DUAL RM: Beyond Rule-based Preference Reward Modeling via Meta-Reward

Xiaobo Liang[♠], Wanfu Wang[♠], Qipeng Huang[♠], Yuyang Ding[♠], Zecheng Tang[♠], Yixin Ji[♠], Qianben Chen, Zhe Zhao[♦], Kehai Chen[♥], Juntao Li[♠] *, Min Zhang[♣,♠]

[♠] Soochow University, [♦] Tencent Beijing Research,

[♥] Harbin Institute of Technology, Shenzhen, [♣] Key Laboratory of General Artificial Intelligence and Large Models in Provincial Universities, Soochow University

{xbliang, ljt, minzhang}@suda.edu.cn, nlpzhezhao@tencent.com

Abstract

The ability to model *sparse and underspecified rewards*, characteristic of human preferences, is fundamental to scaling Reinforcement Learning (RL). Current preference-based reward modeling largely relies on verifiable rewards, where human-annotated labels define rule-based signals. However, these methods face a fundamental bottleneck we term the ***Matryoshka Doll Problem***: *a recursive dependency where each reward verifier requires a meta-verifier*, leading to continuous and costly dependence on human annotation. In this work, we propose DUAL RM, which couples discriminative and generative reward models (DisRMs and GenRMs) under a non-parametric meta-reward. Rather than verifying the correctness of GenRM's reasoning, the meta-reward evaluates its practical impact on response quality. Specifically, GenRM identifies multi-dimensional evaluation rubrics and iteratively refines the response, while DisRM quantifies the quality shifts induced by each rubric. Furthermore, we implement rubric-based test-time scaling to improve sample efficiency and preference alignment under both DPO and GRPO [1]. Our experiments demonstrate that DUAL RM achieves strong performance across major preference benchmarks. Notably, even when trained exclusively on language modality, it exhibits robust cross-modal transfer on Omni-RewardBench.

1 Introduction

Designing learnable reward signals from environmental or external feedback remains a fundamental challenge for reinforcement learning with large language models (LLMs) (Gao et al., 2022; Dong et al., 2024; Wang et al., 2024a; Guo et al., 2025). In practice, current optimization paradigms inevitably encounter what we term the "***Matryoshka Doll Problem***": a recursive verification trap where policies rely on reward models (RMs) for verification, while the RMs themselves require meta-RMs verifiers for validation (Wu et al., 2024; Shao et al., 2025) (See Appendix A.1 for further details). This dependency chain leads to a persistent and unscalable reliance on human annotations (Liu et al., 2025). To break this cycle, we draw inspiration from human cognition, where recursive verification is resolved through grounding reward signals in the observable outcomes rather than abstract meta-verification. In other words, humans do not require a "meta-reward signal" to validate every internal thought; instead, the success or failure of an action provides the final supervision signal.

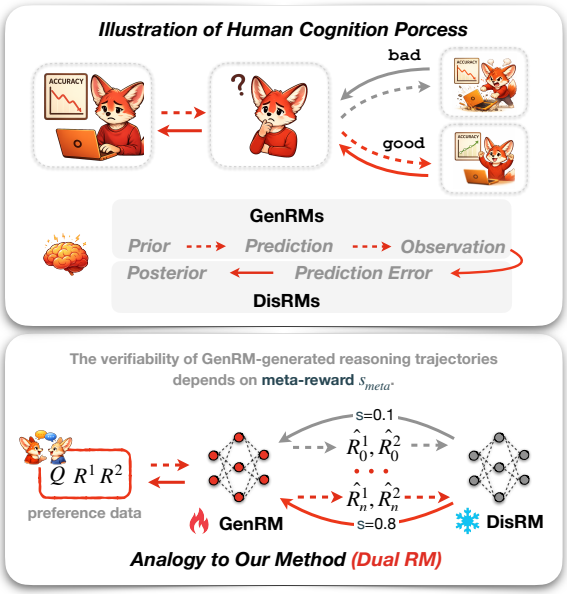

Figure 1: A comparison between human cognition and DUAL RM, both of which avoid recursive verification.

As illustrated in Figure 1, we establish meta-reward capabilities by explicitly modeling the *prior* importance of decision rubrics, analogous to the *Human Cognitive Process*[2]. We define a rubric as a query-relative evaluation dimension (Saha et al.,

* Corresponding Author

[1] https://github.com/dropreg/Tau-omni

[2] For the motivation behind introducing the human cognitive process, please refer to the Appendix A.2.

Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 37281–37296

2024; Gunjal et al., 2025), such as *logical consistency* or *factual precision*, represented by a Chain-of-Thought (CoT) generated by the LLMs. Given paired preference data, the GenRM first identifies multiple rubrics and generates corresponding judge CoTs that predict how applying a specific rubric should ideally improve a response (Mahan et al., 2024). Based on these judge CoTs, we perform targeted refinements on the responses, which serve as *predictions* of rubric-induced quality improvements. Simultaneously, the DisRM evaluates the observed response quality shifts in the refined responses (Bradley and Terry, 1952). The meta-reward is defined by the discrepancy between the GenRM's expected improvement and the DisRM's observed score change. By minimizing this *prediction error*, the GenRM updates the *prior* importance of each rubric into an approximate *posterior*. This feedback loop enables the model to identify and prioritize the most critical rubrics for judgment, grounding its internal judge CoTs in observable outcomes without external supervision.

