

# Self-Guided Alignment: Adaptive Preference Sensing for Multi-Objective Generation

Ning Wang<sup>1</sup>, Zhanyang Liu<sup>1</sup>, Taotao Zhou<sup>1</sup>, Xinrui Zhang<sup>2</sup>,  
Zongru Shao<sup>3</sup>, Haojie Zhou<sup>\*1</sup>,

<sup>1</sup>School of Artificial Intelligence and Computer Science, Jiangnan University, Wuxi, China

<sup>2</sup>School of Business, Jiangnan University, Wuxi, China

<sup>3</sup>Silicon Austria Labs, Linz, Austria

zhouhaojie@jiangnan.edu.cn

## Abstract

Aligning Large Language Models (LLMs) with diverse and potentially conflicting human values necessitates navigating complex multi-objective landscapes. However, existing prompt-conditioned approaches face a critical training-inference discrepancy: they rely on ground-truth scores during training while requiring manual user-specification at inference. We introduce prediction of implicit preferences to bridge this gap while reducing user burden. To this end, we propose Self-Guided Alignment (SGA), a framework that transforms passive reward dependency into an intrinsic adaptive sensing capability. It employs a dual-head architecture to unify preference internalization with conditional generation, enabling the model to learn a latent mapping between raw prompts and preference profiles. Through adaptive preference sensing, the model autonomously predicts the latent preference score to self-guide the generation, thereby eliminating the need for manual specification at inference. Extensive experiments across diverse model scales demonstrate that SGA often outperforms state-of-the-art baselines, achieving superior multi-objective trade-offs and improved preference alignment. Code is available at <https://github.com/python-yyds/SGA>.

## 1 Introduction

Large Language Models (LLMs) (OpenAI et al., 2025; Yang et al., 2025) have demonstrated remarkable capabilities across diverse applications, ranging from complex reasoning (DeepSeek-AI et al., 2025) to creative generation (OpenAI et al., 2025). However, aligning these models with human values remains a challenge. Standard paradigms—such as Supervised Fine-Tuning (SFT) (Taori et al., 2023; Yang et al., 2024b), Reinforcement Learning from Human Feedback (RLHF) (Ouyang et al., 2022;

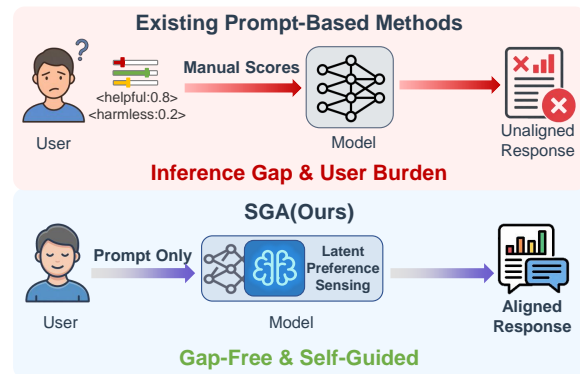


Figure 1: Bridging the training-inference gap in multi-objective alignment. SGA eliminates the burden of manual preference scores and adaptively generates aligned responses from raw prompts.

Bai et al., 2022), and Direct Preference Optimization (DPO) (Rafailov et al., 2023)—typically optimize a scalarized objective. These approaches implicitly approximate an *average-user* preference (Wang et al., 2024a), thereby failing to capture the heterogeneity of human values or accommodate conflicting user needs.

To address this limitation, recent research has pursued multi-objective alignment through distinct methodologies. One line focuses on model merging techniques (Rame et al., 2023; Xie et al., 2025), which interpolate weights fine-tuned on diverse rewards to approximate a Pareto frontier. Another line explores multi-objective optimization via specialized training objectives (Li et al., 2025a; Zhong et al., 2024; Xiong and Singh, 2025; Li et al., 2025c) or inference-time guidance mechanisms (Shi et al., 2024; Lin et al., 2025; Chen et al., 2025; Liu et al., 2025). Meanwhile, prompt-conditioned approaches (Dong et al., 2023; Yang et al., 2024c; Guo et al., 2024; Yang et al., 2024a) have gained traction due to their flexibility. These methods condition the model on explicit reward tokens or control vectors, attempting to dynamically steer

\*Corresponding Author

generation behavior at inference time.

Despite these advancements, there is a practical training-inference discrepancy in existing prompt-conditioned alignment paradigms (Dong et al., 2023; Yang et al., 2024c; Guo et al., 2024; Wang et al., 2024a; Fu et al., 2025). During training, models learn conditional distributions based on ground-truth preference scores (e.g., derived from a reward model or annotated by human experts). In contrast, inference relies on user preference injection (manual inputs such as `<helpful:0.8>` and `<harmless:0.2>`). This reliance creates a significant semantic gap: users often lack the granular intuition to map multi-attribute abstract intent onto the precise scalar scores required by the model. Consequently, relying on such exogenous manual specification, rather than endogenous model understanding, not only leads to suboptimal alignment but also imposes a prohibitive cognitive burden for users, thereby hindering real-world efficiency.

Our objective is to transform preference alignment from an exogenous, user-dependent process into an model-intrinsic capability, as shown in Fig. 1. Note that this transformation poses a structural challenge: the model needs to internalize a complex preference landscape within its representation space, rather than relying on externally provided signals. Standard architectures which decouple reward modeling from generation are limited in capturing this latent relationship. To address this issue, we propose Self-Guided Alignment (SGA), a novel framework that structurally unifies preference sensing with conditional generation. By employing a dual-head architecture on a shared backbone, SGA enables the model to autonomously predict latent preference scores from the context, achieving distributional consistency with negligible inference overhead without imposing cognitive burden on the user.

In summary, our contributions are as follows:

- We propose SGA, a novel framework that unifies conditional generation and preference regression within a shared representation space. By jointly optimizing a composite objective, we enable the model to internalize the preference space—learning latent preference profiles directly from user prompts to construct an endogenous representation.
- During inference, we introduce an adaptive preference sensing mechanism to address the training-inference gap and elimi-

nate the reliance on manual injections. It autonomously infers latent preference scores from the prompts, ensuring distributional consistency and enabling the model to self-guide its generation trajectory with negligible inference overhead.

- Extensive experiments across the Llama and Qwen model families show that SGA outperforms state-of-the-art (SOTA) baselines, achieving superior performance and trade-offs in multi-objective scenarios without manual preference specification.

## 2 Related Work

**Multi-Objective Preference Alignment.** While standard methods like RLHF (Bai et al., 2022) optimize a single scalar reward, real-world alignment often necessitates balancing conflicting values or multiple objectives through Pareto optimization. Early works focused on training-time gradient manipulation. Methods like MORLHF (Li et al., 2024a), MODPO (Zhou et al., 2024), GAPO (Li et al., 2025a) and MOPO (Xiong and Singh, 2025) dynamically reweight gradients to approximate the Pareto frontier, while Panacea (Zhong et al., 2024) seeks to cover the diverse preference space. Parallel to optimization, model merging has emerged as a post-training paradigm. Techniques such as Rewarded Soups (Rame et al., 2023) and Bone Soup (Xie et al., 2025) interpolate weights from models fine-tuned under different reward functions. However, a key limitation of these strategies is their static nature, where fixed preference trade-offs often require retraining to adapt to new user needs, making them less suitable for instance-level personalization.

**Controllable and Prompt-Conditioned Alignment.** To enable dynamic control over model behavior, prompt-conditioned alignment has emerged as a widely adopted paradigm. Frameworks like RiC (Yang et al., 2024c), CPO (Guo et al., 2024), and SteerLM (Dong et al., 2023) have achieved controllability by conditioning models on explicit reward tokens. Despite their flexibility, these approaches face a practical training-inference discrepancy: they rely on ground-truth preferences during training but require explicit user specification at inference, where the difficulty of quantifying abstract intents into scalar vectors imposes a cognitive burden and risks distributional divergence.

**Inference-Time Alignment.** Complementary to prompting, another line of research explores steering generation via auxiliary guidance signals. Inference-time methods such as PARM (Lin et al., 2025), MOD (Shi et al., 2024), and DiffPO (Chen et al., 2025) employ autoregressive reward models, multi-model logits combination, or diffusion-style denoising to guide outputs toward high-reward regions. While enabling fine-grained control, these approaches often require iterative decoding steps or the concurrent execution of multiple auxiliary models. This reliance results in substantial inference latency, which can limit their suitability for real-time applications.

### 3 Methodology

We propose SGA, a unified framework designed to bridge the preference discrepancy between training ground-truth annotations and inference user injections. As shown in Fig. 2, SGA is operated in two stages: (1) a joint training stage, where a shared backbone is optimized via multi-task objectives to simultaneously train a regression head for internalizing the preference space and a Language Model (LM) head for preference-conditioned generation; and (2) an inference stage, where the regression head performs adaptive preference sensing to predict a latent preference score vector, acting as a guidance signal to steer the LM head toward generating multi-objective aligned responses with negligible inference overhead.

#### 3.1 Probabilistic Formulation

Current prompt-conditioned alignment methods typically treat the preference score vector  $s$  as an external condition with the ground-truth ( $s_{\text{gt}}$ ) (Bai et al., 2022; Wang et al., 2024c) during training, whereas inference relies on manual score injection by the users ( $s_{\text{user}}$ ). Thus, this discrepancy introduces a fundamental distributional shift between  $P(y | x, s_{\text{gt}})$  and  $P(y | x, s_{\text{user}})$ . We propose modeling the joint probability of the response  $y$  and the preference score vector  $s$ , conditioned on the prompt  $x$ . We consider that  $x$  often integrates elements such as usage background, application content, and expected outcomes, which often reflect user preferences. For instance, *I'm going on vacation to a ski resort. What should I expect?* reflects the (*background, content, outcome*) as (*vacation, ski, \**), which imply a higher priority for correctness and helpfulness; *can you provide*

*me a joke related to people resigning from their jobs?* represents (*resign, joke, humor*), implying a low priority with correctness/truthfulness. Under this premise, let  $\mathcal{D} = \{(x_i, y_i, s_i)\}_{i=1}^N$  denote the dataset, where  $s \in \mathbb{R}^d$ . We unify preference learning and conditioned generation by explicitly modeling the joint probability distribution  $P_\theta(y, s | x)$ . Our optimization objective is to maximize the expected log-likelihood over the observed data:

$$\mathcal{J}(\theta) = \mathbb{E}_{(x,y,s) \sim \mathcal{D}} [\log P_\theta(y, s | x)]. \quad (1)$$

By applying the chain rule, we factorize this joint distribution into two coupled components: a conditional policy and a preference estimator:

$$P_\theta(y, s | x) = \underbrace{P_\theta(y | x, s)}_{\text{Conditional Policy}} \cdot \underbrace{P_\theta(s | x)}_{\text{Preference Estimator}}. \quad (2)$$

This factorization justifies our dual-head architecture: the language modeling head approximates  $P_\theta(y | x, s)$  by learning to generate response conditioned on explicit scores, while the regression head approximates  $P_\theta(s | x)$  by inferring the optimal preference score vector  $s$  inherent in the prompt. Substituting Eq. 2 into the objective function (Eq. 1) yields the decomposed log-likelihood:

$$\log P_\theta(y, s | x) = \log P_\theta(y | x, s) + \log P_\theta(s | x). \quad (3)$$

