weighted linear combination:

$$R(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{w}) = \sum_{k=1}^{K} w_k r_k(\boldsymbol{x}, \boldsymbol{y}) = \boldsymbol{w}^\top r(\boldsymbol{x}, \boldsymbol{y}) \tag{1}$$

Algorithm 1 CIALLO

Input: Pre-trained LLM π_θ, reference model π_{ref}, reward models $\{R_i\}_{i=1}^K$, dataset $\mathcal{D}$.
Output: Optimized policy π_θ and soft prompts $\{s_i\}_{i=1}^K$.
Initialize the shared soft prompt s_0 and objective-specific soft prompts $\{s_i\}_{i=1}^K$; freeze θ.
// Stage 1: Single-objective Decoupling
for $i = 1$ to K **do**
  Optimize s_i under objective R_i with $\mathcal{L}_{\text{DPO}}$ in Eq. 4.
end for
// Stage 2: Fisher-aware Multi-objective Joint Training
Unfreeze θ, initialize $C = I_K$ and $D = \mathbf{1}$.
for iteration $t = 0, 1, 2, \ldots$ **do**
  Sample queries and preference weights $\boldsymbol{w}$.
  Obtain s_{mix} by Eq. 5, generate a response group, and compute all rewards.
  Compute the preference-weighted $\mathcal{L}_{\text{GRPO}}$ and the per-objective losses $\{\mathcal{L}_i\}_{i=1}^K$.
  if t reaches the relation-update interval **then**
    Compute $\{g_i\}_{i=1}^K$, update D, and update C by Eqs. 9–17.
  end if
  Compute the controller Gram matrix P and $\mathcal{L}_{\text{Gram}}$ by Eqs. 14–15.
  Update θ and $\{s_i\}_{i=1}^K$ by minimizing $\mathcal{L}_{\text{Total}}$ in Eq. 16.
end for

4 METHOD

As shown in Fig. 1, CIALLO follows a two-stage design for Fisher-aware multi-objective alignment. The framework first performs single-objective decoupling training, where each alignment objective is optimized separately to learn objective-specific representations and reduce the interference from other objectives. It then conducts multi-objective joint training to estimate local objective interactions and coordinate their controller geometry, so that the learned representations can remain objective-specific while retaining informative relationships among objectives. The overall training procedure is summarized in Algorithm 1.

4.1 SINGLE-OBJECTIVE STAGE

In the first stage, our goal is to learn a set of objective-specific soft prompts (Lester et al., 2021) $\{s_i\}_{i=1}^K$, where each prompt serves as a lightweight controller for a single alignment objective. Rather than fully fine-tuning the LLM for every objective, we adopt soft prompt tuning to preserve the model's general capabilities while reducing computational overhead. Accordingly, the backbone parameters θ are frozen throughout this stage, and only the continuous prompt parameters are updated.

All objective-specific prompts use the same initial value s_0, which is saved before optimization so that the stage-two controller directions remain anchored to a common reference. Concretely, for each alignment objective i, we define a learnable soft prompt $s_i \in \mathbb{R}^{L\times d}$, consisting of L continuous vectors of dimensionality d. For a given input sequence $\boldsymbol{x} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\}$, let $E(\boldsymbol{x}) \in \mathbb{R}^{n\times d}$ denote its fixed embedding sequence. The prompt-conditioned input representation is constructed by prepending s_i to the input embeddings:

$$X^i_{\text{total}} = [s_i \oplus E(\boldsymbol{x})] \tag{2}$$

where $\oplus$ denotes concatenation along the sequence dimension. The conditional response distribution under objective i is therefore written as $\pi_\theta(\boldsymbol{y} \mid [s_i; \boldsymbol{x}])$. In this way, different alignment objectives are decomposed into different prompt-induced behavioral shifts over the same frozen backbone model.

Based on this formulation, we optimize each soft prompt independently using Online DPO (Guo et al., 2024a). Compared with offline preference optimization, Online DPO dynamically generates preference pairs from the current policy, which helps maintain distribution consistency during training. At each training iteration t, for a query $\boldsymbol{x}$ sampled from the prompt distribution $\mathcal{D}_{\text{prompt}}$, the model generates a pair of responses $\boldsymbol{y}_1, \boldsymbol{y}_2$ conditioned on the current soft prompt $s_i^{(t)}$. These responses are then evaluated and ranked by the corresponding reward model R_i to form an online preference triplet

$(\boldsymbol{x}, \boldsymbol{y}_w, \boldsymbol{y}_l)$:

$$\mathcal{D}^{(t)}_{\text{online}} = \{(\boldsymbol{x}, \boldsymbol{y}_w, \boldsymbol{y}_l) \mid \boldsymbol{x} \sim \mathcal{D}_{\text{prompt}}, \boldsymbol{y}_1, \boldsymbol{y}_2 \sim \pi_\theta(\cdot \mid [s_i^{(t)}; \boldsymbol{x}]), \boldsymbol{y}_w = \text{rank}(\boldsymbol{y}_1, \boldsymbol{y}_2, r_i)\} \tag{3}$$

where rank function selects the winning response according to the reward score, $\boldsymbol{y}_w$ denotes the winning response with a higher reward score. The optimization is performed by minimizing the DPO loss. In our framework, the reference model π_{ref} is defined as the original clean model without any soft prompt intervention, representing the model's intrinsic output distribution. The objective for updating s_i is:

$$\mathcal{L}_{\text{DPO}}(s_i) = -\mathbb{E}_{(\boldsymbol{x}, \boldsymbol{y}_w, \boldsymbol{y}_l) \sim \mathcal{D}^{(t)}_{\text{online}}} \Big[\log \sigma\big(\beta_{\text{DPO}} \Delta_i(\boldsymbol{x}, \boldsymbol{y}_w) - \beta_{\text{DPO}} \Delta_i(\boldsymbol{x}, \boldsymbol{y}_l)\big) \Big] \tag{4}$$

where

$$\Delta_i(\boldsymbol{x}, \boldsymbol{y}) = \log \frac{\pi_\theta(\boldsymbol{y} \mid [s_i; \boldsymbol{x}])}{\pi_{ref}(\boldsymbol{y} \mid \boldsymbol{x})},$$

and $\pi_{ref}(\boldsymbol{y} \mid \boldsymbol{x})$ serves as a static baseline. By minimizing this loss, each soft prompt s_i learns to shift the LLM's output distribution from its original state toward the specific preference encoded by objective i.

4.2 Multi-objective Stage

Unlike the first stage, the second stage jointly updates the backbone parameters θ and the soft-prompt bank. The preference vector $\boldsymbol{w} = [w_1, \ldots, w_K]^\top \in \Delta^{K-1}$ determines both the mixed controller and the scalar reward used by the policy objective. Before stage 2, we retain the common initialization s_0 used for all stage-one prompts. Defining the anchor-centered controller directions $d_i = s_i - s_0$, the mixed prompt can be written as:

$$s_{\text{mix}}(\boldsymbol{w}) = \sum_{i=1}^{K} w_i s_i = s_0 + \sum_{i=1}^{K} w_i d_i \tag{5}$$

4.2.1 Objective-Specific Losses for Relation Estimation

We employ GRPO for joint training. For each query, the mixed prompt generates a group of G responses $\{\boldsymbol{y}_g\}_{g=1}^{G}$. All responses are evaluated by every reward model, producing the reward matrix $R \in \mathbb{R}^{G \times K}$ with $R_{g,i} = r_i(\boldsymbol{x}, \boldsymbol{y}_g)$. To estimate objective relationships independently of preference scalarization, we normalize each reward dimension within the group:

$$A_{g,i} = \frac{R_{g,i} - \mu_i}{\sigma_i}, \qquad \mu_i = \frac{1}{G} \sum_{g=1}^{G} R_{g,i}, \quad \sigma_i^2 = \frac{1}{G} \sum_{g=1}^{G} (R_{g,i} - \mu_i)^2 \tag{6}$$

The quantity $A_{g,i}$ is the group-relative advantage of response g for objective i.