By synergizing DisRMs and GenRMs[3], **DUAL RM** effectively leverages the complementary strengths of different RMs: GenRM enables deeper reasoning through test-time scaling (TTS) (Snell et al., 2024; Zhang et al., 2024), while DisRM provides fast and precise scoring for judgment. During implementation, we encountered several key challenges: 1) *Meta-Reward Design*: How can a meta-reward be formulated to appropriately measure the discrepancy between rubric-based predictions and empirical observations? 2) *RM Policy Optimization*: How can the learned priorities of rubrics be effectively integrated to guide policy updates? In Section 2, we conduct a series of preliminary studies to investigate the design of the meta-reward function. We identify the most effective formulation, which enables precise ranking of rubric importance. In Section 3, we present a rubric-based TTS approach that updates the policy via reinforcement learning, supporting both DPO and GRPO. To enhance exploration, we introduce a rubric planning-then-selection design that allows the model to explore diverse rubrics while refining its policy to select the most effective ones.

In experiments, we evaluate our method across multiple preference benchmarks to assess its effectiveness. On RewardBench (Lambert et al., 2024), **DUAL RM** outperforms the baseline using a single rubric reasoning, demonstrating its ability to capture critical evaluation dimensions. To investigate whether the model internalizes generalized rubrics rather than merely memorizing task-specific patterns, we extend the evaluation to Omni-RewardBench (Jin et al., 2025), and observe similarly strong performance.

Main Contributions and Takeaways

➜ **Conceptual:** We present that **DUAL RM** avoid the "*Matryoshka Doll Problem*" encountered in meta-verifier design and provides an effective **meta-reward** design for ***interpretable*** preference reward modeling.

➜ **Methodological:** We propose an RL approach for RM, **rubric-based TTS** , which can scales the model's ability to identify critical rubrics and supports both DPO and GRPO.

➜ **Empirical:** Our results demonstrate that GenRM achieves strong performance in the language modality and exhibits cross-modal generalization via rubric transfer.

2 META-REWARD DESIGN

In scenarios lacking *verifiable rewards*, meta-verifiers [4] are often designed to mitigate reward hacking during the RL process. However, existing approaches heavily rely on costly human-annotated labels or heuristic supervision signals (Wu et al., 2024; Shao et al., 2025), which inevitably introduce subtle biases and compound errors. In this section, we formalize the meta-reward as the practical impact of a reward signal on policy responses, and define meta-reward modeling as the construction of a higher-level signal for evaluating and optimizing a GenRM. To implement this, we couple GenRM's reasoning with DisRM's scoring to quantify how predicted judgments improve actual policy outputs. We further evaluate various scoring functions to identify the most robust approach for measurement.

2.1 Meta-Reward Formalization

Given a query Q and two responses R^1 and R^2, GenRM aims to generate a language-based CoT judgment J that serves as a proxy for human reasoning. In our work, each judgment J_i in a trajec-

[3]For detailed related works of DisRMs and GenRMs, please refer to the Appendix B.

[4]Meta-verifier is a concrete technical mechanism used to obtain the meta-reward. Specifically, it is a parameterized model trained to produce meta-reward signals.

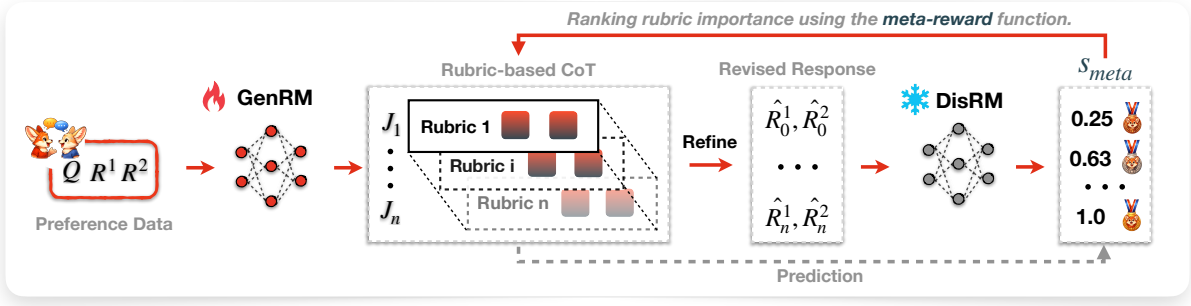

Figure 2: The **meta-reward** is computed as a scalar value by jointly leveraging the GenRM and the DisRM, which implicitly captures the importance ranking among different rubrics and their corresponding reasoning trajectories.