For the generative component, we model the response  $y$  as a sequence of tokens generated autoregressively. Conditioned on  $x$  and  $s$ , the conditional log-likelihood is decomposed via the chain rule:

$$\log P_\theta(y | x, s) = \sum_{t=1}^T \log P_\theta(y_t | y_{<t}, x, s), \quad (4)$$

where  $P_\theta(y_t | \cdot)$  denotes a categorical distribution over the vocabulary, parameterized by the softmax output of the language modeling head. Maximizing this term is equivalent to minimizing the standard Cross-Entropy (CE) loss, denoted as  $\mathcal{L}_{\text{CE}}$ . Regarding the preference estimator  $P_\theta(s | x)$ , we model the underlying preference score vector as a multivariate Gaussian distribution characterized by a predicted mean  $\mu_\theta(x)$  and a fixed diagonal covariance  $\sigma^2 I$ :

$$P_\theta(s | x) = \mathcal{N}(s; \mu_\theta(x), \sigma^2 \mathbf{I}) \propto \exp\left(-\frac{\|s - \mu_\theta(x)\|^2}{2\sigma^2}\right). \quad (5)$$

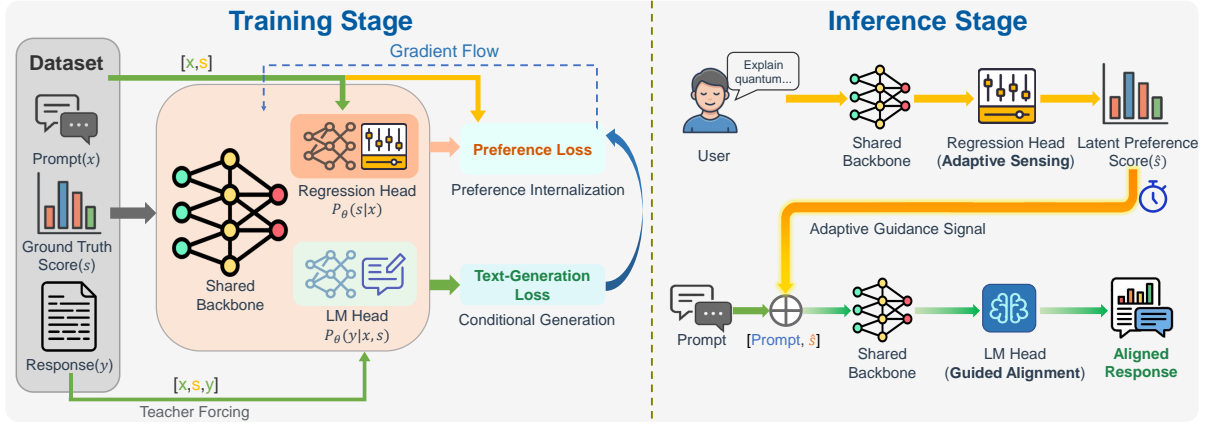


Figure 2: Overview of the SGA Framework. It utilizes a dual-head architecture to internalize the preference space during training. At inference, it employs adaptive sensing to autonomously derive latent preference scores from user prompts, enabling gap-free, self-guided generation.

Maximizing the log-likelihood of this Gaussian term is equivalent to minimizing the squared Euclidean distance (ignoring constant terms):

$$-\log P_{\theta}(s | x) \propto \frac{1}{2\sigma^2} \|s - \mu_{\theta}(x)\|^2. \quad (6)$$

Finally, minimizing the negative joint log-likelihood yields the total loss function  $\mathcal{L}_{\text{total}}$ , formulated as a weighted sum:

$$\begin{aligned} \mathcal{L}_{\text{total}} &= -\log P_{\theta}(y | x, s) - \log P_{\theta}(s | x) \\ &= \mathcal{L}_{\text{CE}} + \lambda \mathcal{L}_{\text{MSE}} \end{aligned} \quad (7)$$

where  $\mathcal{L}_{\text{CE}}$  is the standard cross-entropy loss while  $\mathcal{L}_{\text{MSE}} = \|s - \mu_{\theta}(x)\|^2$ . The weighting coefficient  $\lambda = \frac{1}{2\sigma^2}$  is not a heuristic hyperparameter but theoretically represents the inverse variance (precision) of the preference prior. A larger  $\lambda$  implies a lower variance assumption, treating the regression head as a high-precision guide that must be strictly followed. Appendix A provides more details of this formulation.

While Eq. 1-7 instantiates the conditional policy optimization via Maximum Likelihood Estimation (i.e., SFT with  $\mathcal{L}_{\text{CE}}$ ), our decoupled formulation in Eq. 2 endows the SGA framework with high flexibility. We further extend our method to pairwise preference optimization: given a pairwise preference dataset  $\mathcal{D}_{\text{pref}} = \{(x, s, y_w, y_l)\}$ , where  $y_w$  and  $y_l$  denote the chosen and rejected responses respectively, and  $s$  is the preference score vector associated with the chosen response, we can substitute cross-entropy with the DPO objective for the LM head. The regression head continues to internalize the preference space via the MSE loss

using  $(x, s)$ . Consequently, the total loss for the DPO-variant of SGA is formulated as:

$$\begin{aligned} \mathcal{L}_{\text{DPO}} &= -\log \sigma \left( \beta \log \frac{\pi_{\theta}(y_w | x, s)}{\pi_{\text{ref}}(y_w | x, s)} \right. \\ &\quad \left. - \beta \log \frac{\pi_{\theta}(y_l | x, s)}{\pi_{\text{ref}}(y_l | x, s)} \right). \end{aligned} \quad (8)$$

Finally, we obtain:

$$\mathcal{L}_{\text{total}}^{\text{DPO}} = \mathcal{L}_{\text{DPO}} + \lambda \mathcal{L}_{\text{MSE}} \quad (9)$$

The complete derivation of the DPO-variant of SGA is detailed in Appendix A.

### 3.2 Joint Training for Alignment

**Dual-Head Architecture.** As illustrated in Fig. 2 (Left), we implement SGA using a shared backbone equipped with two task-specific heads. This design enables the model to jointly optimize latent preference sensing and conditional generation policies within a unified representation space. The regression head is instantiated as a Multi-Layer Perceptron (MLP) that maps the hidden state of the prompt’s last token into a preference score vector. It serves as the architectural component of the estimator  $P_{\theta}(s | x)$ , enabling the model to adaptively sense the latent constraints inherent in the user prompt. The LM head functions as the standard causal generation layer. It operates on the augmented input sequence of the prompt and the preference score vector, denoted as  $[x; s]$ , to autoregressively generate the response  $y$ . By conditioning generation on  $[x; s]$ , the head explicitly learns the target distribution  $P_{\theta}(y | x, s)$ , thereby guiding the generation process.

**Optimization Objective and Rationale.** Our primary objective is to simultaneously perform enhanced user preference prediction and aligned response generation. We perform end-to-end training using a composite objective function with Eq. 7 and Eq. 9. Note that  $\mathcal{L}_{\text{CE}}$  and  $\mathcal{L}_{\text{DPO}}$  optimize text generation (in Fig. 2) via Teacher Forcing (Lamb et al., 2016), encouraging the model to strictly adhere to explicit instructions with the normalized ground-truth scores in the prompt.  $\mathcal{L}_{\text{MSE}}$  minimizes the Mean Squared Error of the preference internalization. By optimizing its regression head to predict the ground-truth score  $s$  conditioned solely on the prompt  $x$ , we let the shared backbone to extract latent preference features directly from the input text, assisting a consistent generation process. Note that  $\lambda = \frac{1}{2\sigma^2}$  corresponds to a low-variance prior, encouraging the model to capture robust features for the strict regression constraint. From an optimization perspective, a large  $\lambda$  prevents the regression signal from being diluted, so that the regression head acts as a high-precision internal guide rather than a noisy estimator.

### 3.3 Self-Guided Generation at Inference

SGA forms a closed-loop system where the model autonomously derives its own guidance. The inference is operated in two coordinated phases.

**Adaptive Preference Sensing and Guided Generation.** Given a raw user prompt  $x$ , the inference process begins with adaptive preference sensing. The model utilizes the regression head to map the input representation into a latent preference score vector  $\hat{s}$ . This step retrieves alignment constraints internalized during training, serving as a mechanism for latent intent estimation. Subsequently, in the guided generation stage, this self-predicted score is seamlessly integrated as an intrinsic control signal. By constructing the augmented input  $[x; \hat{s}]$ , the model actively steers the generation trajectory of the LM head towards the desired regions of the distribution  $P_\theta(y | x, \hat{s})$ . Notably, since  $\hat{s}$  is derived from the model’s own internal representation rather than external injection, the generator operates within a familiar, high-confidence regime, which enhances alignment and coherence of the generation. Note that SGA can still receive external steering signal  $s$  although the unified representation space from joint training may perturb the derived Pareto frontier. We consider SGA represents an independent direction rather than a replacement of

traditional Pareto frontier approximation. We include a steering analysis in the ablation study under Section 4.3.2.

**Endogenous Consistency and Efficiency.** SGA promotes endogenous alignment consistency and inference efficiency. By replacing exogenous manual scores with endogenous predicted signals, SGA mitigates the training-inference distributional shift so that the generation is conditioned on the preference space learned during training. Computationally, SGA incurs negligible overhead. By integrating adaptive sensing into the prompt pre-fill stage via KV caching, the regression head acts as a lightweight probe on existing hidden states without requiring backbone re-computation, thereby maintaining the real-time interactivity of standard LLM inference.

## 4 Experimental Results

### 4.1 Experiment Setup

**Models.** We evaluate six base models from two competitive open-source families: the Llama series, including Llama-3.1-8B-Instruct, Llama-3.2-1B-Instruct and Llama-3.2-3B-Instruct (Grattafiori et al., 2024); and the Qwen series, including Qwen2.5-7B-Instruct (Qwen et al., 2025), Qwen3-4B and Qwen3-8B (Yang et al., 2025). For brevity, we omit the “Instruct” suffix in subsequent tables (e.g., Llama3.1-8B, Llama3.2-1B, Llama3.2-3B, and Qwen2.5-7B).