Let

$$\rho_g = \frac{\pi_\theta(\boldsymbol{y}_g \mid \boldsymbol{x}, z)}{\pi_{\text{old}}(\boldsymbol{y}_g \mid \boldsymbol{x}, z)}$$

denote the probability ratio between the current and rollout policies. The objective-specific losses used to estimate the Fisher-aware relationship are:

$$\mathcal{L}_i = -\frac{1}{G} \sum_{g=1}^{G} \min(\rho_g A_{g,i}, \text{clip}(\rho_g, 1-\epsilon, 1+\epsilon) A_{g,i}) \tag{7}$$

4.2.2 Fisher-aware Objective Relationship

Pairwise gradient similarity has been widely used to characterize task cooperation and interference in multi-task optimization (Yu et al., 2020). Motivated by these connections, we define a normalized Fisher-preconditioned gradient similarity. For each objective, we differentiate its objective-specific loss $\mathcal{L}_i$ with respect to the shared mixed-prompt activation, rather than with respect to the entire prompt bank:

$$g_i = \text{vec}(\nabla_z \mathcal{L}_i), \qquad \bar{g}_i = \frac{1}{B} \sum_{b=1}^{B} g_i^{(b)} \tag{8}$$

where $z = s_{\text{mix}}(\boldsymbol{w})$. This places all relation estimates in the same $L \times d$ controller space and prevents the preference coefficient of a bank row from directly rescaling the estimate. In practice, the reduction over the current relation batch directly provides the aggregated gradient $\bar{g}_i$.

We use a diagonal Fisher state in the mixed-prompt space. Its score-gradient form is:

$$D = \text{Diag}\left[\text{vec}\left(\frac{1}{B}\sum_{b=1}^{B}\left(\nabla_z \log \pi_\theta(\boldsymbol{y}^{(b)} \mid \boldsymbol{x}^{(b)}, z^{(b)})\right)^2 + \epsilon_F\right)\right] \tag{9}$$

where $D \in \mathbb{R}^{p\times p}$, $p = Ld$, is diagonal. In practice, only its $L \times d$ diagonal-entry tensor is stored, and all products are computed elementwise.

The Fisher-aware affinity between objectives i and j is:

$$\widehat{C}_{ij} = \frac{\bar{g}_i^\top D^{-1}\bar{g}_j}{\sqrt{\bar{g}_i^\top D^{-1}\bar{g}_i}\sqrt{\bar{g}_j^\top D^{-1}\bar{g}_j} + \epsilon_C}, \qquad \widehat{C}_{ii} = 1 \tag{10}$$

Here D^{-1} is interpreted elementwise, with a small damping term when needed. Because g_i is a loss gradient, positive affinity indicates compatible local descent directions, while negative affinity indicates a local conflict. The magnitude measures the strength of the first-order transfer.

4.2.3 Joint GRPO and Fisher Gram Matching

Gram matrices provide a standard representation of pairwise geometry, and matching such relational structures has been used to preserve similarity patterns across representation spaces (Tung & Mori, 2019). Following this principle, we construct a Fisher-metric controller Gram matrix.

For the preference-conditioned policy update, we aggregate the objective-specific advantages using the sampled preference vector:

$$\hat{A}_g = \sum_{i=1}^{K} w_i A_{g,i} \tag{11}$$

The resulting $\hat{A}_g$ directly represents the relative quality of response g under preference $\boldsymbol{w}$. Substituting this preference-weighted advantage into the clipped GRPO objective gives

$$\mathcal{L}_{\text{GRPO}}(\theta, \{s_i\}) = -\frac{1}{G}\sum_{g=1}^{G}\min\Big(\rho_g \hat{A}_g, \text{clip}(\rho_g, 1-\epsilon, 1+\epsilon)\hat{A}_g\Big) + \beta\mathcal{D}_{KL}^{mix} \tag{12}$$

The mixed-prompt KL penalty is:

$$\mathcal{D}_{KL}^{mix} = \mathbb{D}_{KL}(\pi_\theta(\cdot \mid [s_{mix}; \boldsymbol{x}]) \,\|\, \pi_{ref}(\cdot \mid \boldsymbol{x})) \tag{13}$$

where π_{ref} is frozen and β controls this penalty.

To align the prompt controller geometry with the local policy sensitivity, we form the Fisher-metric controller Gram matrix from the anchor-centered directions d_i:

$$P_{ij} = \frac{d_i^\top D d_j}{\sqrt{d_i^\top D d_i}\sqrt{d_j^\top D d_j} + \epsilon_P}, \qquad d_i = \text{vec}(s_i - s_0), \quad P_{ii} = 1 \tag{14}$$

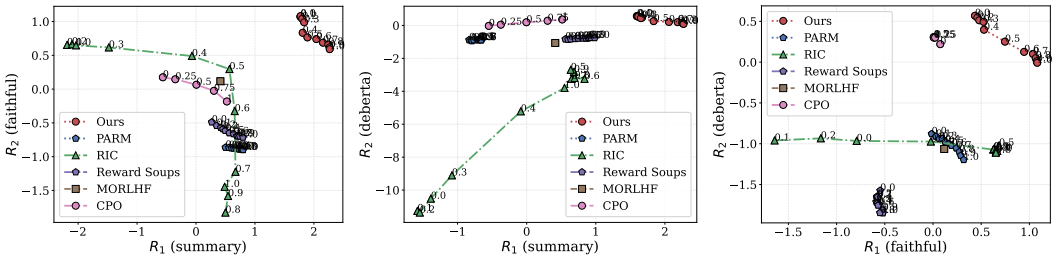
The off-diagonal matching loss is:

$$\mathcal{L}_{\text{Gram}} = \frac{2}{K(K-1)}\sum_{i<j}(P_{ij} - C_{ij})^2 \tag{15}$$

The main version uses unit weight for every pair. This continuous, pair-normalized objective uses the relation strength directly and requires neither a sign branch nor a manually chosen distance margin.

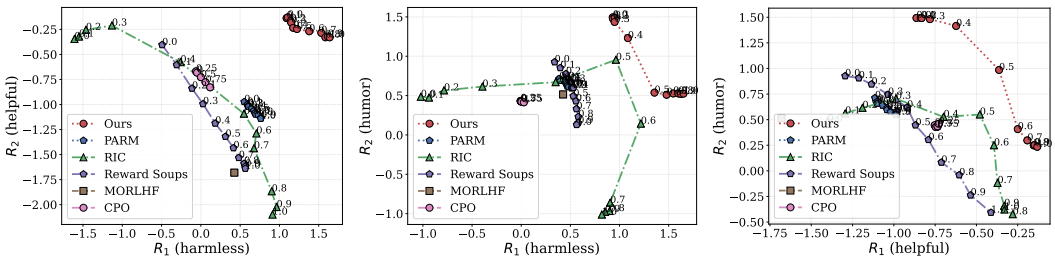
Here λ controls the contribution of Gram matching. Since P and C are cosine-like quantities, the loss is normalized by the number of objective pairs. The final loss is:

$$\mathcal{L}_{\text{Total}} = \mathcal{L}_{\text{GRPO}}(\theta, \{s_i\}) + \lambda\mathcal{L}_{\text{Gram}} \tag{16}$$



(a) "summary" and "faithful" (b) "summary" and "deberta" (c) "faithful" and "deberta"

Figure 2: Results of Reddit Summary



(a) "harmless" and "helpful" (b) "harmless" and "humor" (c) "helpful" and "humor"

Figure 3: Results of Helpful Assistant

4.2.4 Dynamic Relationship Tracking

We initialize the relation state without imposing a prior relationship, $C^{(0)} = I_K$, and update it only every T_C rollout phases. Given the batch estimate $\widehat{C}^{(t)}$ in Eq. 10, the detached affinity state follows:

$$C^{(t+1)} = (1 - \alpha_C)C^{(t)} + \alpha_C \widehat{C}^{(t)} \tag{17}$$

After each update, we symmetrize and clip C to $[-1, 1]$, reset its diagonal to one, and stop its gradient. The relation gradients, Fisher state, and affinity matrix are all detached before the policy optimizer step, so relation estimation does not introduce second-order gradients. This procedure allows the estimated objective interactions to track the evolving policy while keeping the Gram matching update first-order.

