Multi-Agent Collaborative Reward Design for Enhancing Reasoning in Reinforcement Learning

Anonymous ACL submission

Abstract

We present MACRM (Multi-Agent Collaborative Reward Model), a collaborative reward modeling framework that replaces a single black-box reward model with a coordinated set of specialized evaluators to improve robustness and interpretability in reinforcement learning from human feedback (RLHF). Conventional reward models struggle to simultaneously capture multiple, often competing, preference dimensions (e.g., factual correctness, helpfulness, and safety) and provide limited insight into the source of their scores. MACRM addresses these limitations by decomposing preference evaluation into domain-specific reward agents, complemented by global signals such as ranker-based preferences and embedding-based semantic similarity. A centralized aggregation mechanism fuses these heterogeneous signals into a single scalar reward compatible with standard policy optimization, balancing step-wise correctness, inter-agent agreement, and repetition penalties. Experiments on RewardBench and reasoning benchmarks such as GSM8K demonstrate that MACRM significantly improves reasoning accuracy and training stability while preserving dialogue quality and safety.

1 Introduction

Large language models (LLMs) are commonly aligned with human preferences via reinforcement learning from human feedback (RLHF) (Brown et al., 2020; Touvron et al., 2023; Achiam et al., 2023), where a learned reward model (RM) guides policy optimization. Most RLHF pipelines train an RM on preference data and collapse diverse human criteria into a single scalar objective (Stiennon et al., 2020b; Ouyang et al., 2022b). However, human preferences are intrinsically multi-dimensional (e.g., correctness, coherence, helpfulness, and safety), and scalarization often obscures trade-offs and limits interpretability. This opacity also makes debugging difficult and can exacerbate reward hacking, motivating reward modeling designs that are more structured and transparent.

We propose *Multi-Agent Collaborative Reward Model* (MACRM), a structured alternative to monolithic reward models that reframes reward evaluation as the coordination of multiple interpretable evaluators rather than the output of a single learned scalar function. MACRM decomposes preference assessment into domain-specific reward agents (e.g., factual correctness, reasoning validity, format adherence, and repetition control), each acting as an independent evaluator that produces an explicit and interpretable judgment. In addition, MACRM incorporates global evaluators that capture holistic preferences, such as ranker-based judgments and embedding-based semantic similarity, enabling richer and more transparent supervision than a single black-box reward score.

To integrate heterogeneous signals into standard RL pipelines, MACRM uses a centralized, rule-based aggregator that performs structured fusion (e.g., format gating and precision-aware shaping) to produce a single scalar reward. The policy is then optimized with advantage-based reinforcement learning (GRPO) using normalized advantages derived from the aggregated reward. We evaluate MACRM on RewardBench and reasoning benchmarks such as GSM8K, showing consistent gains in reasoning performance and training robustness while maintaining (and sometimes improving) dialogue quality and safety.

- **Collaborative reward paradigm:** We introduce MACRM, a modular alternative to monolithic reward models that decomposes preference evaluation into cooperating, interpretable reward components.
- **Structured reward aggregation:** We design a centralized, rule-based aggregator that fuses multi-dimensional signals into a single reward

compatible with standard policy-gradient optimization.

- **Empirical validation:** We provide comprehensive experiments on RewardBench and reasoning benchmarks, demonstrating improved reasoning accuracy and robustness without sacrificing fluency or safety.

2 Related Work

Reward Modeling and RLHF. Reinforcement learning from human feedback (RLHF) has become the dominant paradigm for aligning large language models (LLMs) with human preferences (Christiano et al., 2017; Stiennon et al., 2020a; Ouyang et al., 2022a; Bai et al., 2022). In this pipeline, a scalar reward model is trained on human preference data to guide policy optimization via reinforcement learning. While successful in systems such as InstructGPT (Ouyang et al., 2022a) and GPT-4 (OpenAI, 2023), scalar reward formulations suffer from limited transparency, making them vulnerable to reward hacking (Skalse et al., 2022) and misalignment under distribution shift (Coste et al., 2023).

To address these issues, prior work has explored ensemble-based reward models (Coste et al., 2023) and multi-objective reward formulations (Wang et al., 2024), which decompose feedback into interpretable dimensions (e.g., helpfulness, honesty, verbosity) and combine them through learned mixtures. However, these approaches typically rely on a single reward model with multiple output heads, limiting explicit control over how different preference dimensions interact during optimization. Another line of work improves interpretability by making reward models self-reflective, such as Critique-out-Loud (Ankner et al., 2024), which augments scalar scores with natural language critiques. In contrast, MACRM structures reward evaluation as a coordinated system of modular evaluators, enabling explicit decomposition and interaction of reward signals at training time.

Multi-Agent and Structured Evaluation. Structured evaluation and multi-agent feedback have emerged as promising alternatives to monolithic reward modeling. AI Safety via Debate (Irving et al., 2018) introduced competitive evaluation by allowing multiple models to argue while a separate evaluator selects the winner. Subsequent work has adopted multi-role or multi-agent feedback in RLAIF-style settings (Cheng et al., 2024), where agents simulate reviewers or judges from diverse perspectives. ChatEval (Chan et al., 2023) further aggregates multiple LLMs into a debating and voting panel to improve alignment with human judgments.