**Datasets and Baselines.** We evaluate two SGA variants,  $\text{SGA}_{\text{SFT}}$  and  $\text{SGA}_{\text{DPO}}$ , on three mainstream datasets for multi-objective alignment supported by recent studies (Kwon et al., 2025; Li et al., 2025b; Wang et al., 2025b): HH-RLHF (Bai et al., 2022), focusing on general utility and safety (Helpful and Harmless); UltraFeedback (Cui et al., 2023), covering four core quality axes (Instruction-Following, Truthfulness, Honesty, and Helpfulness); and HelpSteer2 (Wang et al., 2024c), assessing steerability across five dimensions (Helpfulness, Correctness, Coherence, Complexity, and Verbosity). We evaluate SGA against four representative baselines: Vanilla SFT, RiC, CPO, and MODPO. We report both overall and fine-grained win-rates against reference responses in parallel to an ablation study, utilizing three SOTA reward models (Yang et al., 2024c; Wang et al., 2024b) to ensure a robust and unbiased assessment. Unless otherwise specified, all results are reported

Dataset	Model	RiC	CPO	SFT	MODPO	SGA <sub>SFT</sub>	SGA <sub>DPO</sub>
HH-RLHF	Llama3.2-1B	43.0	52.7	46.4	<u>53.2</u>	53.0	<b>54.1</b>
	Llama3.2-3B	48.6	51.0	45.2	<u>55.0</u>	54.5	<b>55.5</b>
	Llama3.1-8B	50.1	50.8	44.9	<u>52.1</u>	51.9	<b>57.4</b>
	Qwen3-4B	44.7	50.5	44.2	<u>54.7</u>	51.9	<b>60.2</b>
	Qwen2.5-7B	49.7	50.7	44.3	<b>61.3</b>	50.5	<u>61.2</u>
	Qwen3-8B	52.6	49.9	43.9	<u>58.8</u>	53.6	<b>62.2</b>
	<i>Average</i>	48.1	50.9	44.8	<u>55.9</u>	52.6	<b>58.4</b>
UltraFeedback	Llama3.2-1B	14.8	16.1	16.6	<u>19.6</u>	18.4	<b>26.1</b>
	Llama3.2-3B	32.3	30.8	31.3	<u>37.6</u>	34.1	<b>53.5</b>
	Llama3.1-8B	41.5	41.5	39.7	<u>50.8</u>	47.5	<b>62.9</b>
	Qwen3-4B	33.8	33.0	32.4	<u>44.5</u>	42.4	<b>54.0</b>
	Qwen2.5-7B	36.0	37.9	34.2	<u>51.4</u>	42.0	<b>59.5</b>
	Qwen3-8B	36.2	38.7	34.5	<u>49.3</u>	44.5	<b>57.2</b>
	<i>Average</i>	32.4	33.0	31.4	<u>42.2</u>	38.1	<b>52.2</b>
HelpSteer2	Llama3.2-1B	32.0	30.7	29.4	36.1	<b>39.7</b>	<u>38.4</u>
	Llama3.2-3B	49.4	49.5	39.7	53.6	<u>58.3</u>	<b>60.8</b>
	Llama3.1-8B	58.6	57.6	45.0	64.1	<u>66.9</u>	<b>69.2</b>
	Qwen3-4B	36.6	39.8	34.7	<b>54.5</b>	<u>53.2</u>	49.5
	Qwen2.5-7B	39.7	43.4	36.7	<u>58.0</u>	<b>58.3</b>	56.8
	Qwen3-8B	41.9	44.6	36.2	<b>61.0</b>	58.8	<u>58.9</u>
	<i>Average</i>	43.0	44.3	36.9	54.6	<b>55.9</b>	<u>55.6</u>

Table 1: Win rate comparison against reference responses. We report the macro-averaged win rate (%) across individual objectives. The best results are highlighted in **bold**, and the second-best results are underlined.

as percentages (%). More details are provided in Appendix B.2.

**Complementary Benchmarks.** The evaluation is conducted along two complementary dimensions to provide a holistic assessment. First, for general dialogue capabilities, we adopt the standard LLM-as-a-judge paradigm to assess the model’s robustness in open-ended generation. Specifically, we utilize MT-Bench (Zheng et al., 2023) to evaluate multi-turn conversational abilities and reasoning, and Arena-Hard (Li et al., 2024b) to test performance on complex real-world prompts aligned with human preferences in the Chatbot Arena, following prior evaluation practices (Wang et al., 2025a; Yang et al., 2025). Second, we conduct out-of-distribution (OOD) evaluation with TruthfulQA (Lin et al., 2022) and SafetyBench (Zhang et al., 2024). TruthfulQA evaluates resistance to common misconceptions, while SafetyBench measures robustness to unsafe or adversarial prompts. Further implementation details are provided in Appendix C.1.

## 4.2 Main Results

### 4.2.1 Overall Alignment Performance

Table 1 summarizes the comprehensive evaluation of overall alignment performance by reporting macro-averaged win rates across task-specific objectives. A clear performance hierarchy emerges from the comparison, demonstrating the robust generalizability of our method across different model

sizes ranging from 1B to 8B parameters. Notably, SGA achieves an average improvement of 9% over all baselines. Specifically, SGA<sub>SFT</sub> yields substantial and consistent improvements over standard SFT, while also outperforming other baseline methods, including RiC and CPO, by an average of 5.7%. Furthermore, when extended to the preference optimization paradigm to benchmark against state-of-the-art multi-objective approaches, SGA<sub>DPO</sub> successfully surpasses the strong MODPO baseline, ultimately achieving the highest overall win rates (12.3% on average). These results indicate that by internalizing the preference space, SGA establishes an intrinsic adaptive sensing capability and maintains training-inference consistency at the distributional level. This endogenous mechanism allows the model to autonomously navigate complex multi-objective trade-offs, thereby improving alignment performance while eliminating the cognitive burden of manual specification.

### 4.2.2 Fine-grained Multi-Objective Analysis

A fine-grained analysis of complex preference trade-offs is presented in Table 2. We derive observations from two aspects as follows.

**Holistic Structural Control.** Table 2 shows that SGA demonstrates comprehensive superiority on preference aspects such as *Helpful*, *Helpfulness*, *Honesty*, *Truthfulness*, *Correctness*, and *Coherence*, which we denote as Set I, while lagging behind for *Instruction*, *Complexity*, and *Verbosity*, which named as Set II. Note that SFT and RiC excel for *Harmless*, but they fail to maintain comparable performance in *Helpfulness*. Despite a weakness in *Harmless*, SGA’s consistent improvements over Set I suggest that it achieves better structural control across multiple objectives, which correspond to more critical response qualities in practical alignment settings. Notably, higher *Verbosity* does not necessarily indicate better responses. While baselines often surpass SGA in *Verbosity* (e.g., RiC achieves 67.4% vs. SGA<sub>DPO</sub>’s 49.7% on Llama3.1-8B), this gain often comes at the expense of *Complexity* (RiC 53.6% vs. SGA<sub>DPO</sub> 71.4%). This reflects a length bias that does not translate into better response quality, while SGA maintains robust performance across other key dimensions (visualized in Fig. 3).

**Navigating Preference Trade-offs.** Optimizing LLMs often involves navigating inherent trade-offs between different objectives. We observe that

Model	Method	HH-RLHF		UltraFeedback				HelpSteer2				
		Helpful	Harmless	Helpfulness	Honesty	Instruction	Truthfulness	Helpfulness	Correctness	Coherence	Complexity	Verbosity
Llama3.2-1B	RiC	18.3	<b>67.7</b>	16.4	12.8	17.2	12.7	18.1	17.7	19.3	48.8	56.2
	CPO	52.2	53.2	15.1	12.5	18.3	18.4	17.4	17.6	19.3	46.5	52.7
	SFT	30.3	<u>62.4</u>	16.4	13.7	17.5	<u>18.6</u>	14.5	12.4	17.2	43.1	<u>59.9</u>
	MODPO	<u>60.0</u>	46.3	<u>24.5</u>	<u>16.7</u>	<b>21.6</b>	15.7	19.5	18.5	19.3	<b>60.4</b>	<b>62.7</b>
	SGA <sub>SFT</sub>	<b>60.2</b>	45.7	<u>24.5</u>	16.3	19.8	<b>13.0</b>	<u>27.0</u>	<u>25.3</u>	<u>29.7</u>	57.5	59.1
	SGA <sub>DPO</sub>	<u>60.0</u>	48.2	<b>43.2</b>	<b>18.0</b>	19.3	<b>24.0</b>	<b>28.3</b>	<b>25.9</b>	<b>31.2</b>	<u>59.1</u>	47.6
Llama3.2-3B	RiC	44.9	52.2	31.9	30.0	32.0	<u>35.4</u>	45.1	44.1	48.0	50.8	58.9
	CPO	48.8	<u>53.2</u>	28.1	29.4	33.4	32.1	42.0	43.4	55.1	53.7	53.3
	SFT	29.6	<b>60.8</b>	28.7	28.5	33.0	35.1	29.9	28.8	32.1	50.8	57.0
	MODPO	65.6	44.5	<u>38.4</u>	<u>36.0</u>	<u>40.6</u>	35.2	47.3	44.0	47.5	<u>66.0</u>	<b>63.3</b>
	SGA <sub>SFT</sub>	<u>67.4</u>	41.6	35.2	30.5	37.8	32.9	<u>62.0</u>	<u>62.4</u>	<u>61.2</u>	55.0	51.0
	SGA <sub>DPO</sub>	<b>82.5</b>	28.5	<b>67.7</b>	<b>47.3</b>	<b>43.1</b>	<b>56.1</b>	<b>64.1</b>	<b>66.0</b>	<b>64.0</b>	<b>67.7</b>	42.2
Llama3.1-8B	RiC	40.4	<u>59.8</u>	41.3	40.5	39.6	44.6	57.2	55.2	59.7	53.6	<b>67.4</b>
	CPO	48.8	52.9	38.5	41.9	40.9	44.7	58.6	59.5	57.5	55.9	56.5
	SFT	28.0	<b>61.8</b>	33.6	38.0	42.4	44.8	39.2	40.2	41.3	50.0	54.3
	MODPO	56.2	47.9	<u>54.4</u>	<u>47.9</u>	<b>56.0</b>	45.0	64.2	60.9	62.1	<u>71.0</u>	<u>62.4</u>
	SGA <sub>SFT</sub>	<u>57.0</u>	46.9	44.2	47.4	<u>52.1</u>	<u>46.1</u>	<u>71.0</u>	<u>73.4</u>	<u>72.2</u>	61.4	56.6
	SGA <sub>DPO</sub>	<b>72.4</b>	42.4	<b>77.6</b>	<b>58.2</b>	50.1	<b>65.9</b>	<b>74.7</b>	<b>75.1</b>	<b>75.1</b>	<b>71.4</b>	49.7
Qwen3-4B	RiC	29.6	<u>59.8</u>	29.6	32.8	34.1	<u>38.8</u>	35.1	34.7	34.9	38.0	40.2
	CPO	46.3	54.6	29.9	30.9	34.7	36.7	35.6	37.8	37.8	45.2	42.7
	SFT	27.1	<b>61.3</b>	28.0	29.7	33.9	38.1	25.1	25.9	24.5	46.7	51.2
	MODPO	<u>68.3</u>	41.1	44.7	<u>46.0</u>	<b>49.1</b>	38.3	46.1	43.0	44.3	<b>77.0</b>	<b>62.0</b>
	SGA <sub>SFT</sub>	50.1	53.7	<u>45.7</u>	<u>44.2</u>	<u>44.4</u>	35.3	<u>46.3</u>	<u>47.0</u>	<u>48.6</u>	<u>63.6</u>	<u>60.4</u>
	SGA <sub>DPO</sub>	<b>87.1</b>	33.3	<b>62.7</b>	<b>51.8</b>	44.0	<b>57.7</b>	<b>55.0</b>	<b>54.1</b>	<b>48.7</b>	59.1	30.7
Qwen2.5-7B	RiC	43.0	56.4	28.6	33.6	38.2	43.8	37.6	39.3	37.7	40.9	42.9
	CPO	53.3	48.1	30.8	36.5	40.0	44.1	40.5	42.6	42.0	47.9	44.2
	SFT	27.7	<b>60.8</b>	29.8	31.5	36.5	39.2	25.9	29.0	29.3	45.0	54.3
	MODPO	<u>83.3</u>	39.2	<u>51.0</u>	<u>51.2</u>	<b>56.6</b>	46.9	51.9	56.3	50.8	<b>73.8</b>	<b>57.3</b>
	SGA <sub>SFT</sub>	43.0	<u>58.0</u>	32.8	41.4	44.8	<u>49.1</u>	<u>59.2</u>	<u>60.9</u>	<u>57.6</u>	58.2	<u>55.7</u>
	SGA <sub>DPO</sub>	<b>89.6</b>	32.8	<b>70.1</b>	<b>55.4</b>	<u>47.9</u>	<b>64.4</b>	<b>65.9</b>	<b>63.4</b>	<b>62.0</b>	<u>58.8</u>	34.0
Qwen3-8B	RiC	57.2	48.0	29.4	35.5	37.6	<u>42.3</u>	37.9	42.4	40.6	42.8	45.9
	CPO	47.6	<u>52.2</u>	35.2	38.2	40.7	40.8	41.0	42.4	43.9	49.2	46.3
	SFT	27.5	<b>60.4</b>	28.8	32.4	36.4	40.2	27.4	28.0	30.5	44.8	50.4
	MODPO	<u>83.3</u>	34.2	<u>52.1</u>	<u>50.9</u>	<b>52.5</b>	41.8	55.7	55.4	53.6	<b>77.6</b>	<b>62.7</b>
	SGA <sub>SFT</sub>	57.3	50.0	49.8	46.9	45.0	36.2	<u>60.6</u>	<u>62.6</u>	<u>60.6</u>	55.6	<u>54.4</u>
	SGA <sub>DPO</sub>	<b>88.4</b>	36.0	<b>67.7</b>	<b>56.2</b>	<u>45.9</u>	<b>59.1</b>	<b>67.8</b>	<b>71.0</b>	<b>68.4</b>	<u>58.9</u>	28.6