5 Experiments

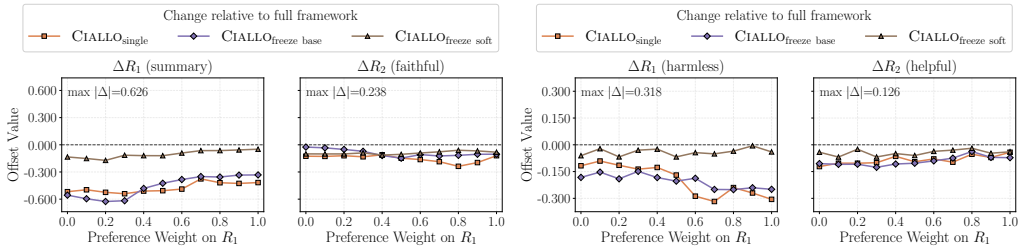
5.1 Experimental Setup

Datasets and Models. To comprehensively evaluate our proposed framework, we conduct experiments on two widely adopted multi-objective alignment benchmarks: Reddit Summary (Völske et al., 2017) and Helpful Assistant (Bai et al., 2022), which are described in Appendix C.

Baselines. We consider a representative set of baselines to evaluate the effectiveness of our method. These baselines are selected to cover different solution paradigms and provide a broad comparison with existing approaches. The detailed introduction of each baseline is provided in Appendix D.

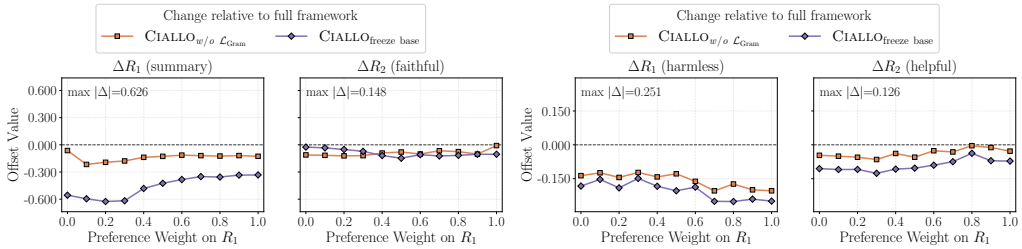
Evaluation Metrics. Following RIC, we uniformly sample 2,000 queries from the test set for both benchmarks. We evaluate the models by conditioning them on a diverse set of preference weight vectors to generate responses. The performance is quantified by calculating the mean reward scores across different dimensions, which are subsequently plotted to visualize the empirical Pareto frontiers.

For direct comparability, we follow the same evaluation protocol as Anonymous (2027) and Liu et al. (2025). Unless otherwise noted, all methods use the same backbone, reward models, filtered data, preference grid, evaluation queries, and decoding settings. Our detailed configurations of experiments are summarized in Appendix B.



(a) "summary" and "faithful" (b) "harmless" and "helpful"

Figure 4: Ablation study on the effectiveness of joint training.



(a) "summary" and "faithful" (b) "harmless" and "helpful"

Figure 5: Ablation study on the effectiveness of correlation modeling.

5.2 Main Results

Figures 2 and 3 present the empirical Pareto frontiers achieved by our method and the baselines on the Reddit Summary and Helpful Assistant benchmarks. Across all evaluated trade-off scenarios and preference configurations, our framework consistently approximates more favorable empirical Pareto frontiers than the baselines, indicating its effectiveness in balancing multiple alignment objectives across different preference configurations.

From the perspective of inter-objective relationship modeling, the fundamental limitations of existing methods become evident. Interpolation-based approaches, such as Reward Soups, exhibit relatively smooth transition curves. However, their strictly decoupled design isolates the optimization of different objectives, rendering them incapable of capturing the intrinsic synergies or conflicts between objectives, which ultimately bottlenecks their performance. Conversely, while prompt-conditioned approaches like RIC rely on the massive parameter scale of the LLM to implicitly perceive these relationships, this implicit mechanism is unreliable and insufficient, leading to highly unstable and sub-optimal trade-offs.

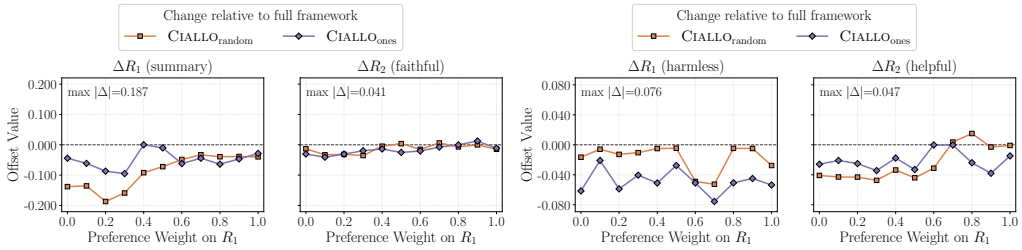
By contrast, our Correlation-aware Joint Training addresses these limitations. By matching the controller Gram matrix to the Fisher-affinity matrix, our method dynamically models the inter-objective relationships during GRPO optimization. This explicit guidance helps the model adapt to complex multi-objective trade-offs during training, empirically improving the trade-offs observed under our evaluation protocol.

5.3 Ablation Study

To better understand the contribution of each component in our proposed framework, we conduct ablation studies on both the Reddit Summary and Helpful Assistant benchmarks. We specifically investigate the effectiveness of the joint training paradigm and the explicit relationship modeling mechanism. The sensitivity analysis of the regularization weight is deferred to Appendix A.

5.3.1 Effectiveness of Joint Training

We first validate the necessity of the joint training stage. As illustrated in Fig. 4, we compare our full framework against three variants: (1) single, which only performs the single-objective



(a) "summary" and "faithful" (b) "harmless" and "helpful"

Figure 6: Ablation study on the correlation matrix C.

decomposition stage without any subsequent multi-objective training. (2) freeze base, which conducts multi-objective joint training of the soft prompts but keeps the backbone LLM frozen, and (3) freeze soft, which unfreezes the backbone LLM but freezes the soft prompts during the joint training. The results indicate that CIALLO achieves a significantly superior Pareto frontier compared to the single variant, demonstrating that single-objective training alone cannot effectively manage the complex trade-offs between competing objectives. Furthermore, CIALLO consistently outperforms the frozen variants. This highlights that merely updating the soft prompts or the backbone LLM is insufficient; unfreezing and jointly optimizing the LLM parameters is crucial for the model to fully internalize the multi-objective dynamics, and trainable soft prompts help capture more details.

5.3.2 Effectiveness of Correlation Modeling

Next, we evaluate the impact of explicitly modeling inter-objective correlation. Fig. 5 presents the comparison between the full framework and two ablated versions: (1) *w/o* $\mathcal{L}_{\text{Gram}}$, which removes the Fisher Gram-Affinity matching loss and relies entirely on the LLM's implicit modeling capabilities during stage 2, and (2) freeze base, which retains $\mathcal{L}_{\text{Gram}}$ but freezes the LLM. The noticeable performance drop in the *w/o* $\mathcal{L}_{\text{Gram}}$ variant confirms that implicit modeling alone struggles to distinguish and exploit different types of inter-objective relationships, indicating that explicitly guiding the controller geometry with the estimated affinity is useful in our evaluated settings. Additionally, the inferior performance of the freeze base variant in this context reveals that explicitly modeled relationships are difficult to capture and leverage without corresponding updates to the backbone LLM parameters.

5.3.3 Effectiveness of Correlation Matrix

To isolate the contribution of the dynamically tracked affinity matrix, we compare the full model with two fixed-matrix variants. (1) random samples three normalized random controller directions and uses their resulting symmetric Gram matrix, with unit diagonal, as a fixed affinity target throughout stage 2, while (2) ones uses the all-ones matrix throughout training. As shown in Fig. 6, both fixed alternatives underperform the original dynamic-C model on both benchmarks. This consistent degradation indicates that the dynamically updated matrix is not merely a generic regularizer: it effectively identifies target-relevant objective relationships and provides useful pairwise guidance during joint optimization.

6 Conclusion

In this paper, we propose CIALLO, a two-stage framework that first learns objective-specific soft-prompt controllers with a frozen backbone and then jointly optimizes the controllers and backbone under mixed preference weights. To expose interactions that may be obscured by preference scalarization, CIALLO estimates a dynamically updated Fisher-aware affinity matrix from objective-specific GRPO gradients. It then uses Fisher Gram matching to align the geometry of the prompt controllers with these local objective relationships. Experiments on two multi-objective alignment benchmarks show that CIALLO achieves more favorable empirical Pareto frontiers, demonstrating the value of explicitly incorporating inter-objective relationships into joint alignment optimization.