Unlike these approaches, which primarily operate at the evaluation stage or provide episodic feedback, MACRM integrates multiple evaluators directly into the reward modeling process used for policy optimization. Specialized reward components—such as reasoning correctness, format adherence, and repetition control—contribute structured, real-time signals during training rather than post-hoc judgments. This design complements recent advances in fine-grained reinforcement learning methods, including GRPO (Zheng et al., 2024), SPIN (Sun et al., 2023), and RAFT (Yang et al., 2023), which focus on how rewards are utilized for optimization. MACRM is orthogonal to these methods, addressing instead how reward signals are structured, decomposed, and combined.

3 Methodology

In this section, we present our *Collaborative Reward Modeling* framework, which integrates multiple structured reward components into a unified reward signal for policy optimization. Unlike conventional reinforcement learning pipelines that rely on a single monolithic scalar reward, our framework decomposes reward evaluation into modular, interpretable components that jointly assess different aspects of model outputs. The overall workflow is illustrated in Fig. 1.

3.1 Problem Formulation

We consider a learning setting in which a policy model π_θ generates a structured rollout $o = \pi_\theta(x)$ in response to an input prompt $x \sim \mathcal{D}$. Each rollout may include intermediate reasoning traces as well as a final answer. Rather than optimizing a single opaque reward function, our objective is to leverage a structured, multi-dimensional evaluation space composed of several specialized reward components.

Formally, the training objective is to optimize π_θ such that the expected aggregated reward is maxi-

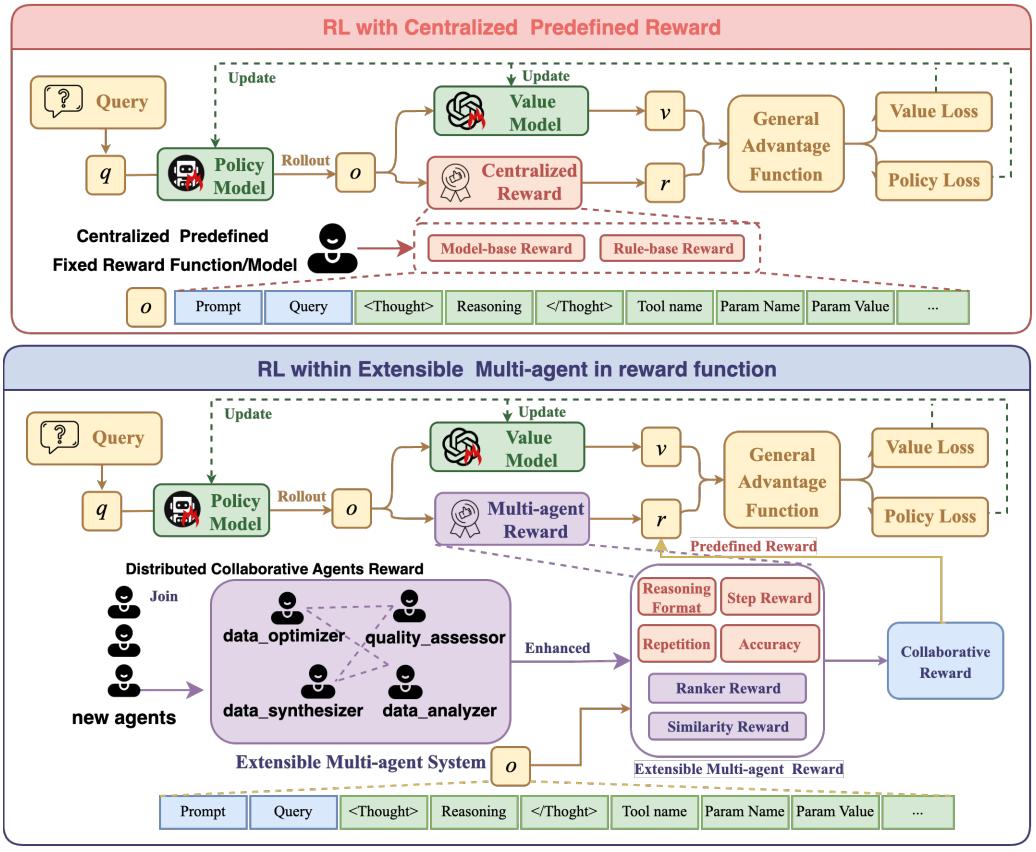

Figure 1: Architecture of MACRM. In comparison with the predefined and fixed reward function in the conventional method, MACRM leverages a multi-agent system to build an extensible intelligent reward function.

mized:

$$\max_{\theta} \mathbb{E}_{x\sim\mathcal{D}}\Big[\mathcal{F}\big(\alpha R_{\text{ranker}}(o) + \beta R_{\text{sim}}(o) + \sum_{i=1}^{K} \lambda_i R_i(o)\big)\Big] \tag{1}$$

where $R_i(o)$ denotes individual reward components evaluating different properties of the rollout, R_{ranker} and R_{sim} provide global preference signals, and $\mathcal{F}(\cdot)$ is a centralized aggregation function that maps heterogeneous reward signals to a scalar training reward. The weighting coefficients $\{\alpha, \beta, \lambda_i\}$ are fixed and specified empirically. This formulation generalizes standard RLHF by replacing a single reward model with a structured reward composition process that is interpretable and extensible.

3.2 Multi-Agent Collaborative Reward Model

MACRM implements reward evaluation as a coordinated system of multiple functional agents, each responsible for assessing a specific aspect of output quality. Instead of relying on a single monolithic reward model, these agents operate in parallel to evaluate policy rollouts from complementary perspectives and jointly produce a structured reward signal for training.