Table 2: Fine-grained performance comparison. SGA achieves the highest win rates in a majority of the evaluated quality and structural dimensions. ‘‘Instruction’’ refers to Instruction-Following.

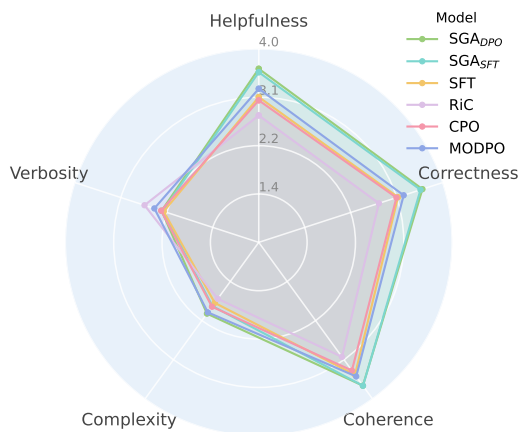


Figure 3: Holistic evaluation on HelpSteer2 with Llama3.1-8B-Instruct. More results for all 6 models are detailed in Appendix C.2.

baselines like MODPO tend to over-optimize attributes in Set II, often at the expense of other critical qualities. In contrast, SGA demonstrates a stronger ability to balance these competing objectives, achieving a better overall trade-off as shown in Fig. 3. While maintaining reasonable perfor-

Model	Method	Arena-Hard	MT-Bench
Llama3.1-8B	SFT	10.8%	6.51
	RiC	15.3%	6.38
	CPO	12.7%	6.08
	MODPO	26.1%	7.62
	SGA <sub>SFT</sub>	25.8%	7.56
	SGA <sub>DPO</sub>	<b>30.4%</b>	<b>7.78</b>
Qwen3-8B	SFT	11.2%	6.46
	RiC	21.9%	6.57
	CPO	16.6%	6.10
	MODPO	40.7%	7.58
	SGA <sub>SFT</sub>	38.1%	7.73
	SGA <sub>DPO</sub>	<b>44.5%</b>	<b>7.92</b>

Table 3: Evaluation on general chat capabilities. We report the Arena-Hard win rate and the MT-Bench average score (1–10 scale, averaged over two turns).

mance on structural aspects, both SGA<sub>SFT</sub> and SGA<sub>DPO</sub> prioritize the essential Set I preferences.

### 4.2.3 Evaluation of Generalization and OOD

The conversational generalization of the finetuned models (based on HH-RLHF, UltraFeedback, and HelpSteer2) on Arena-Hard and MT-Bench is presented in Table 3. OOD benchmark results are shown in Table 4. For each method, the reported

Model	Base	SFT	RiC	CPO	MODPO	SGA <sub>SFT</sub>	SGA <sub>DPO</sub>
Llama3.2-1B	25.4	26.2	25.3	26.2	26.0	<u>26.4</u>	<b>27.0</b>
	42.3	42.6	42.2	43.0	43.1	<u>43.5</u>	<b>43.6</b>
	<u>54.1</u>	50.1	43.7	52.6	53.6	53.6	<b>53.9</b>
Llama3.2-3B	<u>31.7</u>	28.7	26.8	27.2	31.2	31.1	<b>32.4</b>
	50.1	46.9	42.7	44.7	<u>50.8</u>	50.2	<b>51.0</b>
	71.1	68.2	66.2	70.4	<b>72.3</b>	<u>71.5</u>	<u>71.5</u>
Llama3.1-8B	36.0	36.0	35.6	35.8	35.4	<u>36.1</u>	<b>36.5</b>
	<u>53.4</u>	<u>53.4</u>	53.1	53.1	52.5	<u>53.4</u>	<b>54.0</b>
	76.3	<u>77.2</u>	73.3	75.8	77.3	<u>77.5</u>	<b>77.6</b>
Qwen3-4B	34.2	33.2	32.8	32.4	33.1	<u>33.8</u>	<b>33.9</b>
	52.9	52.5	51.9	51.7	52.5	<u>53.0</u>	<b>53.3</b>
	75.5	72.4	74.4	74.0	<u>76.5</u>	76.3	<b>77.2</b>
Qwen2.5-7B	43.5	43.4	42.9	42.9	43.5	<u>44.4</u>	<b>45.3</b>
	62.6	62.2	61.5	62.5	62.4	<u>62.7</u>	<b>63.8</b>
	80.6	80.6	80.0	79.7	80.9	<u>81.0</u>	<b>81.2</b>
Qwen3-8B	32.9	<b>33.3</b>	32.2	32.4	32.5	<u>33.2</u>	<b>33.3</b>
	54.3	54.0	53.8	53.8	53.9	<u>54.5</u>	<b>54.8</b>
	78.9	76.3	76.3	76.6	78.9	<u>79.0</u>	<b>79.3</b>

Table 4: Overall performance comparison on TruthfulQA and SafetyBench. For each model, the three rows correspond to TruthfulQA MC1, TruthfulQA MC2, and SafetyBench accuracy, respectively.

scores are averaged across models trained on different datasets. The consistent performance boosts of SGA<sub>SFT</sub> and SGA<sub>DPO</sub> demonstrate stronger generalization capabilities compared to SOTA methods. It implies that SGA enhances the capability to maintain the natural coherence and reasoning depth, gaining advantage for open-ended tasks.

### 4.3 Ablation Studies and Analysis

We conduct ablation studies to assess the effectiveness of SGA’s core components. We first evaluate the feasibility of preference estimation from the prompts alone. Then, we investigate the efficacy of preference sensing compared to external steering. A complementary analysis of the computational overhead is detailed in Appendix C.2.3. Additional ablation studies examining the sensitivity of the hyperparameter  $\lambda$  (loss weighting) are presented in Appendix C.2.2.

#### 4.3.1 Feasibility of Preference Estimation

We demonstrate that the preference score  $s$  can often be estimated as  $\hat{s}$  from the prompt  $x$  alone given relevant information of (*background, content, outcome*). Note that the estimated preference may differ from the truth when key information is missing or is unable to reflect the users’ intention. Fig. 4 is a statistical visualization on HelpSteer2 of the absolute error  $|s - \hat{s}|$ . Note that  $s \in \{0, 0.25, 0.5, 0.75, 1\}$  given the categorical annotation and  $\hat{s} \in [0, 1]$ . It shows that such a major cluster of samples exists for which  $\hat{s}$  is close to  $s$  ( $|s - \hat{s}| < 0.2$ ).

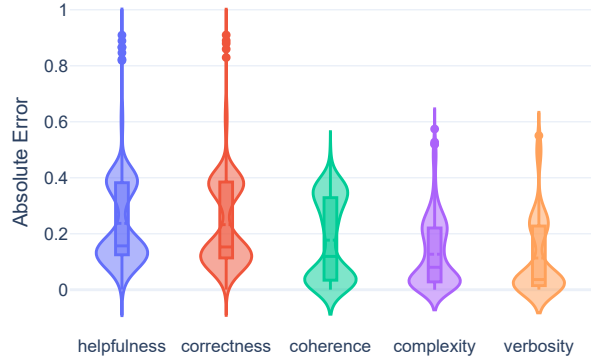


Figure 4: Distribution of absolute error between the prediction of preferences from the prompt alone and the ground-truth scores annotated in HelpSteer2.

Scores ( $s$ )	HH-RLHF	UltraFeedback	HelpSteer2
<b>Model: Llama3.2-3B</b>			
0.2 / 0.25	53.6	33.6	47.1
0.5	53.8	32.7	49.2
1.0	54.2	32.1	54.8
<i>gt</i>	54.1	<b>34.8</b>	53.8
SGA <sub>SFT</sub>	<b>54.5</b>	34.1	<b>58.3</b>
<b>Model: Qwen3-4B</b>			
0.2 / 0.25	49.9	38.8	34.0
0.5	50.8	39.8	41.7
1.0	51.5	42.1	48.4
<i>gt</i>	51.2	41.7	47.4
SGA <sub>SFT</sub>	<b>51.9</b>	<b>42.4</b>	<b>53.2</b>

Table 5: Ablation on inference strategies. We report Win Rate ( $\%$ ) across three datasets. For the low-score setting,  $s = 0.2$  is used for HelpSteer2 and  $s = 0.25$  for UltraFeedback and HH-RLHF. *gt* represents ground-truth steered results.

Meanwhile, there is a larger difference ( $0.3 < |s - \hat{s}| < 0.5$ ) for another collection of samples, which might be caused by misaligned information in the prompts or potential annotation biases. We also investigate significantly deviated samples (e.g.,  $|s - \hat{s}| > 0.9$ ) and they are not facilitated with key background and intentional outcome with *content* alone, failed to reflect the potential preferences. A sampled comparison of such a high deviation with  $\hat{s} = [0.91, 0.92, 0.98, 0.50, 0.56]$  (helpfulness, correctness, coherence, complexity, verbosity) vs.  $s = [0, 0.25, 1, 0.5, 0.5]$  indicates that SGA is likely to adopt a *default* scheme of high preference scores in these situations.