AI USE STATEMENT

During the preparation of this manuscript, we utilized LLMs solely for the purpose of language editing, grammatical correction, and academic phrasing refinement to improve readability. All core ideas, methodologies, experimental designs, and theoretical derivations presented in this paper were conceived and developed entirely by the human authors. The AI tools were not involved in any substantive scientific interpretation or intellectual creation.

REFERENCES

Anonymous. Meta-aligner: Bidirectional preference-policy optimization for multi-objective llms alignment. In *Submitted to International Conference on Learning Representations*, 2027. URL https://openreview.net/forum?id=6fQPzQKGT4. under review.

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.

Sébastien Bubeck, Varun Chandrasekaran, Ronen Eldan, Johannes Gehrke, Eric Horvitz, Ece Kamar, Peter Lee, Yin Tat Lee, Yuanzhi Li, Scott Lundberg, et al. Sparks of artificial general intelligence: Early experiments with gpt-4. *arXiv preprint arXiv:2303.12712*, 2023.

Kawin Ethayarajh, Winnie Xu, Niklas Muennighoff, Dan Jurafsky, and Douwe Kiela. Model alignment as prospect theoretic optimization. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 12634–12651, Vienna, Austria, 2024. PMLR.

Shangmin Guo, Biao Zhang, Tianlin Liu, Tianqi Liu, Misha Khalman, Felipe Llinares, Alexandre Rame, Thomas Mesnard, Yao Zhao, Bilal Piot, et al. Direct language model alignment from online AI feedback. *arXiv preprint arXiv:2402.04792*, 2024a.

Yiju Guo, Ganqu Cui, Lifan Yuan, Ning Ding, Zexu Sun, Bowen Sun, Huimin Chen, Ruobing Xie, Jie Zhou, Yankai Lin, Zhiyuan Liu, and Maosong Sun. Controllable preference optimization: Toward controllable multi-objective alignment. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 1437–1454, Miami, Florida, USA, November 2024b.

Jiwoo Hong, Noah Lee, and James Thorne. ORPO: Monolithic preference optimization without reference model. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 11170–11189, Miami, Florida, USA, November 2024. Association for Computational Linguistics.

Yue Huang, Lichao Sun, Haoran Wang, Siyuan Wu, Qihui Zhang, Yuan Li, Chujie Gao, Yixin Huang, Wenhan Lyu, Yixuan Zhang, et al. TrustLLM: Trustworthiness in large language models. *arXiv preprint arXiv:2401.05561*, 2024.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. Survey of hallucination in natural language generation. *ACM computing surveys*, 55(12):1–38, 2023.

Brian Lester, Rami Al-Rfou, and Noah Constant. The power of scale for parameter-efficient prompt tuning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 3045–3059, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics.

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

Zhuo Li, Guodong Du, Weiyang Guo, Yigeng Zhou, Xiucheng Li, Wenya Wang, Fangming Liu, Yequan Wang, Deheng Ye, Min Zhang, et al. Multi-objective large language model alignment with hierarchical experts. *arXiv preprint arXiv:2505.20925*, 2025.

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 *Proceedings of the 42nd International Conference on Machine Learning*, volume 267, pp. 37874–37888, Vancouver, BC, Canada, 13–19 Jul 2025. PMLR.

Biao Liu, Ning Xu, Junming Yang, and Xin Geng. Preference orchestrator: Prompt-aware multi-objective alignment for large language models. *arXiv preprint arXiv:2511.10656*, 2025.

Yu Meng, Mengzhou Xia, and Danqi Chen. SimPO: Simple preference optimization with a reference-free reward. In *Advances in Neural Information Processing Systems*, volume 37, pp. 124198–124235, 2024.

Subhojyoti Mukherjee, Anusha Lalitha, Sailik Sengupta, Aniket Deshmukh, and Branislav Kveton. Multi-objective alignment of large language models through hypervolume maximization. *arXiv preprint arXiv:2412.05469*, 2024.

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. In *Advances in Neural Information Processing Systems*, volume 35, pp. 27730–27744, 2022.

Qwen Team. Qwen3 technical report, 2025. URL https://arxiv.org/abs/2505.09388.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. In *Advances in Neural Information Processing Systems*, volume 36, pp. 53728–53741, 2023.

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. In *Advances in Neural Information Processing Systems*, volume 36, pp. 71095–71134, 2023.

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

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, et al. DeepSeekMath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

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

Frederick Tung and Greg Mori. Similarity-preserving knowledge distillation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 1365–1374, 2019.

Peter Vamplew, Richard Dazeley, Cameron Foale, Sally Firmin, and Jane Mummery. Human-aligned artificial intelligence is a multiobjective problem. *Ethics and Inf. Technol.*, 20(1):27–40, March 2018.

Michael Völske, Martin Potthast, Shahbaz Syed, and Benno Stein. TL;DR: Mining Reddit to learn automatic summarization. In *Proceedings of the Workshop on New Frontiers in Summarization*, pp. 59–63, Copenhagen, Denmark, 2017.

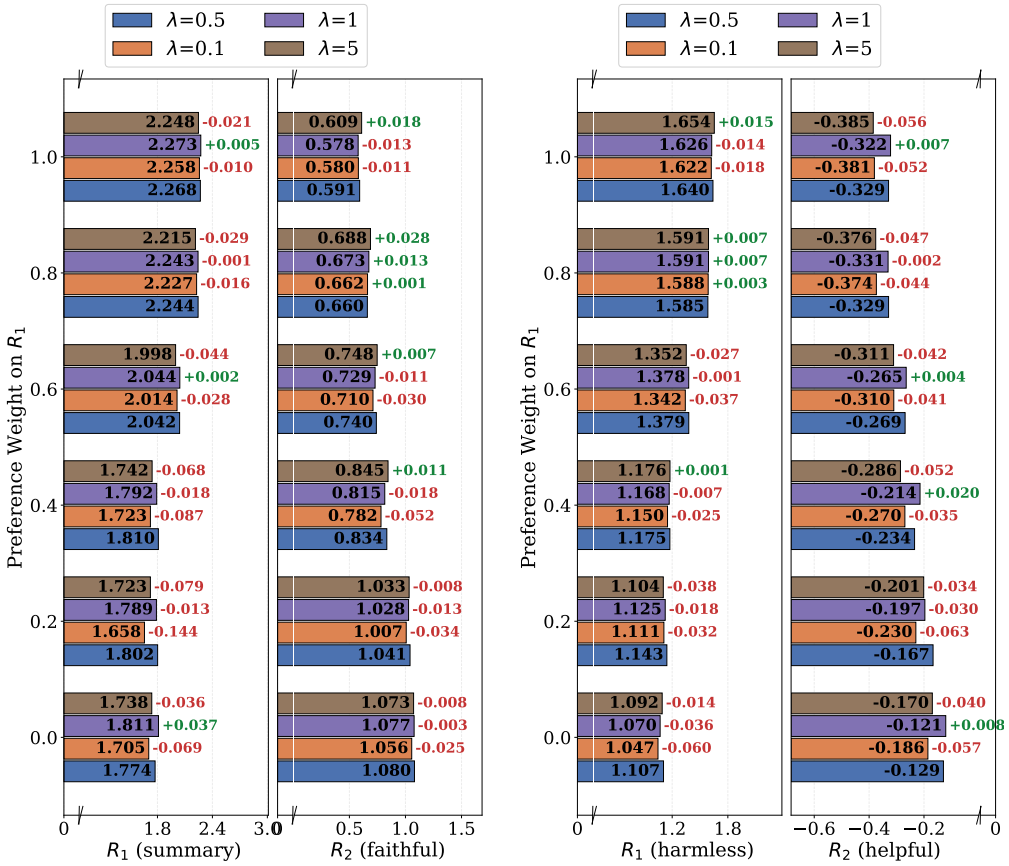
Haoxiang Wang, Yong Lin, Wei Xiong, Rui Yang, Shizhe Diao, Shuang Qiu, Han Zhao, and Tong Zhang. Arithmetic control of LLMs for diverse user preferences: Directional preference alignment with multi-objective rewards. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 8642–8655, Bangkok, Thailand, August 2024.