tory $\tau = \{Q, R^1, R^2, J_i\}_{i=1}^{N}$ is generated based on a different rubric[5] and represents a distinct CoT reasoning path. We define the meta-reward as a function that maps this trajectory τ_i to a higher-order verification signal $s_{\text{meta}}(\tau_i)$, which captures the relative preference among the candidate judgments $\{J_i\}_{i=1}^{N}$ in τ. The optimal meta-reward corresponds to the highest-scoring judgment, which also reflects the most crucial rubric. Although previous work (Liang et al., 2025) has investigated the ranking of judgments, we introduce a more precise and generalizable estimation, which can be jointly estimated using DisRM and GenRM. As shown in Figure 2, each J_i corresponding to a different rubric is used to refine the original responses R^1 and R^2, resulting in $\hat{R}_i^1$ and $\hat{R}_i^2$. The DisRM is then employed to score both the original responses, producing scores s^1 and s^2, and the refined responses, producing scores $\hat{s}_i^1$ and $\hat{s}_i^2$, respectively. Based on these scores, various meta-reward functions can be designed as follows:

1. **`Direct`**: This metric measures the absolute utility of a rubric by aggregating the discriminative scores of the refined responses. It reflects the overall response quality induced by applying rubric J_i: $s_{\text{meta}}(\tau_i) = \hat{s}_i^1 + \hat{s}_i^2$.

2. **`Relative`**: This variant emphasizes the error-correction effect of a rubric by focusing on the response originally judged as inferior:

$$s_{\text{meta}}(\tau_i) = \begin{cases} \hat{s}_i^2 - s^2, & \text{if } J_i = 0, \\ \hat{s}_i^1 - s^1, & \text{if } J_i = 1. \end{cases}$$

3. **`Consistency`**: This formulation evaluates whether the rubric-induced refinements are consistent with the original judgment:

$$s_{\text{meta}}(\tau_i) = \begin{cases} \hat{s}_2^i - s_2, & \text{if } J_i = 0 \text{ and } \hat{s}_1^i > \hat{s}_2^i, \\ \hat{s}_1^i - \hat{s}_2^i, & \text{if } J_i = 0 \text{ and } \hat{s}_1^i < \hat{s}_2^i, \\ \hat{s}_1^i - s_1, & \text{if } J_i = 1 \text{ and } \hat{s}_1^i < \hat{s}_2^i, \\ \hat{s}_2^i - \hat{s}_1^i, & \text{if } J_i = 1 \text{ and } \hat{s}_1^i > \hat{s}_2^i. \end{cases}$$

2.2 Empirical Studies

To quantitatively assess the efficacy of various meta-reward formulations, we introduce an evaluation protocol that measures rubric sensitivity in preference judgment performance. Specifically, we rank all candidate rubrics according to their s_{meta} values and select the rubric at a specific index to guide the GenRM's subsequent inference. For instance, selecting the second-ranked rubric (**index**=2) allows us to observe how second-highest s_{meta} affects final performance. A desirable meta-reward function should exhibit high discriminative power, showing a significant performance gap between high-ranked and low-ranked rubrics. Specifically, an ideal s_{meta} formulation should **maximize the performance margin between the top-tier and bottom-tier rubrics**, thereby providing more reliable supervison for policy optimization.

Results. We use `Qwen3-32B-FP8` as the GenRM and `Skywork-Reward-V2` as the DisRM to systematically evaluate the effectiveness of rubric scaling. In particular, we examine: 1) *whether rubric-based reasoning leads to more accurate preference judgments*, and 2) *whether meta-reward functions yield well-calibrated rubric importance rankings*. As shown in Figure 3, results on RewardBench[6] demonstrate that aggregating judgments from multiple rubric-based reasoning trajectories via voting leads to consistent improvements in accuracy on the Chat and Safety subsets. We also observe notable failure cases: increasing the number of votes

[5] Some prior works refer to rubrics as principles or criteria.

[6] RewardBench is a widely adopted and comprehensive benchmark for evaluating preference-based reward models.

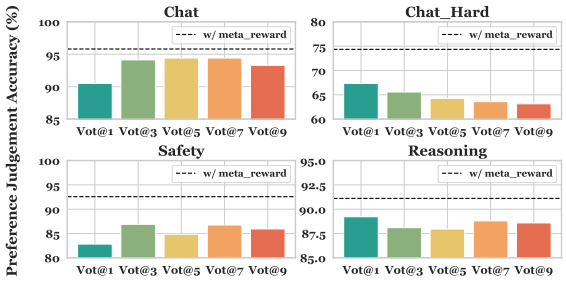

Figure 3: Performance comparison using voting over multiple rubrics on RewardBench. (*Vot@N* denotes majority voting over N rubric-based judgments. Specifically, we obtain judgments and apply majority voting to determine the result. In the empirical studies, all rubric generations are produced by the `Qwen3-32B-FP8` model as the GenRM.)