#### 4.3.2 Efficacy of Adaptive Preference Sensing

We compare SGA<sub>SFT</sub> against two external steering variants: fixed-value steering (e.g.,

{0.2/0.25, 0.5, 1.0} for all preferences) as a vanilla comparison and ground-truth steering as an optimistic reference, as shown in Table 5.

**Sensitivity to Preference Scores.** We observe that there is a positive correlation between the preference scores and the win rates for fixed-value steering. For instance, on HelpSteer2 with Llama3.2-3B, performance scales from 47.1% to 54.8% when  $s$  increases from 0.2 to 1.0. It implies that the SGA acquires a consistent dependency between preference scores and favored generation, supporting the effectiveness of joint training. Note that ground-truth steering is not the upper bound here due to the coarse annotation granularity ( $s \in \{0, 0.25, 0.5, 0.75, 1\}$ ), while steering with fixed-value 1.0 (which reflects an equal emphasis of all preferences) may result in a response that is favored more by the judge reward model.

**Adaptive Sensing vs. Steered Preferences.** We also observe that SGA often outperforms both fixed-value and ground-truth steering. Compared to the former naive equal-high emphasis and the latter coarse-grained scoring, our approach consistently performs better. For example, on Llama3.2-3B (HelpSteer2), SGA achieves a win rate of 58.3%, clearly outperforming the Fixed-1.0 setting (54.8%). These observations suggest that a naive maximization of all attributes can lead to suboptimal outcomes, given the inherent conflicts between competing objectives. This superiority can be attributed to the adaptive preference sensing mechanism, which enables finer-grained control and better alignment.

## 5 Conclusions

In this work, we proposed SGA, a unified framework to address the training-inference gap in prompt-conditioned alignment paradigms. Utilizing the joint training for unified preference space and self-guided generation at inference, SGA exhibited superior performance over several SOTA methods. Facilitated with variants such as  $SGA_{SFT}$  and  $SGA_{DPO}$ , we demonstrated that SGA provided a unified architecture supporting both supervised fine-tuning and preference-based optimization. Our ablation study illustrated the feasibility of estimating the preference scores from the prompts alone contrasting the traditional annotation procedure with prompt-response pairs. Meanwhile, external steering of SGA was demonstrated with fixed-

values and the ground-truth scores, although the jointly acquired representation space marked its essential difference compared to traditional Pareto frontier approximation methods.

## 6 Limitations

While our approach demonstrates promising results, we acknowledge several limitations. First, this work implements SGA with the SFT and DPO, focusing on its design and the contribution of adaptive preference sensing rather than conducting extensively benchmark against a wide range of policy learning strategies (e.g., PPO, GRPO, etc.) or computationally intensive decoding strategies. We consider SGA orthogonal to these approaches and leave the exploration of integrating endogenous sensing with RL-based optimization to future work. Second, although SGA demonstrates strong empirical performance, its evaluation remains limited to the settings explored in this work. Future work will investigate its scalability to larger foundation models and its applicability to broader alignment scenarios.

## References

- Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, Nicholas Joseph, Saurav Kadavath, Jackson Kernion, Tom Conerly, Sheer El-Showk, Nelson Elhage, Zac Hatfield-Dodds, Danny Hernandez, Tristan Hume, and 12 others. 2022. [Training a helpful and harmless assistant with reinforcement learning from human feedback](#). *Preprint*, arXiv:2204.05862.
- Ruizhe Chen, Wenhao Chai, Zhifei Yang, Xiaotian Zhang, Ziyang Wang, Tony Quek, Joey Tianyi Zhou, Soujanya Poria, and Zuozhu Liu. 2025. [DiffPO: Diffusion-styled preference optimization for inference time alignment of large language models](#). In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 18910–18925, Vienna, Austria. Association for Computational Linguistics.
- Ganqu Cui, Lifan Yuan, Ning Ding, Guanming Yao, Wei Zhu, Yuan Ni, Guotong Xie, Zhiyuan Liu, and Maosong Sun. 2023. [Ultrafeedback: Boosting language models with high-quality feedback](#). *Preprint*, arXiv:2310.01377.
- DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, and 181 others.

2025. [Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning](#). *Preprint*, arXiv:2501.12948.
- Yi Dong, Zhilin Wang, Makesh Sreedhar, Xianchao Wu, and Oleksii Kuchaiev. 2023. [SteerLM: Attribute conditioned SFT as an \(user-steerable\) alternative to RLHF](#). In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 11275–11288, Singapore. Association for Computational Linguistics.
- Tingchen Fu, Yupeng Hou, Julian McAuley, and Rui Yan. 2025. [Unlocking decoding-time controllability: Gradient-free multi-objective alignment with contrastive prompts](#). In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 366–384, Albuquerque, New Mexico. Association for Computational Linguistics.
- Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, and 542 others. 2024. [The llama 3 herd of models](#). *Preprint*, arXiv:2407.21783.
- 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. 2024. [Controllable preference optimization: Toward controllable multi-objective alignment](#). In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 1437–1454, Miami, Florida, USA. Association for Computational Linguistics.
- Minchan Kwon, Junwon Ko, Kangil Kim, and Junmo Kim. 2025. [Preference distillation via value based reinforcement learning](#). *Preprint*, arXiv:2509.16965.
- Alex M Lamb, Anirudh Goyal ALIAS PARTH GOYAL, Ying Zhang, Saizheng Zhang, Aaron C Courville, and Yoshua Bengio. 2016. [Professor forcing: A new algorithm for training recurrent networks](#). In *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc.
- Chengao Li, Hanyu Zhang, Yunkun Xu, Hongyan Xue, Xiang Ao, and Qing He. 2025a. [Gradient-adaptive policy optimization: Towards multi-objective alignment of large language models](#). In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11214–11232, Vienna, Austria. Association for Computational Linguistics.
- Jiahui Li, Hanlin Zhang, Fengda Zhang, Tai-Wei Chang, Kun Kuang, Long Chen, and Jun Zhou. 2024a. [Optimizing language models with fair and stable reward composition in reinforcement learning](#). In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 10122–10140, Miami, Florida, USA. Association for Computational Linguistics.
- Long-Fei Li, Yu-Yang Qian, Peng Zhao, and Zhi-Hua Zhou. 2025b. [Provably efficient online rlhf with one-pass reward modeling](#). *Preprint*, arXiv:2502.07193.
- Moxin Li, Yuantao Zhang, Wenjie Wang, Wentao Shi, Zhuo Liu, Fuli Feng, and Tat-Seng Chua. 2025c. [Self-improvement towards pareto optimality: Mitigating preference conflicts in multi-objective alignment](#). *Preprint*, arXiv:2502.14354.
- Tianle Li, Wei-Lin Chiang, Evan Frick, Lisa Dunlap, Tianhao Wu, Banghua Zhu, Joseph E. Gonzalez, and Ion Stoica. 2024b. [From crowdsourced data to high-quality benchmarks: Arena-hard and benchbuilder pipeline](#). *Preprint*, arXiv:2406.11939.
- Baijiong Lin, Weisen Jiang, Yuancheng Xu, Hao Chen, and Ying-Cong Chen. 2025. [PARM: Multi-objective test-time alignment via preference-aware autoregressive reward model](#). In *Forty-second International Conference on Machine Learning*.
- Stephanie Lin, Jacob Hilton, and Owain Evans. 2022. [TruthfulQA: Measuring how models mimic human falsehoods](#). In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3214–3252, Dublin, Ireland. Association for Computational Linguistics.
- Qi Liu, Jingqing Ruan, Hao Li, Haodong Zhao, Desheng Wang, Jiansong Chen, Wan Guanglu, Xunliang Cai, Zhi Zheng, and Tong Xu. 2025. [AMoPO: Adaptive multi-objective preference optimization without reward models and reference models](#). In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 8832–8866, Vienna, Austria. Association for Computational Linguistics.
- OpenAI, :, Ahmed El-Kishky, Alexander Wei, Andre Saraiva, Borys Minaiev, Daniel Selsam, David Dohan, Francis Song, Hunter Lightman, Ignasi Clavera, Jakub Pachocki, Jerry Tworek, Lorenz Kuhn, Lukasz Kaiser, Mark Chen, Max Schwarzer, Mostafa Rohaninejad, Nat McAleese, and 7 others. 2025. [Competitive programming with large reasoning models](#). *Preprint*, arXiv:2502.06807.
- Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F Christiano, Jan Leike, and Ryan Lowe. 2022. [Training language models to follow instructions with human feedback](#). In *Advances in Neural Information Processing Systems*, volume 35, pages 27730–27744. Curran Associates, Inc.
- Qwen, :, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin