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 *Proceedings of the 41st International Conference on Machine Learning*, volume 235, pp. 56276–56297, Vienna, Austria, 2024.

Tianhe Yu, Saurabh Kumar, Abhishek Gupta, Sergey Levine, Karol Hausman, and Chelsea Finn. Gradient surgery for multi-task learning. *Advances in neural information processing systems*, 33: 5824–5836, 2020.

Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, et al. A survey of large language models. *arXiv preprint arXiv:2303.18223*, 2023.

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*, pp. 10586–10613, Bangkok, Thailand, 2024.



(a) "summary" and "faithful"

(b) "harmless" and "helpful"

Figure 7: Sensitivity to the Gram-matching weight λ.

A Hyperparameter Sensitivity Analysis

We vary the Gram-matching weight λ over $\{0.1, 0.5, 1, 5\}$ on both the Reddit Summary and Helpful Assistant benchmarks. As shown in Fig. 7, both $\lambda = 0.1$ and $\lambda = 5$ cause a clear degradation in performance, reflecting under- and over-weighting of the correlation constraint, respectively. Although $\lambda = 1$ occasionally exceeds $\lambda = 0.5$ at individual preference settings, its overall performance is lower. We therefore recommend choosing λ between 0.5 and 1.

B Detailed Experimental Settings

Our training configuration follows the two-stage structure described in table 1. During the first stage, the backbone model is frozen and only the objective-specific soft prompts are optimized. The resulting soft prompts are then used to initialize the objective-specific controllers for joint training.

During the second stage, both the backbone model and the soft prompts are updated. Unless otherwise specified, the same optimization settings are used for both benchmarks; only the base model and the maximum generation length differ according to task complexity.

Stage	Hyperparameter	Value
Common	Soft prompt length (L)	20
	Batch size	2
Stage 1	Training samples	20,000
	Training epochs	2
	Prompt learning rate	1×10^{-2}
	β_{DPO}	0.1
Stage 2	Training epochs	1
	Mini batch size	1
	Gradient accumulation steps	16
	Relation update interval	16
	Dirichlet α_D	0.8
	Group size (G)	8
	LLM learning rate	1×10^{-5}
	Prompt learning rate	5×10^{-5}
	KL weight (β)	0.05
	Gram-matching weight λ	0.5
	EMA update rate (α_C)	0.05
	Clipping epsilon (ϵ)	0.2
	Fisher damping (ϵ_F)	1×10^{-6}
	Affinity and Gram normalization (ϵ_C, ϵ_P)	1×10^{-8}
Generate	Temperature	0.7
	Top-p	0.9
	Max new tokens (Reddit Summary)	64
	Max new tokens (Helpful Assistant)	128
	Random seed	42

Table 1: Detailed hyperparameter settings for both training stages.

C Introduction of Datasets and Models

`Reddit Summary` This task involves generating high-quality summaries for Reddit posts. We employ Qwen3-0.6B (Qwen Team, 2025) as our base language model. Following prior evaluation protocols, the generated responses are assessed across three distinct reward dimensions: *summary* (general summarization quality), *deberta* (alternative quality assessment based on a different corpus), and *faithful* (factual consistency with the original text).

`Helpful Assistant` This benchmark evaluates the model's ability to engage in open-ended, multi-turn dialogues using prompts from the HH-RLHF dataset. For this more complex task, we scale our base policy to Qwen3-4B (Qwen Team, 2025). The multi-objective performance is measured using three reward models: *harmless*, *helpful*, and *humor*.

D Details of Baselines

MORLHF(Li et al., 2020). A standard reinforcement learning approach that optimizes the model using a fixed, scalarized sum of multiple reward signals. We do not include every MORLHF result

because of the high computational cost and low test-time flexibility. It is considered the model's baseline performance following RLHF.

REWARD SOUPS(Rame et al., 2023). A parameter-space interpolation method that merges the weights of multiple expert models, each independently fine-tuned via PPO for a specific objective.

RIC(Yang et al., 2024). A test-time adaptation strategy that appends objective reward scores directly into the input context to guide generation via supervised fine-tuning.

PARM(Lin et al., 2025). A dynamic decoding approach that trains a preference-aware autoregressive reward model to adjust output logits at test time without retraining the backbone LLM.

CPO(Guo et al., 2024b). An alignment approach that trains an LLM with explicit multi-objective preference tokens through SFT and DPO, enabling controllable generation at inference time. We followed the official configuration and used the same five-level setup.

It is worth noting that all reward models utilized in our experiments are directly adopted from Yang et al. (2024).

E ANALYSIS OF FIRST-STAGE OPTIMIZATION METHODS

We compare different optimization methods for the first-stage objective-specific prompt learning under two extreme preference settings, i.e., $[1, 0]$ for emphasizing "summary" and $[0, 1]$ for emphasizing "faithful". As shown in Table 2, SFT, PPO, and DPO exhibit an objective reversal phenomenon: when "faithful" is assigned the maximum weight, "summary" obtains the higher score, while when "summary" is assigned the maximum weight, "faithful" obtains the higher score. Although they are able to change the model's performance, they do not always preserve the intended direction of the preference vector. In contrast, Online DPO produces scores that are consistent with the specified preference direction. This observation motivates our use of Online DPO in the first stage.

Method	$[0, 1]$		$[1, 0]$	
	summary ↓	faithful ↑	summary ↑	faithful ↓
SFT	-0.496	0.232	-0.352	0.238
PPO	1.285	0.713	-0.145	0.478
DPO	1.194	0.492	0.926	0.551
Online DPO	1.257	0.955	1.852	0.471

Table 2: First-stage optimization comparison under extreme preference settings.

F THEORETICAL ANALYSIS OF FISHER-AWARE RELATIONSHIP MODELING

This appendix provides a local and conditional justification for the Fisher-aware relationship regularizer. We address two questions: what information about objective interactions can be lost when only scalarized gradients are observed, and under what idealized conditions Fisher Gram matching is geometrically consistent with natural-gradient transfer.

The results below do not constitute a global convergence guarantee. In particular, we distinguish the exact Fisher matrix used in the population analysis from the damped diagonal estimate used by the practical algorithm.

F.1 FISHER AFFINITY AS LOCAL CROSS-OBJECTIVE TRANSFER

Fix the current policy parameters, context distribution, and mixed soft-prompt activation. Vectorize the activation as $z \in \mathbb{R}^p$, where $p = Ld$, and let $\ell_i(z)$ denote the local surrogate loss associated with

objective i. Define

$$g_i = \nabla_z \ell_i(z),$$

and assume that the gradients being compared are nonzero.

The exact Fisher matrix in the mixed-prompt space is:

$$F(z) = \mathbb{E}_{Y \sim \pi_z} \left[\nabla_z \log \pi_z(Y) \nabla_z \log \pi_z(Y)^\top \right]$$

where the expectation may additionally include the current context distribution. Under standard differentiability and regularity conditions,

$$\mathrm{KL}\big(\pi_z \,\|\, \pi_{z+\delta}\big) = \frac{1}{2} \delta^\top F(z) \delta + o(\|\delta\|_2^2) \tag{18}$$

The Fisher matrix is generally positive semidefinite and may be singular. We therefore let $M \succ 0$ denote a fixed local metric. When $F(z)$ is nonsingular, one may take $M = F(z)$. Otherwise, M may be a regularized metric such as $F(z) + \gamma I$, or a positive-definite approximation. In the latter cases, M defines a regularized Fisher geometry rather than being exactly the Hessian in Eq. 18.