As illustrated in Fig. 2, MACRM implements a coordinated multi-agent reward system in which each agent fulfills a distinct and complementary evaluation role. Specifically, the **Data Optimizer** quantifies rollout efficiency and diversity, penalizing redundant or degenerate reasoning traces while encouraging balanced exploration; the **Quality Assessor** provides fine-grained judgments on reasoning accuracy, factual consistency, and logical coherence across intermediate steps; the **Data Synthesizer** augments supervision by injecting synthetic perturbations and incorporating external knowledge signals, thereby improving robustness and domain generalization; and the **Data Analyzer** continuously monitors statistical properties of reward signals, supporting training stability by detecting repetition patterns and distributional drift.

Each agent is implemented as a task-specific reward *evaluator* that independently processes policy rollouts and outputs an interpretable scalar signal. Importantly, these agents are *not* autonomous policies and do not learn through reinforcement or

interact through environment dynamics; coordination occurs only through centralized aggregation of their signals into a single scalar training reward. Together, the agents provide a multi-perspective characterization of output quality, enabling fine-grained, interpretable, and robust reward shaping within standard reinforcement learning pipelines.

3.3 Reward Function Design

A concrete instantiation of MACRM is realized through a set of structured reward components aligned with the evaluation dimensions commonly used in reasoning benchmarks. Each component produces an interpretable scalar signal, and together they form the basis of collaborative reward shaping.

Step-level Rewards. To encourage coherent reasoning throughout multi-step derivations, MACRM incorporates step-level rewards that assess intermediate reasoning quality. An **Outcome Reward** evaluates whether partial reasoning steps remain consistent with expected intermediate objectives, while an **Enhanced Data Reward** leverages augmented or counterfactual samples to provide stronger supervision. These signals discourage shortcuts that optimize only the final answer while neglecting reasoning validity.

Model-based Rewards. At the sequence level, MACRM employs a semantic similarity reward to capture global alignment between generated outputs and reference responses. Specifically, sentence embeddings are computed using the `all-MiniLM-L6-v2` encoder, and cosine similarity is used as the reward signal:

$$R_{\text{sim}} = \cos(\mathbf{h}_{\text{pred}}, \mathbf{h}_{\text{ref}}), \tag{2}$$

where $\mathbf{h}_{\text{pred}}$ and $\mathbf{h}_{\text{ref}}$ denote the embedding representations of the predicted and reference outputs, respectively. This reward complements discrete correctness signals by capturing semantic equivalence under surface-level variation.

Multi-dimensional Reward Components. Beyond step- and model-level rewards, MACRM incorporates additional components that encode structural and behavioral preferences. The **Accuracy Reward** (R_{acc}) validates mathematical or symbolic equivalence using tools such as `latex2sympy2` and `math_verify`. The **Format Reward** (R_{fmt}) enforces adherence to predefined reasoning templates (e.g., `<think>` and `<answer>` tags). The **Reasoning-Step Reward** (R_{step}) encourages organized and interpretable multi-step explanations. To prevent verbosity, a **Cosine-Scaled Reward** (R_{cs}) rescales accuracy scores as a function of completion length. Finally, a **Repetition Penalty** (R_{rep}) penalizes n-gram redundancy and degenerate looping behavior.

These components are combined through a fixed-weight linear composition:

$$R_{\text{collab}} = \alpha R_{\text{acc}} + \beta R_{\text{sim}} + \gamma R_{\text{fmt}} + \delta R_{\text{step}} - \eta R_{\text{rep}}, \tag{3}$$

where the coefficients $(\alpha, \beta, \gamma, \delta, \eta)$ are selected empirically to balance correctness, reasoning clarity, and linguistic fluency.

3.4 Reward Aggregation and Policy Optimization

To produce a single training signal, the collaborative reward is passed through a centralized aggregation function:

$$r_t = \mathcal{F}\big(R_{\text{collab}}(o_t), R_{\text{enhanced}}(o_t)\big), \tag{4}$$

where o_t denotes the rollout at timestep t. The aggregation function $\mathcal{F}$ applies structured, rule-based fusion, incorporating mechanisms such as format gating, precision-aware shaping, and repetition control to regularize the reward.

Policy optimization is performed using advantage-based reinforcement learning under the GRPO framework. Normalized advantages computed from the aggregated reward guide policy updates without training an explicit value function. This design enables efficient policy learning while benefiting from structured, multi-perspective reward shaping.

3.5 Discussion

MACRM reframes reward modeling as a structured composition of modular and interpretable reward components rather than a monolithic scoring oracle. By explicitly decomposing preference evaluation and combining heterogeneous signals through centralized aggregation, MACRM improves transparency and robustness in policy optimization. The modular design further allows new reward components to be incorporated with minimal changes, providing a scalable pathway toward more interpretable and extensible RLHF systems.