- Yang, Jiayi Yang, Jingren Zhou, and 25 others. 2025. [Qwen2.5 technical report](#). *Preprint*, arXiv:2412.15115.
- Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2023. [Direct preference optimization: Your language model is secretly a reward model](#). In *Advances in Neural Information Processing Systems*, volume 36, pages 53728–53741. Curran Associates, Inc.
- Alexandre Rame, Guillaume Couairon, Corentin Dancette, Jean-Baptiste Gaya, Mustafa Shukor, Laure Soulier, and Matthieu Cord. 2023. [Rewarded soups: towards pareto-optimal alignment by interpolating weights fine-tuned on diverse rewards](#). In *Thirty-seventh Conference on Neural Information Processing Systems*.
- Ruizhe Shi, Yifang Chen, Yushi Hu, Alisa Liu, Hananeh Hajishirzi, Noah A. Smith, and Simon S. Du. 2024. [Decoding-time language model alignment with multiple objectives](#). In *Advances in Neural Information Processing Systems*, volume 37, pages 48875–48920. Curran Associates, Inc.
- Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2023. Stanford alpaca: An instruction-following llama model. [https://github.com/tatsu-lab/stanford\\_alpaca](https://github.com/tatsu-lab/stanford_alpaca).
- Haoxiang Wang, Yong Lin, Wei Xiong, Rui Yang, Shizhe Diao, Shuang Qiu, Han Zhao, and Tong Zhang. 2024a. [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)*, pages 8642–8655, Bangkok, Thailand. Association for Computational Linguistics.
- Haoxiang Wang, Wei Xiong, Tengyang Xie, Han Zhao, and Tong Zhang. 2024b. [Interpretable preferences via multi-objective reward modeling and mixture-of-experts](#). In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 10582–10592, Miami, Florida, USA. Association for Computational Linguistics.
- Zhilin Wang, Alexander Bukharin, Olivier Delalleau, Daniel Egert, Gerald Shen, Jiaqi Zeng, Oleksii Kuchaiev, and Yi Dong. 2025a. [Helpsteer2-preference: Complementing ratings with preferences](#). In *International Conference on Learning Representations*, volume 2025, pages 56548–56575.
- Zhilin Wang, Yi Dong, Olivier Delalleau, Jiaqi Zeng, Gerald Shen, Daniel Egert, Jimmy J. Zhang, Makesh Narsimhan Sreedhar, and Oleksii Kuchaiev. 2024c. [Helpsteer 2: Open-source dataset for training top-performing reward models](#). In *Advances in Neural Information Processing Systems*, volume 37, pages 1474–1501. Curran Associates, Inc.
- Zhilin Wang, Jiaqi Zeng, Olivier Delalleau, Ellie Evans, Daniel Egert, Hoo-Chang Shin, Felipe Soares, Yi Dong, and Oleksii Kuchaiev. 2025b. [Rlbf: Binary flexible feedback to bridge between human feedback & verifiable rewards](#). *Preprint*, arXiv:2509.21319.
- Guofu Xie, Xiao Zhang, Ting Yao, and Yunsheng Shi. 2025. [Bone soups: A seek-and-soup model merging approach for controllable multi-objective generation](#). In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 27237–27263, Vienna, Austria. Association for Computational Linguistics.
- Nuoya Xiong and Aarti Singh. 2025. [Projection optimization: A general framework for multi-objective and multi-group RLHF](#). In *Forty-second International Conference on Machine Learning*.
- An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, and 41 others. 2025. [Qwen3 technical report](#). *Preprint*, arXiv:2505.09388.
- Kailai Yang, Zhiwei Liu, Qianqian Xie, Jimin Huang, Tianlin Zhang, and Sophia Ananiadou. 2024a. [Metaaligner: Towards generalizable multi-objective alignment of language models](#). In *Advances in Neural Information Processing Systems*, volume 37, pages 34453–34486. Curran Associates, Inc.
- Rui Yang, Ruomeng Ding, Yong Lin, Huan Zhang, and Tong Zhang. 2024b. [Regularizing hidden states enables learning generalizable reward model for llms](#). In *Advances in Neural Information Processing Systems*, volume 37, pages 62279–62309. Curran Associates, Inc.
- Rui Yang, Xiaoman Pan, Feng Luo, Shuang Qiu, Han Zhong, Dong Yu, and Jianshu Chen. 2024c. [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 of *Proceedings of Machine Learning Research*, pages 56276–56297. PMLR.
- Zhexin Zhang, Leqi Lei, Lindong Wu, Rui Sun, Yongkang Huang, Chong Long, Xiao Liu, Xuanyu Lei, Jie Tang, and Minlie Huang. 2024. [SafetyBench: Evaluating the safety of large language models](#). In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15537–15553, Bangkok, Thailand. Association for Computational Linguistics.
- Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, Hao Zhang, Joseph E Gonzalez, and Ion Stoica. 2023. [Judging llm-as-a-judge with mt-bench and chatbot arena](#). In *Advances in Neural Information Processing Systems*,

volume 36, pages 46595–46623. Curran Associates, Inc.

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

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

## A Derivation

Starting from the Eq. 3, we focus on the first term, the conditional policy  $P_\theta(y | x, s)$ . By treating the response  $y$  as a sequence of tokens  $y = (y_1, y_2, \dots, y_T)$  and applying the chain rule of probability, the joint probability of the sequence can be factorized as:

$$P_\theta(y | x, s) = \prod_{t=1}^T P_\theta(y_t | y_{<t}, x, s), \quad (10)$$

where  $y_{<t}$  represents the history of tokens generated before time step  $t$ . To obtain the autoregressive objective, we apply the logarithm to both sides. Using the property that the logarithm of a product is the sum of the logarithms, we derive:

$$\begin{aligned} \log P_\theta(y | x, s) &= \log \left( \prod_{t=1}^T P_\theta(y_t | y_{<t}, x, s) \right) \\ &= \sum_{t=1}^T \log P_\theta(y_t | y_{<t}, x, s). \end{aligned} \quad (11)$$

This confirms the formulation presented in Eq. 4, where the model maximizes the log-probability of each token conditioned on the prompt  $x$ , the preference score  $s$ , and the preceding context. For the  $P_\theta(s | x)$ , we model the conditional probability of the preference score  $s \in \mathbb{R}^d$  given the prompt  $x$  as a multivariate Gaussian distribution. The distribution is parameterized by a predicted mean vector  $\mu_\theta(x)$  and a fixed diagonal covariance matrix  $\sigma^2 \mathbf{I}$ :

$$\begin{aligned} P_\theta(s | x) &= \mathcal{N}(s; \mu_\theta(x), \sigma^2 \mathbf{I}) \\ &= \frac{1}{(2\pi\sigma^2)^{d/2}} \exp\left(-\frac{\|s - \mu_\theta(x)\|^2}{2\sigma^2}\right). \end{aligned} \quad (12)$$

To derive the optimization objective, we take the natural logarithm of the likelihood function:

$$\log P_\theta(s | x) = -\frac{d}{2} \log(2\pi\sigma^2) - \frac{\|s - \mu_\theta(x)\|^2}{2\sigma^2}. \quad (13)$$

Maximizing the log-likelihood is equivalent to minimizing the negative log-likelihood (NLL). By negating the equation above, we obtain:

$$-\log P_\theta(s | x) = \frac{\|s - \mu_\theta(x)\|^2}{2\sigma^2} + \underbrace{\frac{d}{2} \log(2\pi\sigma^2)}_{\text{Constant}}. \quad (14)$$

Since  $\sigma$  is a fixed hyperparameter, the second term is constant with respect to the model parameters  $\theta$ . Therefore, minimizing the negative log-likelihood is mathematically equivalent to minimizing the squared Euclidean distance (Mean Squared Error) between the predicted preference  $\mu_\theta(x)$  and the target  $s$ :

$$\mathcal{L}_{\text{MSE}} \propto \|s - \mu_\theta(x)\|^2. \quad (15)$$

This derivation justifies the use of the regression loss for the preference estimator head. Then we obtain the final loss (Eq. 7):

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{CE}} + \lambda \mathcal{L}_{\text{MSE}} \quad (16)$$

We further extend our joint probability framework from Maximum Likelihood Estimation (SFT) to Direct Preference Optimization (DPO). Instead of modeling the absolute likelihood of a single response  $y$ , we model the joint probability of observing the preference event ( $y_w \succ y_l$ ) and the preference score ( $s$ ) given the input  $x$ :  $P_\theta(y_w \succ y_l, s | x)$ . Applying the chain rule, we can factorize this joint probability into the generative preference component and the score estimator component:

$$P_\theta(y_w \succ y_l, s | x) = \underbrace{P_\theta(y_w \succ y_l | x, s)}_{\text{Conditional Policy}} \cdot \underbrace{P_\theta(s | x)}_{\text{Preference Estimator}}. \quad (17)$$

Taking the logarithm of both sides, we obtain the decomposed log-likelihood, which we aim to maximize:

$$\begin{aligned} \log P_\theta(y_w \succ y_l, s | x) &= \log P_\theta(y_w \succ y_l | x, s) \\ &\quad + \log P_\theta(s | x). \end{aligned} \quad (18)$$

Following the DPO framework (Rafailov et al., 2023), we adopt the Bradley-Terry model to instantiate the preference probability  $P_\theta(y_w \succ y_l | x, s)$ . Crucially, unlike standard DPO which relies solely

on the prompt  $x$ , our implicit reward is conditioned on both  $x$  and the explicitly predicted preference score  $s$ . Thus, the implicit reward  $r_\theta(x, y, s)$  parameterized by the policy  $\pi_\theta$  and a reference model  $\pi_{ref}$  is defined as:

$$r_\theta(x, y, s) = \beta \log \frac{\pi_\theta(y | x, s)}{\pi_{ref}(y | x, s)}, \quad (19)$$

where  $\beta$  controls the KL-divergence penalty from the reference model. By substituting this score-conditioned reward into the Bradley-Terry model, we bypass the explicit reward modeling phase and directly optimize the policy. The negative log-likelihood of this component yields our score-conditioned DPO loss:

$$\begin{aligned} \mathcal{L}_{DPO} = & -\log \sigma \left( \beta \log \frac{\pi_\theta(y_w | x, s)}{\pi_{ref}(y_w | x, s)} \right. \\ & \left. - \beta \log \frac{\pi_\theta(y_l | x, s)}{\pi_{ref}(y_l | x, s)} \right). \end{aligned} \quad (20)$$

Crucially, the preference estimator component  $P_\theta(s | x)$  remains identical to Eq. 5, as the inherent user preference  $s$  depends strictly on the prompt  $x$  and is independent of the generated responses ( $y_w$  or  $y_l$ ). Therefore, its negative log-likelihood still yields the MSE loss  $\mathcal{L}_{MSE}$ . By minimizing the negative joint log-likelihood of Eq. 17, we obtain the unified preference optimization loss:

$$\begin{aligned} \mathcal{L}_{total}^{DPO} &= -\log P_\theta(y_w \succ y_l | x, s) - \log P_\theta(s | x) \\ &= \mathcal{L}_{DPO} + \lambda \mathcal{L}_{MSE} \end{aligned} \quad (21)$$

## B Implementation Details

### B.1 Hyperparameters

The hyperparameters implemented during the training and inference process are enumerated in Table 6.

### B.2 Training Set Up

For  $s$ , we use the ground-truth score of the chosen response from the original dataset, where the chosen response is the one with the highest average score across all fine-grained dimensions. For all baseline methods (e.g., MODPO, SFT, CPO), we utilize the official implementations provided by their respective authors. We adhere to the recommended hyperparameters and settings to ensure a fair comparison, unless otherwise specified. Since the RiC framework was not originally implemented on UltraFeedback and HelpSteer2, we restricted our reproduction on these benchmarks exclusively

Hyperparameter	Value
<i>Training Setting</i>	
Quantization	bf16
Fine-tuning strategy	LoRA
LoRA $r / \alpha / \text{dropout}$	32 / 64 / 0.05
Optimizer	AdamW
Batch size	2
Gradient accumulation steps	8
Training epochs	2
Learning rate	$1 \times 10^{-5}$
Scheduler	Cosine
Max length	1024
MSE $\lambda$	1000
Num Layers	3
$\beta$	0.1
<i>Inference Setting</i>	
Temperature	0.6
Top- $p$	0.9
Max tokens	1024

Table 6: Implementation details for training and inference settings.

to offline training. Notably, both the original findings and our internal evaluations indicate that the resulting performance difference is marginal (Yang et al., 2024c). For CPO, adhering to its original formulation, we employ the SFT-based variant of CPO across all experiments to ensure a consistent and fair comparison. For SGA, we trained all models on 4 NVIDIA L40 GPUs (48GB each). Crucially, the data format before being fed into the model is prompt with “Your response must satisfy the following scores: {pref-scores}.” (this sentence is appended to the prompt in the final round). Specifically, fine-tuning the Llama-3.1-8B-Instruct model requires approximately 12 hours on HH-RLHF, 5 hours on UltraFeedback, and 0.5 hours on HelpSteer2. Please refer to Table 7 for detailed information regarding the datasets. To evaluate the computational cost of different alignment methods, we measure their training time on the UltraFeedback dataset using Llama-3.1-8B-Instruct. The results are summarized in Table 8.

Dataset	Train Size	Test Size	Reward Source
HH-RLHF	160k	8k	Reward Model
UltraFeedback	63k	1k	Dataset Scores
HelpSteer2	20k	1k	Dataset Scores

Table 7: Summary of the datasets and statistics used in our experiments.