Proposition 1 (local transfer under a metric-preconditioned step). Consider the local step for objective i:

$$\delta_i = -\eta M^{-1} g_i, \qquad \eta > 0 \tag{19}$$

If ℓ_j has a locally Lipschitz gradient and M is held fixed over the step, then:

$$\ell_j(z + \delta_i) - \ell_j(z) = -\eta g_j^\top M^{-1} g_i + O(\eta^2) \tag{20}$$

Proof. A Taylor expansion of ℓ_j at z gives:

$$\ell_j(z + \delta_i) - \ell_j(z) = g_j^\top \delta_i + O(\|\delta_i\|_2^2)$$

Because M and the local gradients are fixed, $\|\delta_i\|_2 = O(\eta)$. Substituting Eq. 19 proves Eq. 20. □

Define the metric-normalized affinity:

$$C_{ij}^M = \frac{g_i^\top M^{-1} g_j}{\sqrt{g_i^\top M^{-1} g_i} \sqrt{g_j^\top M^{-1} g_j}} \tag{21}$$

Equation 20 can equivalently be written as:

$$\ell_j(z + \delta_i) - \ell_j(z) = -\eta C_{ij}^M \|g_i\|_{M^{-1}} \|g_j\|_{M^{-1}} + O(\eta^2)$$

Thus, $C_{ij}^M > 0$ indicates beneficial first-order transfer under the step in Eq. 19, whereas $C_{ij}^M < 0$ indicates interference. Its magnitude measures normalized directional alignment; the absolute transfer magnitude also depends on the two gradient norms and the step size.

When $M = F(z)$, Eq. 19 is the local natural gradient step. When M is a diagonal or damped approximation, the same transfer identity remains valid for the corresponding preconditioned step, but it is only an approximation to exact Fisher-natural-gradient transfer.

Proposition 2 (coordinate covariance of the full metric). Let $z' = Az$ for an invertible matrix A, and transform

$$g_i' = A^{-\top} g_i, \qquad M' = A^{-\top} M A^{-1}$$

Then:

$$C_{ij}^{M'}(g_i', g_j') = C_{ij}^M(g_i, g_j)$$

Likewise, if controller directions transform as $d_i' = A d_i$, then

$$(d_i')^\top M' d_j' = d_i^\top M d_j \tag{22}$$

Proof. Since

$$(M')^{-1} = AM^{-1}A^\top, \qquad (g_i')^\top (M')^{-1} g_j' = g_i^\top M^{-1} g_j.$$

The controller identity follows by direct substitution. Applying the same identities to the normalization terms proves the affinity result. □

For the exact Fisher matrix, the transformation of M in Proposition 2 follows from the transformation law of score gradients. A diagonal Fisher approximation is not covariant under a general mixing of coordinates. It retains covariance only for restricted transformations such as coordinate permutations and coordinate-wise rescalings, subject to the treatment of the damping term. Therefore, Proposition 2 explains the population geometry but should not be interpreted as full reparameterization invariance of the implemented diagonal estimator.

F.2 Conditional Consistency of Fisher Gram Matching

Let s_0 be the common prompt initialization and define the nonzero, anchor-centered controller direction:

$$d_i = \mathrm{vec}(s_i - s_0)$$

For a positive-definite local metric M, define the controller Gram matrix:

$$P_{ij}^M = \frac{d_i^\top M d_j}{\sqrt{d_i^\top M d_i}\sqrt{d_j^\top M d_j}} \tag{23}$$

Centering at s_0 removes the shared initialization, so that d_i represents the finite controller displacement associated with objective i.

Proposition 3 (consistency under ideal natural-gradient controllers). Suppose that, at the same local point and under the same metric M, an ideal controller direction for objective i satisfies:

$$d_i^\star = -c_i M^{-1} g_i, \qquad c_i > 0 \tag{24}$$

Then:

$$P_{ij}^M(d_i^\star, d_j^\star) = C_{ij}^M(g_i, g_j) \tag{25}$$

Proof. Equation 24 gives:

$$(d_i^\star)^\top M d_j^\star = c_i c_j g_i^\top M^{-1} g_j$$

and

$$(d_i^\star)^\top M d_i^\star = c_i^2 g_i^\top M^{-1} g_i$$

The positive scale factors cancel after normalization, proving Eq. 25. □

To clarify the geometric meaning, define the normalized whitened directions:

$$v_i = \frac{M^{1/2} d_i}{\sqrt{d_i^\top M d_i}}, \qquad u_i = \frac{-M^{-1/2} g_i}{\sqrt{g_i^\top M^{-1} g_i}}$$

Then

$$P_{ij}^M = v_i^\top v_j, \qquad C_{ij}^M = u_i^\top u_j$$

If the complete matrices satisfy $P^M = C^M$, equality of their Gram matrices implies that the two indexed sets of normalized directions agree up to a common isometry on their spans. Gram matching therefore constrains pairwise geometry without prescribing the scale or absolute orientation of each controller.

Proposition 3 is a conditional consistency result, not a recovery or convergence theorem. The learned displacement $s_i - s_0$ is a finite direction produced by prompt and policy optimization; it is not assumed by the algorithm to equal the instantaneous natural gradient. Moreover, the practical target uses finite-sample gradients, a diagonal metric, and a temporally averaged affinity matrix. Consequently, the proposition motivates the form of the regularizer but does not imply that its minimum is attained or that minimizing it alone improves the policy objective.

F.3 Pointwise Non-Identifiability of Scalarized Gradients

We next characterize what can be recovered from scalarized gradients at a fixed local point. Fix (θ, z) and either a common rollout group or the corresponding population expectation. At the rollout policy, where the probability ratio is one and clipping is locally inactive, the surrogate gradient associated with a preference vector $\boldsymbol{w}$ is linear in the objective-specific gradients:

$$g(\boldsymbol{w}) = \sum_{i=1}^{K} w_i g_i \tag{26}$$

Suppose that N counterfactual preference vectors are evaluated at this same local point. Let $\Gamma \in \mathbb{R}^{K\times p}$ contain $g_i^\top$ as its rows, and let $W \in \mathbb{R}^{N\times K}$ contain the preference vectors. The observed scalarized gradients are:

$$Q = W\Gamma \tag{27}$$

Proposition 4 (pointwise non-identifiability under rank-deficient coverage). Assume $K \geq 2$. If $\mathrm{rank}(W) < K$, the mapping $\Gamma \mapsto W\Gamma$ is not injective. In particular, the observations Q do not identify the unnormalized Fisher cross-objective matrix:

$$G(\Gamma) = \Gamma M^{-1}\Gamma^\top$$

For generic nonzero gradients and perturbations, they also do not identify the normalized affinity matrix in Eq. 21.

Proof. Choose a nonzero $a \in \mathrm{null}(W)$ and a nonzero $v \in \mathbb{R}^p$, and define:

$$\widetilde{\Gamma} = \Gamma + av^\top$$

Then:

$$W\widetilde{\Gamma} = W\Gamma + Wav^\top = W\Gamma$$

However,

$$\begin{aligned}\widetilde{\Gamma}M^{-1}\widetilde{\Gamma}^\top = \Gamma M^{-1}\Gamma^\top + \Gamma M^{-1}va^\top \\ + av^\top M^{-1}\Gamma^\top + a(v^\top M^{-1}v)a^\top\end{aligned} \tag{28}$$

which differs from $\Gamma M^{-1}\Gamma^\top$ except when the gradients and perturbation satisfy special cancellation conditions. Normalization does not generally remove this change: equality of every normalized pairwise entry imposes additional exceptional constraints on v. Therefore, identical scalarized observations can correspond to different local objective geometries. □

For example, with two nonzero objectives satisfying $g_2 = -g_1$ and $\boldsymbol{w} = (1/2, 1/2)$:

$$g(\boldsymbol{w}) = 0, \qquad C_{12}^M = -1$$

The scalarized update at this preference contains no first-order signal, although the two objective gradients exhibit maximal local conflict. Evaluating the two reward heads separately on the same rollout group retains both gradients and exposes this cancellation.

The fixed-point qualification is essential. In the actual algorithm, $z = s_{\mathrm{mix}}(\boldsymbol{w})$, so gradients collected at different preferences generally satisfy:

$$g_i = g_i\big(z(\boldsymbol{w})\big)$$

They therefore cannot in general be pooled into one global factorization $Q = W\Gamma$. Proposition 4 should instead be read pointwise: at a particular rollout location, observing only one or an otherwise rank-deficient collection of scalarized gradients does not determine the objective-specific geometry at that location.

Conversely, if K linearly independent preferences are evaluated at the same local point, the gradients are observed without noise, and W has full column rank, then Γ can be recovered from Q. The proposition does not claim that every implicit method fails; it isolates information that a single scalarized update may discard and that separate reward-head evaluation makes directly available.