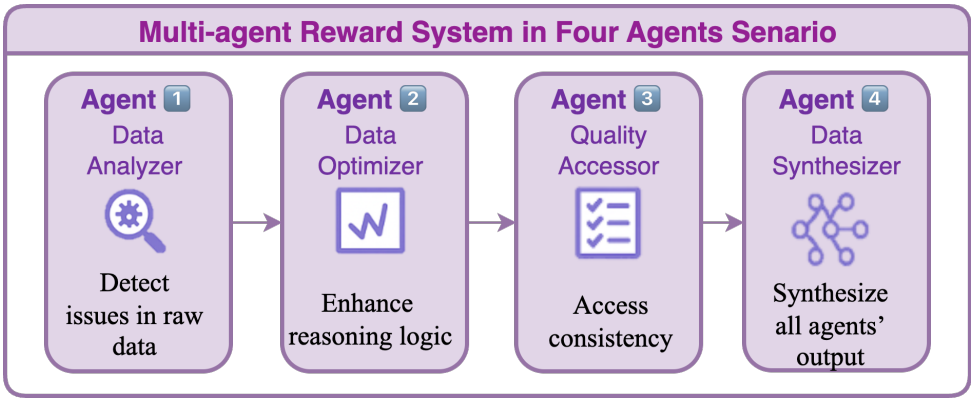

Figure 2: Decomposition of structured reward components in MACRM.

4 Experiments

4.1 Experimental Setup

We evaluate the proposed MACRM framework on RewardBench and two widely used mathematical reasoning benchmarks, GSM8K and Math. To balance evaluation fidelity with experimental efficiency, we employ backbone models at two different scales. Specifically, **Qwen2.5-7B-Instruct** is used for the overall comparison across training paradigms, where the goal is to assess the effectiveness of MACRM relative to supervised fine-tuning and alternative reward modeling baselines under realistic model capacity. In contrast, **Qwen2.5-0.5B-Instruct**, which contains approximately 494M parameters, is used for detailed analyses of agent composition and reward aggregation strategies, enabling controlled and reproducible investigation of structural design choices at a lower computational cost. Unless otherwise specified, all reinforcement learning experiments are conducted under the Generalized Reinforcement Policy Optimization (GRPO) framework, an advantage-based method that naturally supports heterogeneous reward signals and provides stable policy updates for multi-component reward shaping.

Training and evaluation are performed on the **AI-MO/NuminaMath-TIR** dataset, which consists of 3,800 training samples and 1000 held-out test samples. To facilitate fine-grained and interpretable reward evaluation, we employ a structured system prompt that enforces explicit reasoning traces enclosed within `<think>` tags and final answers enclosed within `<answer>` tags. This explicit separation between reasoning processes and final outputs allows different reward components to independently assess intermediate reasoning quality and final correctness, and ensures that improvements in reasoning-oriented metrics reflect genuine structural alignment rather than superficial answer matching.

Training Paradigm	Chat	Chat Hard	Safety	Reasoning
SFT (Math-only)	0.285	0.546	0.350	0.581
Single RM + RL	0.506	0.508	0.483	0.528
Multi-head RM + RL	0.536	**0.553**	0.539	0.525
MACRM (Ours)	**0.556**	0.520	**0.569**	**0.634**

Table 1: Overall comparison across training paradigms on RewardBench. All methods use the same policy backbone (Qwen2.5-1.5B-Instruct) and are evaluated under identical settings.

We further report training efficiency (time breakdown, memory usage, and throughput/MFU) in Appendix B, showing that MACRM introduces negligible computational overhead compared to standard RLHF pipelines.

4.2 Overall Comparison Across Training Paradigms

We first present an overall comparison across different training paradigms in Table 1. This experiment is designed to answer a fundamental question: whether the performance gains of MACRM stem merely from reinforcement learning itself, from introducing multiple reward heads, or from structured collaboration among heterogeneous reward components.

As shown in Table 1, moving from supervised fine-tuning (SFT) to reinforcement learning with preference-based rewards leads to substantial performance changes across evaluation dimensions. While SFT trained on math-only data achieves reasonable reasoning performance (0.581), it performs poorly on conversational and safety-oriented metrics, highlighting the limitations of purely supervised objectives.

Under the same RLHF framework, MACRM consistently outperforms both the single reward model and the simple multi-head reward baseline. In particular, MACRM improves the *Reasoning* score to 0.634, compared to 0.528 for Single RM + RL and 0.525 for Multi-head RM + RL, corresponding to absolute gains of 10.6 and 10.9 points, respectively. Notably, these improvements are not confined to reasoning alone. In the *Safety* dimension, MACRM achieves the highest score (0.569), exceeding both Single RM + RL (0.483) and Multi-head RM + RL (0.539).

Although multi-head reward modeling improves over a single scalar reward by exposing multiple objectives, its gains remain inconsistent, particularly for reasoning and robustness. In contrast, MACRM explicitly coordinates heterogeneous reward components through structured aggregation, enabling complementary evaluators to jointly influence policy updates. As a result, MACRM achieves stronger reasoning alignment and improved safety without sacrificing conversational quality, as reflected by competitive performance on *Chat* and *Chat Hard*.

4.3 Effect of Agent Composition

We next analyze how the composition of collaborative reward agents affects performance. Table 2 reports results under two-, three-, and four-agent configurations.

Table 2 further analyzes how the composition of collaborative reward agents affects performance. Overall, introducing the **Quality Assessor** (moving from two to three agents) improves the reward's ability to capture reasoning quality and solution validity. For the vanilla MACRM variant, the three-agent setup increases *Chat Hard* from 0.582 to 0.593, *Safety* from 0.553 to 0.587, and *Reasoning* from 0.659 to 0.672, while also improving *Math* from 0.549 to 0.621 and *GSM8K* from 59.64% to 60.74%. A similar trend is observed for the reranker-based aggregation, where adding the Quality Assessor yields the best *GSM8K* accuracy (62.87%) and the highest *Safety* score (0.618) among the rerank variants, suggesting that the additional evaluator provides complementary signals that better penalize logically inconsistent or poorly justified solutions.