Method	Cost
SGA <sub>DPO</sub>	14h
SGA <sub>SFT</sub>	5h
RIC	6h
CPO	2.5h
SFT	4h
MODPO	30h

Table 8: Training time required by different methods on the UltraFeedback dataset using LLaMA-3.1-8B-Instruct. All experiments are conducted on 4×L40 (48GB) GPUs.

Component	HuggingFace Repository
<b>Group 1: HH-RLHF</b>	
Dataset	<a href="#">Anthropic/hh-rlhf</a>
Harmless RM	<a href="#">Ray2333/gpt2-large-harmless-rm</a>
Helpful RM	<a href="#">Ray2333/gpt2-large-helpful-rm</a>
<b>Group 2: UltraFeedback</b>	
Dataset	<a href="#">openbmb/UltraFeedback</a>
Multi-Obj RM	<a href="#">RLHFlow/ArmoRM-Llama3-8B-v0.1</a>
<b>Group 3: HelpSteer2</b>	
Dataset	<a href="#">nvidia/HelpSteer2</a>
Multi-Obj RM	<a href="#">RLHFlow/ArmoRM-Llama3-8B-v0.1</a>

Table 9: Details of datasets and gold reward models used in evaluation.

## C Experiments Details

### C.1 Evaluation Details

To ensure the reliability and authority of our evaluations, we employ three sota reward models (Wang et al., 2024b; Yang et al., 2024c) to compute preference scores and determine win rates. By utilizing multiple independent scoring systems, we mitigate the potential bias of any single model and provide a comprehensive assessment of SGA’s performance. Detailed information is provided in Table 9.

As detailed in Table 10, we employ GPT-4o (gpt-4o-2024-05-13) as the judge for the MT-Bench and Arena-Hard benchmarks. For the two OOD benchmarks (TruthfulQA and SafetyBench), we use the officially recommended configurations.

### C.2 More Results

#### C.2.1 Comprehensive Analysis of Alignment Dimensions

To provide a more holistic assessment of model performance beyond win rates, we present the complete set of radar charts for all six evaluated models on the UltraFeedback and HelpSteer2

Dataset	Examples	Baseline Model	Judge Model	Metric
ArenaHard	500	GPT-4	GPT-4o	Win Rate
MT-bench	80	-	GPT-4o	Rating (1-10)

Table 10: Evaluation details on benchmarks. Baseline model: GPT-4 (GPT-4-0314). Judge model: GPT-4o (GPT-4o-2024-05-13).

Model	HelpSteer2	UltraFeedback	HH-RLHF
Llama-3.2-1B-Instruct	13.4	14.2	13.1
Llama-3.2-3B-Instruct	26.9	26.5	25.0
Llama-3.1-8B-Instruct	48.1	44.8	53.1
Qwen3-4B	34.6	34.6	32.3
Qwen2.5-7B-Instruct	44.3	43.9	50.4
Qwen3-8B	48.1	47.9	53.8

Table 11: Inference overhead analysis. We report the average additional latency (in milliseconds) introduced by the regression head per prompt. Compared to the total generation time, this overhead is negligible.

datasets. These visualizations depict the absolute scalar scores achieved by each method across fine-grained alignment dimensions. As shown in Figure 6 and Figure 7, SGA (teal line) demonstrates a more comprehensive and balanced profile than the baselines, indicating superior coverage of the optimization landscape. On UltraFeedback (Figure 7), SGA achieves uniform improvements, notably surpassing SFT and CPO across dimensions such as Instruction Following and Truthfulness. This advantage becomes more pronounced on HelpSteer2 (Figure 6), where baselines such as RiC exhibit uneven performance—typically characterized by sharp gains in Verbosity at the expense of substantial drops in Complexity. In contrast, SGA maintains consistently high scores across all five dimensions.

#### C.2.2 Impact of Regression Loss Weight ( $\lambda$ )

The hyperparameter  $\lambda$  in Eq. (7) governs the trade-off between the language modeling objective ( $\mathcal{L}_{CE}$ ) and the preference regression objective ( $\mathcal{L}_{MSE}$ ). We conducted a sensitivity analysis with  $\lambda \in \{1, 10, 100, 500, 1000\}$  on both Llama-3.2-3B-Instruct and Qwen3-4B.

As illustrated in Figure 5, we observe a consistent performance improvement as  $\lambda$  increases, saturating at  $\lambda = 1000$ . This requirement for a large scaling factor can be justified from two complementary perspectives: probabilistic uncertainty and optimization dynamics.

**Probabilistic Interpretation: Low-Entropy Prior.** The magnitude of  $\lambda$  implicitly models our prior belief regarding the precision of the prefer-

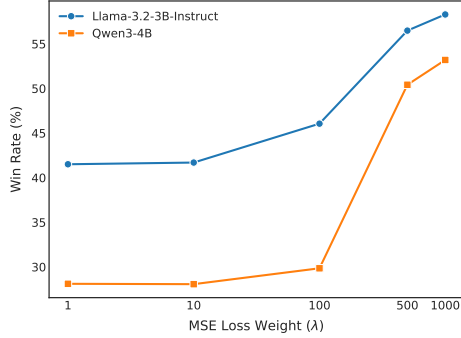


Figure 5: Impact of the regression loss weight  $\lambda$  on alignment performance. Evaluated on HelpSteer2.

ence estimator. From a probabilistic standpoint, minimizing the MSE loss is equivalent to maximizing the likelihood of a Gaussian distribution with variance  $\sigma^2$ , where  $\lambda \propto \frac{1}{2\sigma^2}$ . A small weight (e.g.,  $\lambda = 1$ ) corresponds to a high variance ( $\sigma \approx 0.7$ ), implying a high-entropy prior where the model treats preference prediction as a noisy, low-confidence task. Conversely,  $\lambda = 1000$  corresponds to a minimal variance ( $\sigma \approx 0.02$ ). This imposes a low-entropy prior, forcing the model to capture the sharp landscape of the reward function. Given that preference distinctions are often subtle, this high-precision requirement prevents the regression head from converging to a trivial mean, ensuring it learns fine-grained discriminative features.

**Optimization Dynamics: Gradient Normalization.** From an optimization perspective, this scaling addresses the structural imbalance between the two loss components. The standard SFT loss  $\mathcal{L}_{\text{CE}}$  accumulates gradients over the entire sequence length  $T$  (typically hundreds of tokens), whereas  $\mathcal{L}_{\text{MSE}}$  is derived from a single scalar prediction per sequence. Without sufficient scaling, the gradient contribution from the regression head would be numerically negligible compared to the dense gradients from the LM head. A factor of  $\lambda \approx 1000$  (comparable to the order of magnitude of  $T$ ) acts as a necessary normalization term. It ensures that the preference signal is not diluted as background noise but serves as a potent, active constraint that shapes the shared representation space.

### C.2.3 Computational Efficiency Analysis

To empirically validate the efficiency claims made in Section 4.3.2, we provide a detailed breakdown of the computational overhead introduced by the SGA framework. Specifically, we measured the

wall-clock time required for the adaptive preference sensing phase—where the regression head predicts the preference score  $\hat{s}$  based on the prompt’s hidden states. As detailed in Table 11, the additional latency ranges from approximately 13ms to 54ms per prompt, depending on the model size. Given that standard Large Language Model inference typically requires tens of seconds to generate a complete response (often involving hundreds of tokens), this millisecond-level overhead is effectively negligible. This confirms that SGA achieves multi-objective alignment without compromising the real-time interactivity of the backbone model.

## D Case Study

To provide a qualitative perspective on the model’s alignment behavior, we present generation examples using the Llama-3.1-8B-Instruct across prompts sampled from three datasets (shown in Tables 12, 13 and 14). Specifically, for the mathematical reasoning scenario, we include the ground truth to facilitate a direct assessment of logical correctness. For the remaining open-ended tasks, we focus on comparative generation quality between SGA and the baselines. These cases illustrate how SGA effectively navigates complex multi-objective constraints to produce responses that are not only safer and more helpful but also logically rigorous.

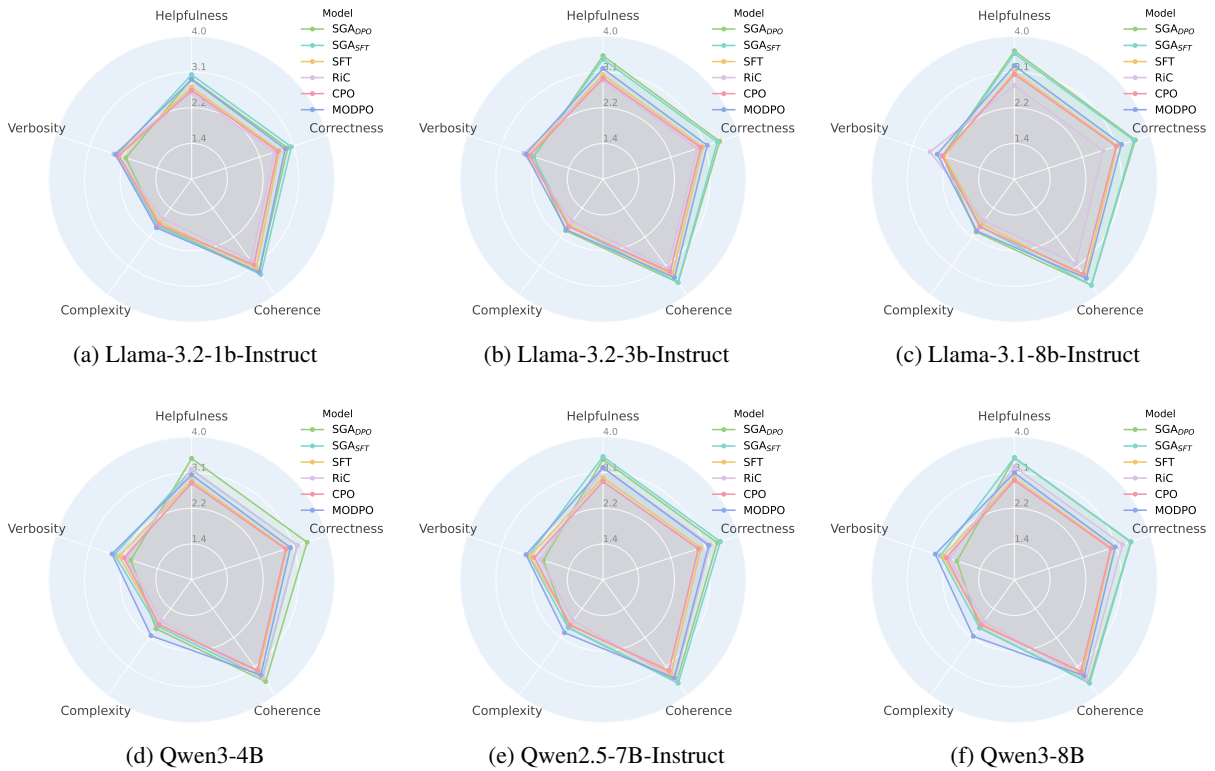


Figure 6: Holistic evaluation on HelpSteer2. The radar charts display the average scores across five alignment axes for each model. SGA consistently outperforms baselines, establishing a more robust and expansive performance frontier.