F.4 Conditional Limitations of Post-Hoc Relationship Handling

The preceding results concern local gradient information. We next describe two distinct, conditional limitations of post-hoc strategies. Endpoint parameter interpolation can be inaccurate when the optimal solution path has uniformly one-sided curvature in a locally strongly convex region. Reward-conditioned generation can be unidentified at desired reward vectors outside its observed support. Neither result is an unconditional failure theorem for the corresponding method class.

F.4.1 Endpoint Interpolation and Curved Solution Paths

Consider two objectives with preference $w = (\tau, 1-\tau)$, where $\tau \in [0,1]$. Let $\ell_1(\theta)$ and $\ell_2(\theta)$ be regularized population policy losses, and define:

$$\begin{aligned} \ell_\tau(\theta) &= \tau\ell_1(\theta) + (1-\tau)\ell_2(\theta), \\ \theta^\star(\tau) &= \arg\min_\theta \ell_\tau(\theta) \end{aligned} \tag{29}$$

This scalarization is an analyzable abstraction; the clipped and group-normalized GRPO objective need not equal it exactly.

An endpoint interpolation method constructs:

$$\overline{\theta}(\tau) = (1-\tau)\theta^\star(0) + \tau\theta^\star(1) \tag{30}$$

Proposition 5 (strict interpolation error under uniformly signed curvature). Suppose that $\theta^\star(\tau)$ is twice continuously differentiable. For each τ, suppose $\theta^\star(\tau)$ is an unconstrained stationary minimizer of ℓ_τ, and that ℓ_τ is μ_θ-strongly convex on a convex neighborhood containing the line segment between $\theta^\star(\tau)$ and $\overline{\theta}(\tau)$. Suppose further that there exist a unit vector v and $\kappa > 0$ such that

$$v^\top \frac{\mathrm{d}^2\theta^\star(s)}{\mathrm{d}s^2} \geq \kappa, \qquad s \in [0,1] \tag{31}$$

Then, for every $\tau \in (0,1)$,

$$\left\|\overline{\theta}(\tau) - \theta^\star(\tau)\right\|_2 \geq \frac{\kappa}{2}\tau(1-\tau), \tag{32}$$

and

$$\ell_\tau(\overline{\theta}(\tau)) - \ell_\tau(\theta^\star(\tau)) \geq \frac{\mu_\theta\kappa^2}{8}\tau^2(1-\tau)^2 > 0 \tag{33}$$

Proof. Let $f(s) = v^\top\theta^\star(s)$. The curvature assumption gives $f''(s) \geq \kappa$. Applying the linear-interpolation error bound to f, or equivalently using the convexity of $f(s) - \kappa s^2/2$, yields

$$(1-\tau)f(0) + \tau f(1) - f(\tau) \geq \frac{\kappa}{2}\tau(1-\tau).$$

The left-hand side equals

$$v^\top\big(\overline{\theta}(\tau) - \theta^\star(\tau)\big).$$

Since $\|v\|_2 = 1$, Cauchy–Schwarz proves Eq. 32.

Strong convexity and $\nabla\ell_\tau(\theta^\star(\tau)) = 0$ imply

$$\ell_\tau(\overline{\theta}(\tau)) - \ell_\tau(\theta^\star(\tau)) \geq \frac{\mu_\theta}{2}\left\|\overline{\theta}(\tau) - \theta^\star(\tau)\right\|_2^2.$$

Substituting Eq. 32 proves Eq. 33. □

The curvature assumption can arise when the objectives have different local Hessians or when the policy response is nonlinear. To see this, additionally assume that ℓ_1 and ℓ_2 are three times continuously differentiable and that

$$H_\tau = \nabla^2\ell_\tau(\theta^\star(\tau))$$

is invertible. Define

$$\Delta H_\tau = \nabla^2\ell_1(\theta^\star(\tau)) - \nabla^2\ell_2(\theta^\star(\tau)),$$

and

$$T_\tau = \nabla^3 \ell_\tau(\theta^\star(\tau)).$$

Differentiating $\nabla \ell_\tau(\theta^\star(\tau)) = 0$ gives:

$$\frac{\mathrm{d}\theta^\star}{\mathrm{d}\tau} = -H_\tau^{-1} \left[\nabla \ell_1(\theta^\star(\tau)) - \nabla \ell_2(\theta^\star(\tau))\right], \tag{34}$$

and

$$\frac{\mathrm{d}^2\theta^\star}{\mathrm{d}\tau^2} = -H_\tau^{-1} \left(2\Delta H_\tau \frac{\mathrm{d}\theta^\star}{\mathrm{d}\tau} + T_\tau \left[\frac{\mathrm{d}\theta^\star}{\mathrm{d}\tau}, \frac{\mathrm{d}\theta^\star}{\mathrm{d}\tau}\right] \right) \tag{35}$$

These identities show possible sources of curvature, but they do not imply the uniformly signed lower bound in Eq. 31. Proposition 5 therefore identifies a specific regime in which endpoint parameter interpolation, such as Reward Soups, has a strict population approximation error. It does not assert that every nonlinear or curved solution path produces such an error.

The second stage of CIALLO is trained on interior preference vectors rather than constructing every interior policy solely from endpoint parameters. This removes the particular endpoint-only restriction analyzed above, but it does not imply that a shared finite-capacity model realizes every separate optimum $\theta^\star(\tau)$.

F.4.2 Reward Conditioning Outside the Observed Support

Reward-conditioned generation learns a conditional map from a desired reward vector to a response. Let $\mathcal{D}$ be the training distribution over (X, Y). For $\mathcal{D}_X$-almost every query x, define the conditional reward support:

$$\mathcal{S}_x = \mathrm{supp}\big(r(x, Y) \mid X = x\big) \subseteq \mathbb{R}^K \tag{36}$$

Let $\mathcal{D}_{X,R}$ be the pushforward of $\mathcal{D}$ under $(X, Y) \mapsto (X, r(X, Y))$. The population conditional SFT objective is:

$$\mathcal{L}_{\mathrm{cond}}(q) = -\mathbb{E}_{(x,y)\sim\mathcal{D}} \log q\big(y \mid x, r(x, y)\big) \tag{37}$$

Proposition 6 (off-support non-identifiability in a sufficiently rich policy class). Let q_1 and q_2 be two conditional policies in the considered hypothesis class satisfying:

$$q_1(y \mid x, R) = q_2(y \mid x, R) \tag{38}$$

for $\mathcal{D}_{X,R}$-almost every (x, R) and all responses y. Then

$$\mathcal{L}_{\mathrm{cond}}(q_1) = \mathcal{L}_{\mathrm{cond}}(q_2).$$

Consequently, if the hypothesis class contains two such policies that differ at an off-support target, the conditional SFT objective alone does not identify the generated policy at that target.

Proof. Every pair $(x, r(x, y))$ appearing in Eq. 37 lies in the support of $\mathcal{D}_{X,R}$ almost surely. Equation 38 therefore makes the two integrands equal almost surely. Values outside this support have zero probability under the training objective and cannot change its value. □

The qualification on the hypothesis class is necessary. Parameter sharing, continuity, or other architectural inductive biases may select a particular off-support extension. Such an extension is determined by those restrictions rather than by direct supervision at the off-support target.

Exact dependence between reward dimensions makes this issue especially transparent. Let

$$\Sigma_r(x) = \mathrm{Cov}(r(x, Y) \mid X = x).$$

If $\Sigma_r(x)$ is singular, there exists a nonzero vector v_x such that

$$\mathrm{Var}\big(v_x^\top r(x, Y) \mid X = x\big) = 0.$$

Hence, for some $b(x)$:

$$\mathcal{S}_x \subseteq \big\{R \in \mathbb{R}^K : v_x^\top R = b(x)\big\} \tag{39}$$

A desired reward vector violating this equality is necessarily off-support. Near-singular covariance does not imply exact non-identifiability, but it concentrates observations in a thin region and can make finite-sample extrapolation poorly conditioned.

This describes a regime in which a reward-conditioned method such as RIC may be sensitive to its inductive bias outside the observed reward support. CIALLO instead conditions on a preference weight w and evaluates sampled responses through $R_{\text{total}} = w^\top r$. This changes the control variable and avoids requiring the policy to invert an arbitrary desired reward vector. It does not make an infeasible reward vector attainable and does not establish unconditional superiority over reward conditioning.