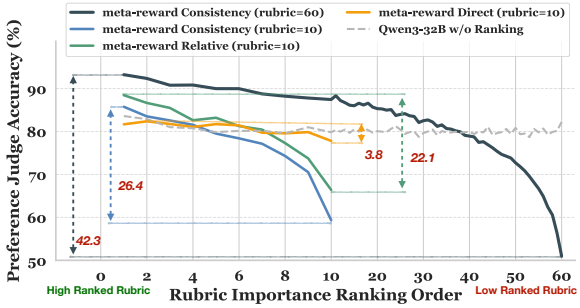

Figure 4: Performance margin of multiple rubrics ranked by different meta-reward functions. (Larger margins indicate better rubric discrimination.

does not yield further improvements and instead degrades performance on the Chat_Hard and Reasoning subsets. These observations suggest that different rubrics may capture distinct, and sometimes conflicting, evaluation dimensions, leading to instability in simple voting aggregation. To better align with human preferences, it is necessary to identify the most critical rubrics, which in turn requires a reliable measure of rubric importance.

We further evaluate the effectiveness of the meta-reward in Figure 4, which depicts the average performance across four RewardBench subsets. Each point corresponds to a single rubric. The gray curve denotes the baseline without ranking, where rubrics are applied in the model's original generation order, showing minimal performance variation. In contrast, applying the proposed ranking reveals a clear trend: top-ranked rubrics substantially improve performance, while lower-ranked ones lead to degradation. Here, the x-axis indicates the importance ranking of rubrics. We also observe that ranking rubrics according to their meta-verifier scores leads to substantial improvements, indicating that the proposed meta-reward effectively captures rubric importance.

To quantify the impact of different meta-reward designs, we also analyze the performance margin in results: 1) Introducing `Consistency` meta-reward function produces smooth and well-calibrated ranking curves, indicating that it provides a stable and reliable ordering of rubrics. 2) Expanding the number of rubrics from 10 (average accuracy = 85.76) to 60 (average accuracy = **93.23**) not only further improves performance but also yields better ranking curves and a effective rubric ordering. The primary purpose of Figure 4 is to illustrate that many critical rubrics are not prioritized or reliably generated by GenRM. Some rubrics yield strong performance, while others lead to substantially worse results. This observation motivates the need for our RL-based improvement strategy, which aims to identify and promote high-importance rubrics.

Notes: Here, we show that DisRM can estimate the quality of GenRM generated rubric without relying on an extra meta-verifier. Moreover, we find that scaling rubrics further enhances preference judgement performance.

3 Dual RM Policy Optimization

In this section, we integrate meta-rewards into our training pipeline to enhance RM performance. We first introduce rubric-based TTS, which enables continuous policy improvement through exploration of diverse rubrics. Meta-rewards are then used to score and rank candidate rubrics, allowing the model to internalize and prioritize the most effective rubric. Finally, we adapt this paradigm to two widely used RL algorithms, DPO and GRPO, demonstrating its flexibility.

3.1 Rubric-based TTS

In traditional rubric-based preference judgment, candidate rubrics are synthesized based on the RM's inherent capability. During training process, this often results in limited rubric diversity, causing the model to repeatedly exploit a limited set of rubrics and hindering further exploration. We therefore introduce a decoupled workflow that improves flexibility: As shown in Figure 5, Dual RM first generates a diverse candidate rubric set through explicit *rubric planning*, followed by a *rubric selection* process to identify the most effective rubric for the final judgment.

Under this workflow, we sample multiple rubric-based reasoning trajectories and use meta-reward