Adding the **Data Synthesizer** (four agents) brings a different trade-off pattern rather than uniformly improving all metrics. In particular, MACRM (rerank) achieves the strongest *Chat* score (0.582) and the best *Math* performance (0.692) across all configurations, and its *Reasoning* score also increases substantially to 0.610, indicating that synthetic perturbations and augmented supervision can strengthen compositional reasoning and mathematical robustness. However, *GSM8K* does not strictly increase further (62.87% $\rightarrow$ 61.87%), and the vanilla MACRM scores on *Safety*/*GSM8K* revert to the two-agent level, highlighting that the benefit of the Data Synthesizer is sensitive to the aggregation strategy and may require additional tuning to consistently translate into better end-task accuracy. **Take-away:** the three-agent configuration provides the most balanced and stable gains across benchmarks, while the four-agent setup offers stronger improvements on *Math*/*Chat* but introduces noticeable trade-offs on other dimensions.

4.4 Statistical Significance Analysis

To validate that the observed improvements in Table 2 are not due to random variation, we conduct statistical significance analysis using paired non-parametric tests. All significance tests are performed against the backbone baseline (Qwen2.5-0.5B-Instruct) under the same agent configuration.

For each metric, we apply paired tests at the instance level. Accuracy-based metrics such as GSM8K are evaluated using per-instance correctness indicators, while continuous metrics from RewardBench (e.g., Chat, Safety, and Reasoning) are evaluated using per-sample score contributions. We adopt a two-sided test with a significance level of $p < 0.05$.

In Table 2, statistically significant improvements over the backbone baseline are marked with $*$.

4.5 Efficient Reward Utilization Under Disagreement

A key motivation of MACRM is to *efficiently utilize reward signals* when reward components disagree, rather than discarding valuable updates due to partial conflicts. We log the component-wise reward vector

$$\mathbf{r} = [r_{\text{acc}}, r_{\text{fmt}}, r_{\text{step}}], \tag{5}$$

and measure per-sample disagreement by the mean pairwise absolute deviation:

$$D_{\text{pair}}(\mathbf{r}) = \frac{2}{m(m-1)} \sum_{i<j} |r_i - r_j| , \tag{6}$$

where m is the number of components. For each method, we set a method-specific threshold τ as

Table 2: Effect of agent composition on RewardBench, Math, and GSM8K using Qwen2.5-0.5B-Instruct. $*$ indicates statistically significant improvement over the backbone baseline ($p < 0.05$). The backbone results are identical across all agent configurations and are reported once for clarity.

Methods	Chat	Chat Hard	Safety	Reasoning	Math	GSM8K (EM%)
Qwen2.5-0.5B-ins	0.373	0.560	0.532	0.547	0.529	48.5
Two Agents (Analyzer + Optimizer)						
MACRM	0.490*	**0.582***	0.553*	**0.659***	0.549*	59.64*
MACRM(rerank)	0.482*	0.545	**0.566***	0.423	**0.636***	60.16*
MACRM(emb)	**0.518***	0.561	0.536	0.587*	0.131	**62.33***
Three Agents (+ Quality Assessor)						
MACRM	0.497*	**0.593***	0.587*	**0.672***	0.621*	60.74*
MACRM(rerank)	0.520*	0.567	**0.618***	0.428	**0.623***	**62.87***
MACRM(emb)	**0.599***	0.532	0.570*	0.637*	0.571*	60.15*
Four Agents (+ Data Synthesizer)						
MACRM	0.570	0.557	0.553*	**0.659***	0.649*	59.64*
MACRM(rerank)	**0.582***	**0.568***	0.527	0.610*	**0.692***	61.87*
MACRM(emb)	0.579*	0.557	**0.573***	0.578*	0.652*	**62.60***

Table 3: Diagnostics configuration for disagreement analysis (Sec. 4.5).

Setting	Value
Backbone	Qwen2.5-0.5B-Instruct
Algorithm	GRPO
Steps	100
Subsampled set	$N = 256$
Max prompt / response	1024 / 2048
Reward weights	acc 1.0, fmt 0.5, step 0.5, rep 0.2
Similarity	Qwen3-Reranker-0.6B

Table 4: High- vs. low-disagreement subsets split by D_{pair} (top 20% as High-div).

Group	N	Reward	r_{acc}	r_{fmt}	r_{step}
High-div	2560	0.472	0.505	0.474	0.728
Low-div	10240	0.088	0.024	0.472	0.280

the global 80-th percentile of D_{pair} over all logged samples, and define the *conflict rate* at step t as

$$\text{conflict\_rate}(t) = \mathbb{E}\left[\mathbb{I}\left(D_{\text{pair}} > \tau\right) \mid t\right]. \quad (7)$$

Since τ is percentile-based, conflict rates are expected to concentrate near 0.2; thus we use them primarily to diagnose *temporal stability* rather than absolute cross-method differences.

Implementation details. We run a lightweight diagnostic setup and log per-sample component rewards at each step; Table 3 summarizes the key hyperparameters.