F.4.3 PREFERENCE-AWARE METHODS SUCH AS PARM

Preference-aware methods do not necessarily share the endpoint-only limitation in Proposition 5. When they condition on preference weights rather than desired reward outcomes, they also do not necessarily face the specific inverse-support problem in Proposition 6. A sufficiently expressive method such as PARM may encode objective interactions in its reward model and decoding process.

The distinction is therefore about which information is made explicit at a given local update. Proposition 4 shows that a rank-deficient set of scalarized gradients evaluated at one local point does not determine the objective-specific geometry at that point. A preference-aware architecture may nevertheless infer useful structure across locations when coverage, capacity, and regularity are sufficient.

Our method explicitly evaluates every reward head on the same response group, constructs objective-specific surrogate gradients at the same mixed-prompt activation, and regularizes the controller geometry using their estimated Fisher affinity. This is a mechanism-level distinction, not an architecture-specific lower bound or an unconditional failure theorem for PARM.

F.5 IMPLICATIONS AND SCOPE

The results support the proposed construction in the following limited sense. Proposition 1 gives Fisher-gradient affinity a local cross-objective transfer interpretation under a metric-preconditioned step. Proposition 2 establishes coordinate covariance for the full metric. Proposition 3 shows that Fisher Gram matching is exactly consistent with ideal local natural-gradient controller directions. Proposition 4 identifies information that pointwise scalarization can discard. Propositions 5 and 6 characterize separate conditional limitations of endpoint interpolation and off-support reward conditioning.

Several gaps remain between this population analysis and the implemented algorithm. The practical method uses a finite-sample diagonal Fisher estimate rather than the full matrix, aggregates noisy objective gradients, and tracks affinity with an exponential moving average. Controller directions are finite learned displacements rather than instantaneous natural-gradient steps. Moreover, the backbone and prompts change during training, so the local metric and gradients are nonstationary.

Accordingly, the analysis does not establish global convergence, attainment of zero Gram-matching loss, recovery of a unique objective relationship, or uniform superiority over implicit and post-hoc methods. It instead shows that the proposed regularizer targets a well-defined local geometry and makes objective-specific information available that a single scalarized update need not retain.

G TRAINING TIME, TRAINABLE PARAMETERS, AND MODEL COUNT

This section reports the training cost of the Reddit Summary experiment with Qwen3-0.6B on four RTX 4090 GPUs. Unless stated otherwise, "training time" means wall-clock time for the complete training stage, including any method-required rollout or preference-data construction, but excluding final test-set generation. RIC performs 1 offline SFT run and 2 online runs, PARM trains 2 epochs, Reward Soups trains 1 full policy expert per objective before parameter interpolation, and CPO trains 3 CPSFT epochs and 3 CDPO epochs.

Table 3: Summary of training time, trainable parameters, and all components.

Method	**Time**	**Trainable Parameters**	**Models/Components**
CIALLO	≈ 9 h $(2+7)$	20,480 (Stage 1); 596,111,360 (Stage 2)	1 policy +3 SPs
RIC	≈ 8 h $(5+1.5\times 2)$	596,049,920	1 policy
PARM	≈ 2 h	1,611,008	1 policy +1 ARM
REWARD SOUPS	≈ 20 h	596,049,920 per expert	3 PPO experts
CPO	≈ 10 h	596,111,360	1 policy

H MACRO-AVERAGED REWARD GAIN

To summarize performance across objective pairs, we compute the macro-average over the six reward entries reported for each benchmark. Specifically, the gain of CIALLO over a baseline is defined as:

$$\Delta_{\text{macro}} = \frac{1}{6}\sum_{m=1}^{6}\left(R_m^{\text{CIALLO}} - R_m^{\text{baseline}}\right) \tag{40}$$

where each R_m is the mean normalized reward over all evaluated preference weights for one objective in one objective pair.

Table 4: Mean obtained scores on the Helpful Assistant.

	Harmless–Helpful		Harmless–Humor		Helpful–Humor	
Method	Harmless	Helpful	Harmless	Humor	Helpful	Humor
Ours	1.3310	-0.2391	1.2824	0.9338	-0.4704	0.8909
PARM	0.6644	-1.0610	0.4577	0.6501	-1.0486	0.6203
RIC	-0.1184	-1.0472	0.2607	0.0082	-0.7056	0.2211
Reward Soups	0.1994	-1.1966	0.5090	0.5090	-0.8655	0.3811
CPO	0.1199	-0.7368	0.0034	0.4246	-0.7400	0.4397

Table 5: Mean obtained scores on the Reddit Summary.

	Faithful–Deberta		Summary–Deberta		Summary–Faithful	
Method	Faithful	Deberta	Summary	Deberta	Summary	Faithful
Ours	0.7620	0.2771	1.9085	0.3310	1.9871	0.8252
PARM	0.1620	-1.0150	-0.7213	-0.8828	0.6859	-0.8832
RIC	0.0858	-1.0325	-0.1484	-6.0446	-0.4051	-0.2765
Reward Soups	-0.5611	-1.7263	0.7835	-0.7903	0.5578	-0.6291
CPO	-0.0157	0.0356	-0.0219	0.1732	0.0255	0.2870

I LLM AS JUDGE

Reward-model scores alone may overestimate the quality of an aligned model when the model exploits artifacts or blind spots of the reward models used for optimization. Such behavior, commonly referred to as reward hacking, can make the conclusions appear stronger without corresponding improvements in the actual responses. To reduce the influence of reward hacking on our conclusions, we additionally conduct an independent output-level pairwise evaluation using an LLM judge.

We evaluate the responses generated by CIALLO against PARM, RIC, and Reward Soups, on the Helpful Assistant benchmark. We consider the three objective pairs *harmless–humor*, *helpful–humor*, and *harmless–helpful*. We use DeepSeek-V4.1-Flash to evaluate 200 examples.

For a given input and preference vector, the judge receives the original conversation and the two generated responses, but does not receive the reward-model scores. The positions of the two responses are randomly permuted to reduce position bias. The judge returns only a winner label (*A*, *B*, or *tie*), and a tie contributes one half of a win. The resulting pairwise win rate is computed as

$$\widehat{\mathrm{WR}} = \frac{W + \frac{1}{2}T}{W + L + T}, \tag{41}$$

where W, L, and T denote the numbers of wins, losses, and ties, respectively.

Table 6: Estimated pairwise win rates of CIALLO on the *harmless–humor* objective pair.

Preference	CIALLO vs. PARM	CIALLO vs. RIC	CIALLO vs. REWARD SOUPS
$[0.0, 1.0]$	0.882	0.805	0.855
$[0.2, 0.8]$	0.865	0.890	0.890
$[0.4, 0.6]$	0.820	0.905	0.950
$[0.6, 0.4]$	0.890	0.980	0.965
$[0.8, 0.2]$	0.882	0.990	0.980
$[1.0, 0.0]$	0.845	0.975	0.985

Table 7: Estimated pairwise win rates of CIALLO on the *helpful–humor* objective pair.

Preference	CIALLO vs. PARM	CIALLO vs. RIC	CIALLO vs. REWARD SOUPS
$[0.0, 1.0]$	0.863	0.755	0.815
$[0.2, 0.8]$	0.877	0.885	0.880
$[0.4, 0.6]$	0.865	0.980	0.945
$[0.6, 0.4]$	0.870	0.980	0.973
$[0.8, 0.2]$	0.900	0.990	0.980
$[1.0, 0.0]$	0.845	0.990	0.980

Table 8: Estimated pairwise win rates of CIALLO on the *harmless–helpful* objective pair.

Preference	CIALLO vs. PARM	CIALLO vs. RIC	CIALLO vs. REWARD SOUPS
$[0.0, 1.0]$	0.855	0.995	0.995
$[0.2, 0.8]$	0.897	0.995	0.975
$[0.4, 0.6]$	0.863	0.975	0.965
$[0.6, 0.4]$	0.870	0.945	0.985
$[0.8, 0.2]$	0.853	0.970	0.995
$[1.0, 0.0]$	0.860	0.985	0.995