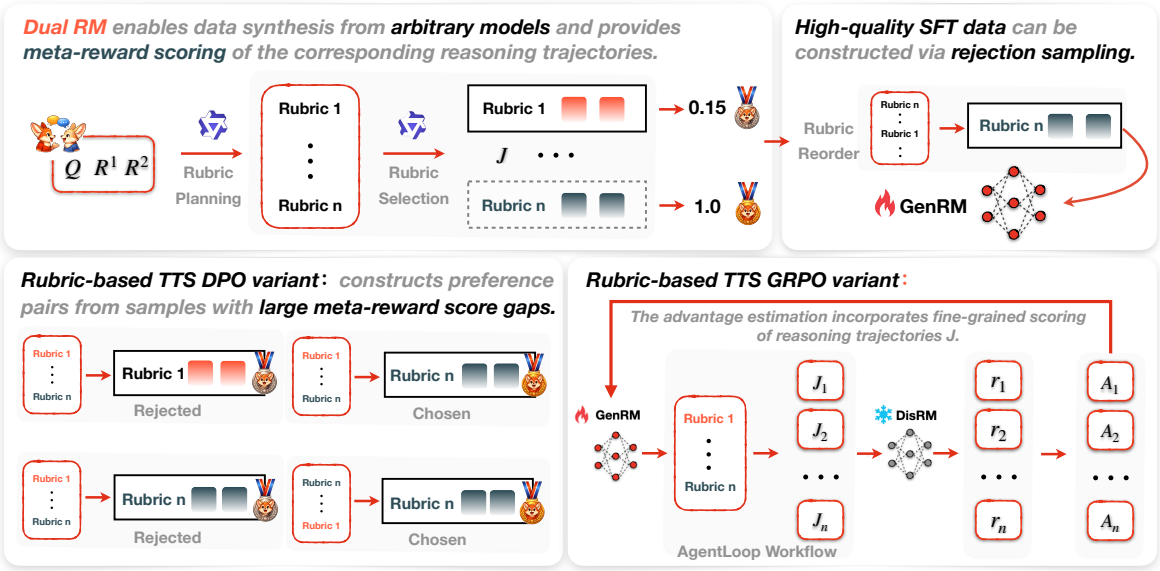

Figure 5: Overview of the DUAL RM framework. The flexibility of meta-rewards allows rubric-based TTS to be integrated with any training paradigm. It is worth noting that *rubric planning* and *selection* are introduced to decouple rubric generation, but in practice, they can be executed within a single inference pass.

function $s_{\text{meta}}(\tau_i)$ to identify the most critical rubrics. To construct a high-quality SFT dataset, we select trajectories with top meta-reward scores:

$$\mathcal{D}_{\text{SFT}} = \left\{ \tau_i \;\middle|\; \tau_i = \max_i s_{\text{meta}}(\tau_i) \right\}.$$

By prioritizing these high-scoring trajectories, the GenRM learns to focus on the most crucial rubrics. Furthermore, we adapt DUAL RM to RL paradigm, leveraging preference gaps among rubrics and negative feedback to guide the model internalize the corresponding reasoning strategies.

3.2 DPO variant

The effectiveness of DPO critically depends on the construction of high-quality preference pairs. While traditional DPO relies solely on ground-truth labels for binary comparisons, our DUAL RM leverages meta-rewards to construct fine-grained chosen-rejected pairs. Here, ϵ is a fixed threshold:

$$\mathcal{D}_{\text{DPO}} = \{(\tau_c, \tau_r) \mid s_{\text{meta}}(\tau_c) - s_{\text{meta}}(\tau_r) > \epsilon\}.$$

During multi-turn training, we construct data for different objectives at each turn: *Rubric Planning Optimization:* To refine rubric planning, we generate multiple candidate rubrics and score them using the meta-reward, enabling the GenRM to prioritize the most effective rubrics. *Rubric Judgment Optimization:* To improve the quality of the reasoning process itself, we fix the planning and selection choices and generate multiple judgment trajectories under the chosen rubric. This ensures that the GenRM receives high-quality reasoning paths.

3.3 GRPO variant

We further adapt DUAL RM to the online RL setting using GRPO. A key challenge arises because our GenRM relies on multiple rollouts to sample diverse rubric-based reasoning trajectories, while standard GRPO treats each trajectory independently. This mismatch can cause the policy to make inconsistent decisions across different rubric plans. To address this, we introduce a fixed list of rubric candidates and compute scalar rewards using the `Consistency` meta-reward. In other words, the GenRM is conditioned on this shared set of rubrics and must select the most appropriate one to guide its judgment. Standard GRPO relies on rule-based outcome rewards that only distinguish whether a trajectory's final judgment matches the ground truth. Concretely, the advantage is computed by normalizing binary rewards:

$$A_i = \frac{r_i - \text{mean}(r)}{\text{std}(r)}, \quad r = \{r_1, \ldots, r_N\},$$

where $r_i \in \{0, 1\}$ indicates whether the judgment J_i matches the ground truth, N is the group size. However, outcome-level rewards collapse the entire reasoning trajectory into a single binary signal, preventing the model from distinguishing high-quality reasoning from merely correct outcomes and resulting in coarse, misaligned credit assignment. As

Models	Size	RewardBench					RM Bench				
		Chat	Chat H.	Safety	Reason.	Avg.	Chat	Math	Code	Safety	Avg.
GPT-O4 MINI	-	95.3	81.8	91.6	98.4	91.8	77.6	93.0	80.8	93.4	86.2
DEEPSEEK-R1	671B	93.6	79.2	86.9	97.4	89.3	78.6	66.2	81.9	88.7	78.8
RISE-JUDGE-7B	7B	92.2	76.5	88.0	96.1	88.2	–	–	–	–	–
RM-R1 QWEN-7B	7B	94.1	74.6	85.2	86.7	85.2	66.6	67.0	54.6	92.6	70.2
RM-R1 QWEN-14B	7B	93.6	80.5	86.9	92.0	88.2	75.6	75.4	60.6	93.6	76.1
J1-LLAMA-8B	7B	–	–	–	–	85.7	–	–	–	–	73.4
R3-QWEN-7B	7B	91.4	73.8	85.1	90.6	85.2	66.8	82.0	65.0	87.0	75.2
R3-QWEN3-8B	7B	93.8	78.6	86.3	**96.7**	88.8	69.1	**93.2**	**75.9**	87.6	**81.4**
QWEN3-VL-8B	7B	94.3	66.8	83.7	84.1	82.2	55.7	65.3	57.4	84.3	65.7
DUAL RM: (QWEN3-VL-8B)											
SFT	8B	93.0	66.2	85.8	89.4	83.6	57.3	70.4	57.7	84.5	67.5
w/ meta-verifier	8B	**96.9**	72.4	90.4	93.6	88.3	73.2	80.2	75.2	94.3	80.7
DPO	8B	90.3	73.5	88.5	92.2	86.1	71.7	68.4	57.5	84.6	70.6
w/ meta-verifier	8B	92.4	81.7	90.2	93.2	89.4	76.7	74.5	64.9	90.2	76.6
GRPO	8B	92.2	82.0	88.3	91.1	88.4	73.0	68.0	57.5	93.0	72.9
w/ meta-verifier	8B	94.7	**84.6**	**91.3**	93.7	**91.1**	**77.9**	73.9	63.3	**94.4**	77.4