Training dynamics under disagreement. Figure 3 compares MACRM against Single RM and Multi-head. Across methods, mean reward remains stable (Fig. 3a), indicating no optimization collapse. MACRM more consistently reduces the *magnitude* of disagreement: its mean D_{pair} decreases over training (Fig. 3b), while Single RM is closer to flat and Multi-head shows a weaker decline. Moreover, MACRM exhibits a smoother conflict-rate trajectory with smaller step-to-step fluctuations in later training (Fig. 3c), suggesting improved coordination among objectives.

Disagreement concentrates on *valuable* samples. We split samples into **High-div** (top 20% by D_{pair}) and **Low-div** (remaining 80%). Table 4 shows that High-div samples have much higher r_{acc} and r_{step}, while r_{fmt} is nearly unchanged, indicating that disagreement often arises on *valuable* learning signals rather than low-quality noise. Thus, naively suppressing high-disagreement updates would waste informative rewards; MACRM instead reconciles partial conflicts so these updates can still drive optimization.

5 Conclusion

In this work, we presented MACRM, a multi-Agent collaborative reward model framework that refor-

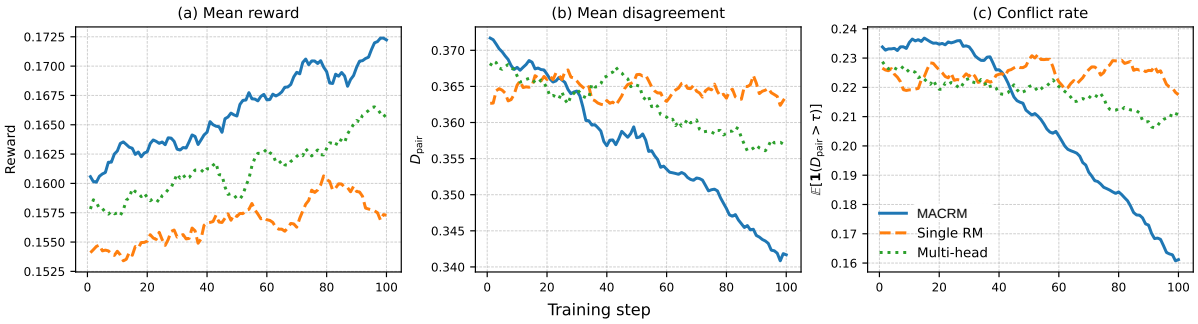

Figure 3: Disagreement diagnostics over training steps (MACRM vs. Single RM vs. Multi-head). (a) mean aggregated reward; (b) mean disagreement D_{pair}; (c) conflict rate $\mathbb{E}[\mathbb{I}(D_{\text{pair}} > \tau)]$. For each method, τ is set to its own global 80-th percentile of D_{pair}.

mulates reward learning as a structured process of coordinating multiple interpretable evaluators, rather than relying on a single black-box reward model. By decomposing preference assessment into specialized reward components and explicitly reconciling their signals through centralized aggregation, MACRM enables more transparent, robust, and controllable reward shaping for reinforcement learning from human feedback (RLHF).

Extensive experiments on RewardBench, Math, and GSM8K demonstrate that structured multi-agent collaboration yields consistent improvements in reasoning accuracy, mathematical precision, and training stability, without degrading conversational quality or safety. Moreover, our analysis shows that these gains do not arise simply from reinforcement learning or naïve multi-head reward designs, but from explicit coordination among heterogeneous evaluators. In particular, the introduction of specialized roles such as the Quality Assessor and Data Synthesizer further enhances generalization and robustness by providing complementary feedback during policy optimization.

Beyond empirical performance, MACRM offers a modular and extensible design that integrates naturally into existing RLHF pipelines. New evaluators can be added as plug-in components without modifying the core optimization procedure, making MACRM a flexible foundation for future reward modeling research. Overall, this work suggests that treating reward modeling as a system design problem—rather than purely as function approximation—provides a principled pathway toward more interpretable, self-regularizing reward systems aligned with complex, multi-dimensional human preferences.

6 Future Work

There are several promising directions for future research. One natural extension is to learn adaptive aggregation strategies that dynamically weight reward components based on task context or training dynamics, rather than relying on fixed rules.

Another direction is to expand the collaborative framework to a broader range of tasks, including open-ended generation, long-context reasoning, and multimodal alignment, to further evaluate the generality of MACRM.

Finally, integrating lightweight or learned reward agents may help reduce computational overhead, enabling scalable and efficient collaborative reward modeling for larger models and real-world RLHF systems.

7 Limitations

While MACRM demonstrates consistent improvements in reasoning accuracy and robustness, it also has several limitations. First, the current framework relies on manually designed reward components and fixed aggregation rules. Although this design improves interpretability and stability, it may limit adaptability when transferring MACRM to domains with substantially different preference structures.

Second, collaborative reward modeling introduces additional computational overhead due to the evaluation of multiple reward components. While this cost remains manageable in our experiments, it may pose challenges for large-scale deployment or low-latency settings without further optimization.

Finally, our evaluation focuses primarily on reasoning-centric benchmarks. The effectiveness of MACRM for open-ended generation tasks, long-form dialogue, or multimodal settings remains to be explored.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, and 1 others. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Zachary Ankner, Mansheej Paul, Brandon Cui, Jonathan D. Chang, and Prithviraj Ammanabrolu. 2024. Critique-out-loud reward models. *arXiv preprint arXiv:2408.11791*.