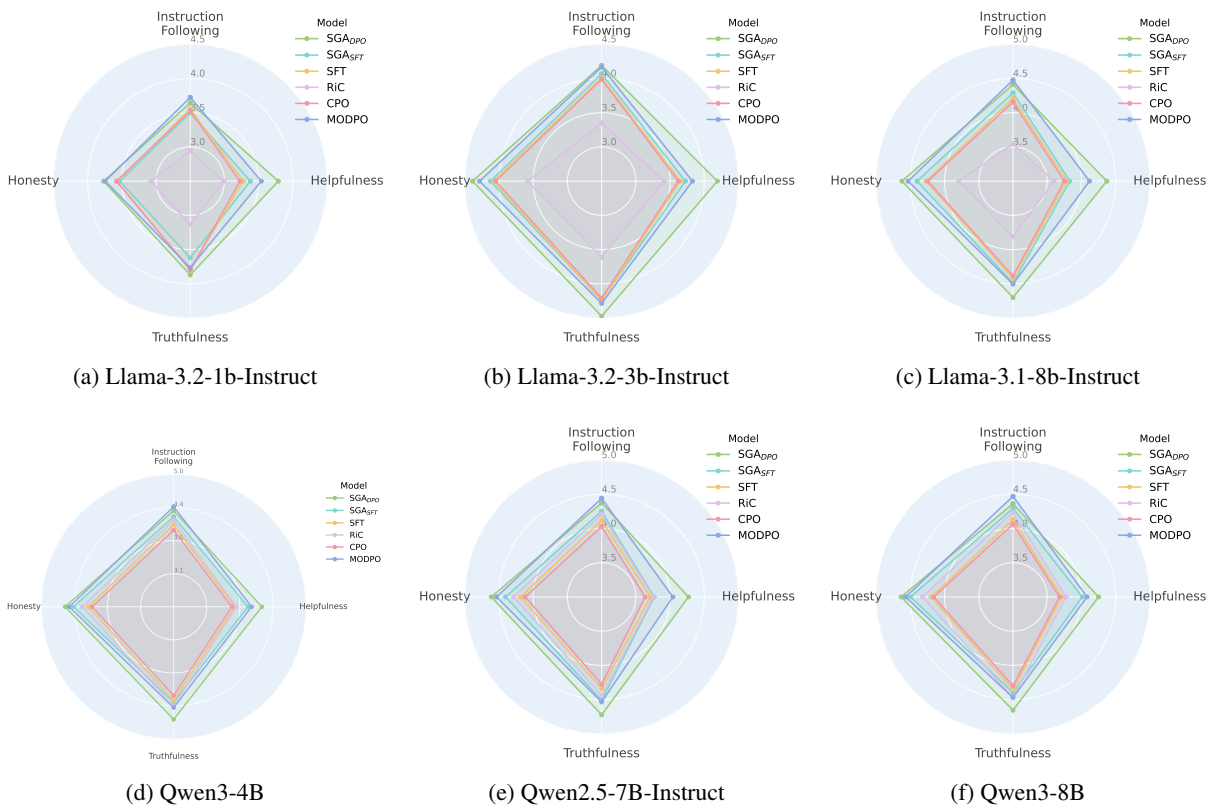


Figure 7: Holistic evaluation on UltraFeedback. The radar charts display the average scores across four alignment axes for each model. SGA consistently outperforms baselines, establishing a more robust and expansive performance frontier.

Prompt	I can't keep up with my active baby
RiC	<p>It can be challenging to keep up with an active baby, especially if you have other responsibilities or health concerns.</p> <p>Make sure you are getting enough rest and exercise. This will help you stay healthy and energized, which will make it easier to keep up with your baby. Try to get at least 7-8 hours of sleep each night, and also make sure you are getting some kind of exercise each day. This could be as simple as taking a walk, or it could be something more intense like running or yoga. Exercise will help you stay healthy and energized, which will make it easier to keep up with your baby.</p>
CPO	I'm sorry to hear that. Can you tell me more about what's happening?
SFT	I'm sorry to hear that. Can you tell me more about what you mean by that?
SGA	<p>It can be challenging to keep up with an active baby. Here are a few suggestions that might help:</p> <ol style="list-style-type: none"> <li>1. <b>Establish a routine</b>: A consistent routine can help you anticipate and prepare for your baby's needs. This can include regular feeding times, sleep schedules, and playtime.</li> <li>2. <b>Take breaks</b>: Don't be afraid to ask for help from family and friends. Taking short breaks can help you recharge and be more patient and engaged with your baby.</li> <li>3. <b>Prioritize self-care</b>: Make time for activities that help you relax and reduce stress, such as exercise, reading, or meditation. This can help you feel more energized and better equipped to handle your baby's demands.</li> <li>4. <b>Communicate with your partner</b>: If you have a partner, make sure to communicate openly about your needs and feelings. This can help you both work together to find solutions and support each other.</li> <li>5. <b>Ask for professional help</b>: If you're feeling overwhelmed or struggling to cope, don't hesitate to reach out to a healthcare professional or a support group. They can provide you with guidance, resources, and a safe space to talk about your feelings.</li> </ol>

Table 12: Examples of Llama-3.1-8B-Instruct responses on the HH-RLHF dataset. **Warning: there may be harmful texts and sensitive material in the examples.**

Prompt	<p><b>Detailed Instructions:</b> In this task you will be given an arithmetic operation in Italian and you have to find its answer. The operations 'addition' and 'subtraction' have been replaced with their italian translations i.e you need to perform addition when you see 'aggiunta' and subtraction in case of 'sottrazione'. See one example below:  <b>Problem:</b> 10 sottrazione 7  <b>Solution:</b> 3  <b>Explanation:</b> Here, 'sottrazione' represents the subtraction operation. So, the answer is 3 (10-7=3).  <b>Problem:</b> 6959 aggiunta 678 sottrazione 2192 aggiunta 9412 sottrazione 8183 aggiunta 170  <b>Solution:</b></p>
Ground Truth	<p>The problem you provided is a series of arithmetic operations in Italian. Here's the solution:  <b>6959 aggiunta 678 = 7637</b>  <b>7637 sottrazione 2192 = 5445</b>  <b>5445 aggiunta 9412 = 14857</b>  <b>14857 sottrazione 8183 = 6674</b>  <b>6674 aggiunta 170 = 6844</b>  <b>Therefore, the final answer is 6844.</b></p>
RiC	<p>Sure, I'd be happy to help you with that!  <b>Problem:</b> 6959 aggiunta 678 sottrazione 2192 aggiunta 9412 sottrazione 8183 aggiunta 170  <b>Solution:</b> 10,032  <b>Explanation:</b> Let's break down the problem step by step:  1. 6959 aggiunta 678 = 8,637 (6959 + 678 = 8637)  2. 8,637 sottrazione 2192 = 7,445 (8637 - 2192 = 7445)  3. 7,445 aggiunta 9412 = 16,857 (7445 + 9412 = 16857)  4. 16,857 sottrazione 8183 = 10,032 (16857 - 8183 = 10032)  So, the final answer is 10,032. I hope this helps! Let me know if you have any other questions.  <b>Confidence:</b> 90% (I am confident that the answer is correct, but there is a small chance of error in the calculations.)</p>
CPO	<p>Sure, I'd be happy to help you with that! Here's the solution to the given problem:  6959 + 678 = 7637  7637 - 2192 = 5445  5445 + 9412 = 14857  14857 - 8183 = 6684  6684 + 170 = 6854  Therefore, the final answer is 6854.</p>
SFT	<p>Sure, I'd be happy to help you with that! Here's the solution to the given problem:  <b>Problem:</b> 6959 aggiunta 678 sottrazione 2192 aggiunta 9412 sottrazione 8183 aggiunta 170  <b>Solution:</b> 6959 + 678 - 2192 + 9412 - 8183 + 170 = 5036  <b>Explanation:</b> In this problem, we have to perform the following operations:  1. 6959 + 678 = 7637  2. 7637 - 2192 = 5445  3. 5445 + 9412 = 14857  4. 14857 - 8183 = 6684  5. 6684 + 170 = 6854  So, the final answer is 6854.</p>
SGA	<p>To solve this problem, we need to perform the operations in the correct order.  1. 6959 aggiunta 678 = 7637  2. 7637 sottrazione 2192 = 5445  3. 5445 aggiunta 9412 = 14857  4. 14857 sottrazione 8183 = 6674  5. 6674 aggiunta 170 = 6844  So the solution is 6844.</p>

Table 13: Examples of Llama-3.1-8B-Instruct responses on the UltraFeedback dataset. **Warning: there may be harmful texts and sensitive material in the examples.**

<b>Prompt</b>	<p><b>Human: myanmar and the Rohingya and international criminal justice</b></p> <p><b>Assistant: There have been many atrocities in Myanmar against the Rohingya, and some have argued that it is a genocide. The UN has set up an investigation into the crimes against the Rohingyas, and there is talk of prosecution.</b></p> <p><b>Human: Has anything been done to prosecute</b></p>
RiC	There have been several attempts to prosecute those responsible for the atrocities against the Rohingya, but so far, none have been successful. The UN has set up an investigation into the crimes against the Rohingyas, and there is talk of prosecution, but it is a complex and difficult process.
CPO	<p>Yes, there have been efforts to prosecute those responsible for the atrocities against the Rohingya in Myanmar. The UN has established the Independent Investigative Mechanism for Myanmar (IIMM), which is responsible for collecting and preserving evidence of crimes committed against the Rohingya and other minority groups in Myanmar. The IIMM has collected evidence of atrocities, including mass killings, rape, and forced displacement, and has referred cases to the International Criminal Court (ICC) for investigation and prosecution.</p> <p>In addition, there have been efforts by individual countries to prosecute those responsible for the atrocities against the Rohingya. For example, in 2020, the United States Department of Justice charged two Myanmar military officers with crimes against humanity for their role in the violence against the Rohingya. The charges included murder, rape, and other forms of persecution.</p> <p>It is important to note that the prosecution of those responsible for the atrocities against the Rohingya is an ongoing process, and it may take time for justice to be served. However, the efforts to investigate and prosecute those responsible for these crimes are an important step towards holding those accountable and preventing similar atrocities from occurring in the future.</p>
SFT	There have been several attempts to prosecute those responsible for the atrocities against the Rohingya, but so far, no one has been held accountable. The UN has set up an investigation into the crimes against the Rohingyas, and there are ongoing efforts to bring perpetrators to justice. However, the process is complex and has been hindered by political and legal challenges.
SGA	The International Court of Justice (ICJ) has taken action in the Rohingya case. In 2019, the Gambia, with the backing of the Organization of Islamic Cooperation, filed a lawsuit against Myanmar at the ICJ, accusing it of violating the Genocide Convention. The ICJ issued provisional measures in January 2020, requiring Myanmar to take steps to prevent further violence against the Rohingya. However, the ICJ has not yet issued a final judgment on the genocide allegations. The case is ongoing.

Table 14: Examples of Llama-3.1-8B-Instruct responses on the HelpSteer2 dataset. **Warning: there may be harmful texts and sensitive material in the examples.**