Table 1: The performance of DUAL RM on text benchmarks is evaluated using accuracy metrics. For SFT, DPO, and GRPO settings, we report performance across individual rubrics. Specifically, in the "w/ meta-verifie" configuration, a meta-reward score is employed to select the optimal rubric for each instance from a candidate set of 10 rubrics.

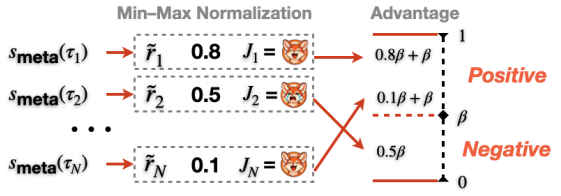

Figure 6: Advantage re-estimation in GRPO.

shown in Figure 6, we leverage the meta-reward to estimate the relative quality of reasoning trajectories and redesign the advantage computation. Given a set of trajectories $\{\tau_i\}_i^N$, we normalize the meta-reward scores using min–max normalization:

$$\tilde{r}_i = \frac{s_{\text{meta}}(\tau_i) - \min_j s_{\text{meta}}(\tau_j)}{\max_j s_{\text{meta}}(\tau_j) - \min_j s_{\text{meta}}(\tau_j)}.$$

We then incorporate ground-truth correctness by remapping the normalized scores as:

$$r_i = \begin{cases} \beta\,(\tilde{r}_i + 1), & \text{if } J_i = \text{ground truth}, \\ \beta\,\tilde{r}_i, & \text{otherwise}, \end{cases} \quad (1)$$

where β is a scaling factor, set by default to the mean of the rule-based rewards. This design preserves alignment with ground-truth correctness while introducing fine-grained priority feedback on the quality of reasoning trajectories.

4 Experiments

To validate the effectiveness of our proposed methods, we conduct experiments on widely used preference judgment benchmarks. For text-based reward modeling, we evaluate on RewardBench (Lambert et al., 2025) and RM Bench (Liu et al., 2024), while Omni-RewardBench (Jin et al., 2025) serves as a multi-modality testbed. Notably, includes human-annotated rubrics as golden preferences, making it particularly suitable for assessing a model's adaptability to diverse scenarios. For model training, we use the `Skywork-Reward-Preference-80K-v0.2`[7] dataset, which provides high-quality preference annotations. The GenRM backbone is QWEN3-VL-8B INSTRUCT, while the DisRM is instantiated as `Skywork-Reward-V2-LLaMA-3.1-8B`[8]. Across all experiments, only the GenRM is trained; the DisRM is used solely for scoring and meta-reward estimation. We synthesized the initial dataset using QWEN3-32B FP8, which was used to warm up the model. All experiments were conducted on 8 × NVIDIA A800 GPUs. The SFT experiments required approximately 2 hours, DPO experiments took around 10 hours, and GRPO experiments

[7] https://huggingface.co/datasets/Skywork/Skywork-Reward-Preference-80K-v0.2

[8] https://huggingface.co/Skywork/Skywork-Reward-V2-LLaMA-3.1-8B

Models	T2T	TI2T	TV2T	T2I	T2V	TI2I	Average
GPT-4o	86.89	75.58	77.11	69.61	73.18	73.91	76.04
GPT-4o-mini	87.43	74.65	77.89	67.80	74.89	66.67	74.89
Qwen2.5-VL-7B-Instruct (Bai et al., 2025)	80.87	66.28	78.95	65.53	64.59	50.72	67.82
InternVL2.5-8B (Chen et al., 2024)	72.13	64.88	65.00	64.40	61.59	53.14	63.52
InternVL3-8B (Zhu et al., 2025)	84.70	71.63	76.84	69.39	65.67	53.62	70.32
Unified Reward (Wang et al., 2025b)	68.58	59.77	79.47	68.93	**79.83**	46.86	67.24
Unified Reward 1.5 (Wang et al., 2025a)	67.76	67.39	78.68	67.57	78.97	50.72	68.52
Omni-RewardModel R1 (Jin et al., 2025)	81.77	69.53	75.53	**71.20**	62.02	55.56	69.26
Omni-RewardModel BT (Jin et al., 2025)	**85.79**	**72.79**	**79.47**	67.12	72.75	65.70	73.93
Dual RM: (Qwen3-VL-8B)							
Qwen3-VL-8B	75.95	70.00	77.77	66.80	74.72	67.81	72.18
SFT	78.51	66.11	76.05	67.73	74.62	57.35	70.06
GRPO	83.88	70.46	76.57	70.68	77.41	**68.11**	**74.51**

Table 2: Comparison of model performance on Omni-RewardBench across 6 multimodal subsets. Here, **I** denotes Image and **V** denotes Video. For example, TextV2Text indicates that the user query contains both text and video, while the model response is text only. (All results are evaluated against the human-annotated golden rubrics.)

required about 30 hours. Detailed training and inference setups, model hyperparameters, and prompts are provided in Appendix C.