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, and 1 others. 2022. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, and 1 others. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Chi-Min Chan, Weize Chen, Yusheng Su, Jianxuan Yu, Wei Xue, Shanghang Zhang, Jie Fu, and Zhiyuan Liu. 2023. Chateval: Towards better llm-based evaluators through multi-agent debate. *arXiv preprint arXiv:2308.07201*.

Ruoxi Cheng, Haoxuan Ma, Shuirong Cao, Jiaqi Li, Aihua Pei, Zhiqiang Wang, Pengliang Ji, Haoyu Wang, and Jiaqi Huo. 2024. Reinforcement learning from multi-role debates as feedback for bias mitigation in llms. *arXiv preprint arXiv:2404.10160*.

Paul F. Christiano, Jan Leike, Tom B. Brown, Miljan Martic, Shane Legg, and Dario Amodei. 2017. Deep reinforcement learning from human preferences. In *NeurIPS*.

Thomas Coste, Usman Anwar, Robert Kirk, and David Krueger. 2023. Reward model ensembles help mitigate overoptimization. *arXiv preprint arXiv:2310.02743*.

Geoffrey Irving, Paul Christiano, and Dario Amodei. 2018. AI safety via debate. *arXiv preprint arXiv:1805.00899*.

OpenAI. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Katarina Slama, Alex Ray, John Schulman, and 1 others. 2022a. Training language models to follow instructions with human feedback. *arXiv preprint arXiv:2203.02155*.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. 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. 2022b. Training language models to follow instructions with human feedback. *arXiv preprint arXiv:2203.02155*.

Joar Skalse, Victoria Krakovna, Jonathan Uesato, Tom Everitt, and Jan Leike. 2022. Defining and characterizing reward hacking. *arXiv preprint arXiv:2209.13085*.

Nisan Stiennon, Long Ouyang, Jeffrey Wu, Daniel Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul Christiano. 2020a. Learning to summarize with human feedback. In *NeurIPS*.

Nisan Stiennon, Long Ouyang, Jeffrey Wu, Daniel Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul F Christiano. 2020b. Learning to summarize with human feedback. *Advances in neural information processing systems*, 33:3008–3021.

Jifan Sun, Fan Yang, Tianyi Tang, Zhilin Yang, Zihang Dai, and Quoc Le. 2023. Spin: Reinforcement learning from structured feedback. *arXiv preprint arXiv:2310.01280*.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, and 1 others. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Haoxiang Wang, Wei Xiong, Tengyang Xie, Han Zhao, and Tong Zhang. 2024. Interpretable preferences via multi-objective reward modeling and mixture-of-experts. *arXiv preprint arXiv:2406.12845*.

Ling Yang, Yi Liu, Jiaxin Zhang, Yuzhuo Shen, Ming Gao, Xiaozhi Yang, and 1 others. 2023. Raft: Reward alignment with feedback tree. *arXiv preprint arXiv:2312.00772*.

Yitao Zheng, Yanlin Zhu, Lei Yu, Ming Gong, Bing Liu, and Shuchang Zhou. 2024. Grpo: Guided reinforcement preference optimization for reward learning. *arXiv preprint arXiv:2405.05079*.

A Case Study on Interpretable Reward Decomposition

While the main experiments demonstrate the quantitative advantages of MACRM, we further provide a qualitative case study to illustrate how collaborative rewards are assigned at the level of individual samples. Unlike conventional reward models that output a single opaque scalar, MACRM explicitly decomposes the training signal into multiple semantically grounded components. This appendix analyzes representative examples from a small-scale diagnostic run, highlighting how different reward

Response Type	r_{acc}	r_{sim}	r_{fmt}	r_{step}	r_{rep}	Final R
Correct but verbose	1.00	1.00	0.40	0.00	0.28	0.69
Incorrect but fluent	0.00	0.27	1.00	0.00	0.00	0.25
Correct & well-structured	1.00	1.00	0.40	0.50	0.00	**0.80**
Repetitive reasoning	0.00	0.00	0.40	0.00	0.51	0.04

Table 5: Representative examples illustrating how MACRM decomposes rewards into interpretable components and aggregates them into a final training signal.

dimensions interact to shape the final optimization signal.

A.1 Diagnostic Setup

We conduct a tiny-scale MACRM run on 32 samples to facilitate detailed inspection of reward assignments. For each generated response, MACRM logs a structured reward vector $\{r_{\text{acc}}, r_{\text{sim}}, r_{\text{fmt}}, r_{\text{step}}, r_{\text{rep}}\}$, corresponding to accuracy, semantic similarity, format compliance, reasoning structure, and repetition penalty, respectively. The final training reward is computed via a transparent rule-based aggregator with fixed weights. All reward components and generated outputs are recorded at the sample level.

A.2 Discussion

Several insights emerge from Table 5. First, correctness alone is insufficient to achieve a high final reward: responses with correct answers but verbose or repetitive reasoning are explicitly penalized through the repetition component r_{rep}. Second, fluent and well-formatted but incorrect answers receive low aggregated rewards due to the accuracy component r_{acc}, despite high semantic similarity or format compliance. Third, responses that are simultaneously correct, concise, and well-structured consistently achieve high scores across all dimensions, leading to the highest aggregated rewards.

These examples demonstrate that MACRM enforces a balanced optimization objective through complementary reward dimensions. Rather than encouraging superficial patterns that exploit a single metric, MACRM aligns correctness, reasoning quality, and conciseness in a transparent and human-auditable manner. Such fine-grained reward inspection is infeasible for monolithic black-box reward models, highlighting a key interpretability advantage of the proposed framework.