4.1 Main Results

As summarized in Table 1, Dual RM consistently benefits from training, achieving substantial gains across SFT, DPO, and GRPO paradigms. Notably, Dual RM achieves a competitive preference judgment accuracy of 88.4 using only a single rubric. In particular, it outperforms previous GRPO baselines, such as J1-LLaMA (Whitehouse et al., 2025) and RM-R1 (Chen et al., 2025a) with 7B backbones, highlighting the effectiveness of rubric-based meta-reward optimization. We further investigate whether using a meta-verifier to rank the generated rubrics can help identify the most effective rubric and thereby improve model performance. The results indicate that incorporating meta-reward further improves GRPO performance by 2.7 points, achieving **91.1**, which is on par with the closed-source model GPT-4o Mini.

Despite these gains, model convergence remains sensitive to training data distribution. Performance on RewardBench's Chat Hard subset improves, as it favors shorter responses, but this shift negatively impacts the Chat and Reasoning subsets, which prefer longer outputs. A detailed analysis of these effects is provided in Section 7.

4.2 Learning Generalization from Rubrics

We further investigate whether our method can acquire rubric-based judgment capabilities and, more importantly, whether these capabilities generalize across diverse scenarios. To this end, we evaluate the model on Omni-RewardBench, a multimodal preference benchmark featuring human-curated rubrics. Notably, our model is trained exclusively on the language modality of the Omni dataset. To enhance the quality of reasoning trajectories during GRPO training, we replace generated rubrics with gold-standard rubrics in the planning phase. At test time, we also use human annotated golden rubrics for preference evaluation, which serves as the default and fair evaluation protocol in the benchmark.

As shown in Table 2, evaluation on 6 unseen subsets across multiple modalities reveals that Dual RM achieves an average improvement of 4 points (from 70.06 to 74.51). Notably, Dual RM also outperforms Omni-RewardModel R1, which was trained with multi-modality dataset. These results demonstrate that our method enables GenRMs to generalize rubric-based reasoning far beyond their initial modality.

5 Analysis

5.1 Does the Choice of DisRM Affect the Final Performance of GenRM?

In previous experiments, we employed a strong DisRM as the meta verifier to help GenRM select the best rubric. However, it remains unclear

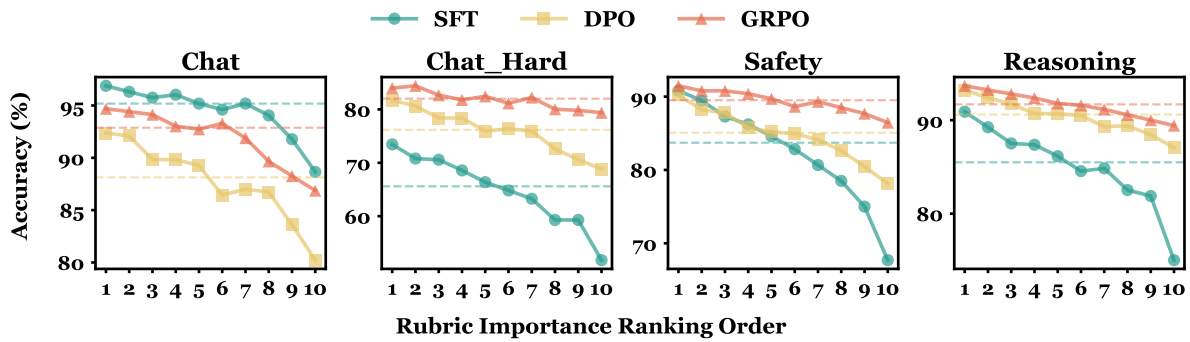

Figure 7: Comparison of GenRM performance on RewardBench with 10 rubrics generated by different models.

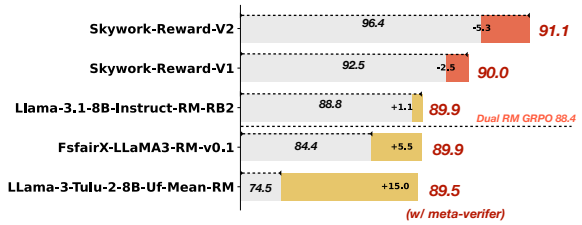

Figure 8: Performance comparison on Reward Bench using different DisRMs as meta-verifiers.