Table 6: Time breakdown of a single training step.

Component	Time (s)	Ratio (%)
Sequence generation (rollout)	2.07	85.3
Reward computation (MACRM)	**0.005**	**0.2**
Actor update	0.28	11.5
Log probability computation	0.07	2.9
Others	0.01	0.1
Total	**2.43**	**100.0**

Table 7: Memory consumption during MACRM training.

Metric	Value
GPU memory allocated	63.7 GB
GPU memory reserved	66.1 GB
CPU memory usage	19.4 GB

B Efficiency Analysis

We analyze the training efficiency of MACRM from three aspects: (i) per-step time breakdown, (ii) peak memory usage, and (iii) training throughput / utilization. Overall, MACRM introduces negligible overhead: the runtime is dominated by rollout generation, while collaborative reward computation remains a rounding error in end-to-end training.

B.1 Per-Step Time Breakdown

Table 6 shows that rollout generation dominates the wall-clock time (85.3%), which is typical for on-policy RL where each step requires fresh samples from the current policy. In contrast, MACRM reward computation costs only 0.005s per step (0.2%), demonstrating that multi-agent collaborative scoring does not become a computational bottleneck.

Why the overhead is negligible. MACRM reward evaluation is performed on completed sequences and avoids storing large activation graphs. Most reward components are lightweight (e.g., rule-based checks or compact evaluators), so the dominant cost remains autoregressive decoding during rollout rather than reward computation. This suggests that improving end-to-end efficiency should prioritize faster rollout generation instead of simplifying the reward stack.

B.2 Memory Usage

Table 7 reports peak memory consumption. GPU memory dominates the footprint, primarily due to

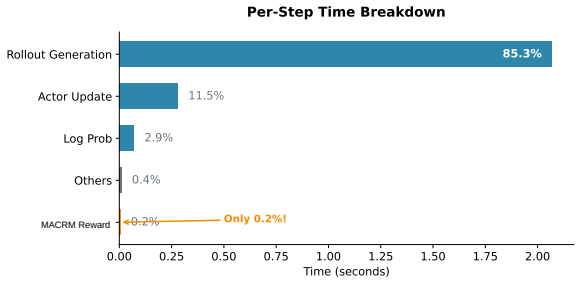

Figure 4: Per-step wall-clock time breakdown (seconds). Rollout generation dominates the step time, while MACRM reward computation contributes only 0.005s (0.2%).

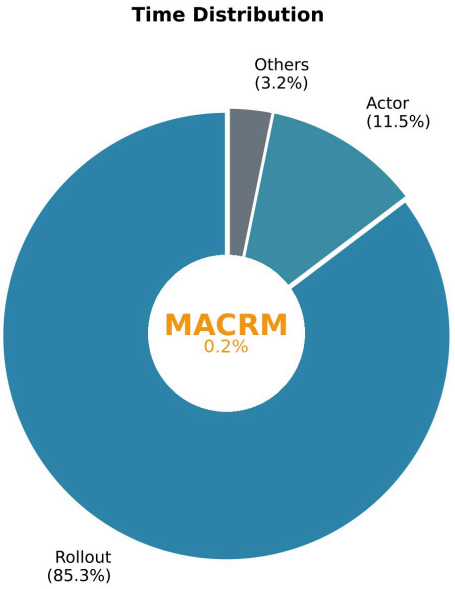

Figure 5: Per-step time ratio view. MACRM reward computation accounts for only 0.2% of total time, indicating negligible overhead.

the actor/critic states, optimizer buffers, and decoding KV caches. Importantly, MACRM does not introduce noticeable additional memory pressure, as reward computation operates on text outputs and low-dimensional signals.

B.3 Throughput and Utilization

Table 8 shows that the system sustains 527.8 tokens/s with MFU in the range of 0.33–0.56. The MFU variability across steps is commonly driven by changes in effective sequence lengths and padding efficiency, yet remains within a stable band. Together with Figure 6, these results indicate that MACRM does not cause throughput regression and that the main throughput limiter remains rollout decoding.

Table 8: Training throughput metrics.

Metric	Value
Throughput	527.8 tokens/s
Model FLOPs Utilization (MFU)	0.33 – 0.56

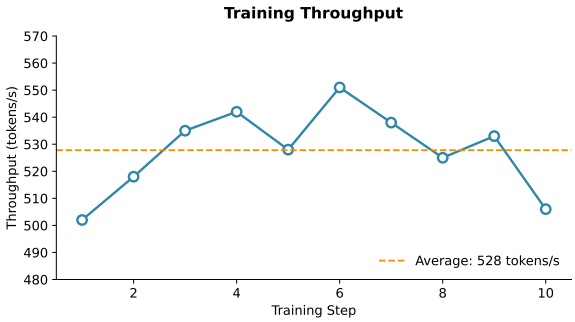

Figure 6: Training throughput (tokens/s) and MFU over training steps.

Key Finding. Across time, memory, and throughput measurements, MACRM introduces virtually no additional computational burden compared to single-RM baselines: reward evaluation accounts for only 0.2% of per-step time (Figures 4 and 5), and overall throughput remains stable (Figure 6). Therefore, MACRM improves the *quality and interpretability* of reward shaping without paying a proportional cost in end-to-end training compute.