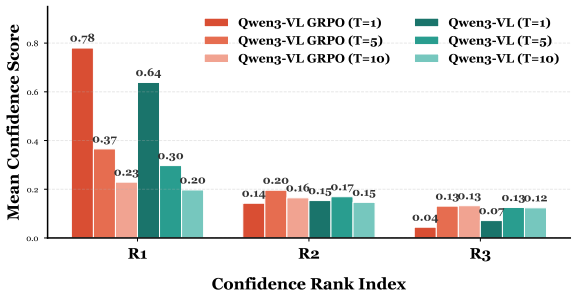

Figure 9: Meta-Reward scores on RewardBench can be interpreted as confidence for each rubric, and the weighting can be adjusted by tuning the temperature parameter (e.g., T=1, 5, 10) to smooth the distribution.

whether the observed improvements in GenRM performance are driven by the inherent capabilities of the DisRM, or if they depend entirely on it. To investigate this, we systematically evaluate the impact of using DisRMs with varying levels of capability as meta-verifiers for ranking GenRM outputs. As shown in Figure 8, even the weakest DisRM, `LLaMA-3-Tulu-2-8B-UF-Mean-RM`, which achieves only 74.5 accuracy on RewardBench, is able to improve the GenRM performance from **88.4** to **89.5** (**15** point gain relative to its own score). Stronger DisRMs further enhance ranking consistency, leading to improved top-ranked GenRM outputs. More detailed numerical results from the experiments can be found in the Appendix C.2. These findings indicate that DUAL RM effectively leverages DisRMs of varying quality to extend the reasoning capabilities of GenRM.

5.2 Does the DUAL RM Learn to Generate Crucial Rubrics?

We further investigate whether the trained model can generate more critical rubrics. To this end, we compare the SFT, DPO, and GRPO models by having each generate 10 candidate rubrics, which are then ranked by a meta-verifier. As shown in Figure 7, the GRPO model consistently produces more accurate top-ranked rubrics compared to models trained with other methods. Additionally, the median evaluation scores of rubrics generated by GRPO surpass those of the other approaches. These results show that GRPO not only improves the quality of the top-ranked rubrics, but also enhances the overall judgment ability across all generated rubrics. This suggests that our approach enables the model to continually internalize new rubrics through exploration, thereby strengthening its rubric-based reasoning capabilities.

5.3 Meta-Reward as a Measure of Confidence

Meta-reward scores can be interpreted as a form of confidence for model judgments. Given multiple candidate rubrics, we normalize their meta-reward scores using a softmax function to obtain confidence for each rubric: higher scores indicate that the model assigns greater importance to the corresponding rubric in guiding its reasoning. As illustrated in Figure 9, after GRPO training, the model's confidence in the top-ranked rubric (R1) increases substantially, indicating that it has learned to prioritize the most effective evaluative rubric.

6 Related Works

Current reward modeling paradigms for meta-verifiers (Wu et al., 2025; Li et al., 2025) rely heavily on human-designed priors, which hinders their

scalability across diverse tasks and domains. To address this, recent efforts have integrated DisRM and GenRM to introduce meta-level supervision signals beyond scalar rewards. Specifically, Ankner et al. (2024) jointly optimize generative and discriminative objectives by leveraging reasoning rationales to improve scoring accuracy. Similarly, Yu et al. (2025) utilize self-generated critiques as explicit evidence to improve scalar reward predictions. However, these self-rewarding approaches (Yuan et al., 2024b; Prasad et al., 2024) remain bottlenecked by the model's inherent reasoning limits or consistency constraints. In contrast, we propose DUAL RM, which synergizes DisRM and GenRM to exploit their inherent complementarities: while DisRM provides robust ranking, GenRM offers interpretable, fine-grained feedback.

7 Conclusion

DUAL RM goes beyond rule-based preference reward modeling and introduces a flexible paradigm with several key advantages: First, it improves the scalability of meta-verification by reducing reliance on human designed rules, allowing task-specific policy optimization and reward modeling to be jointly guided by preference data; Second, it is training-agnostic and can be seamlessly integrated with mainstream paradigms such as SFT, DPO, and GRPO, with meta-rewards providing adaptive supervision; Third, it offers rubric-level interpretability for its judgments, providing actionable insights for analyzing model behavior and identifying potential data distribution issues. Overall, DUAL RM substantially reduces human supervision costs while enabling continual model improvement.

Limitations

DUAL RM is the first work to leverage both DisRM and GenRM to address the recursive dependency problem in meta-verifier design. Despite its effectiveness, several limitations remain in practice:

- **Dependence on DisRM capability**: The performance of meta-rewards is influenced by the capability of the DisRM. Using a weak DisRM may lead to unreliable supervision signals, as DisRMs excel at in-distribution reward modeling but often fail under out-of-distribution evaluation scenarios.
- **Sensitivity to training data distribution**:GenRM training is highly sensitive to the underlying data distribution, meaning that the distribution of training and inference data should be consistent to ensure reliable performance. In our experiments, using only the `Skywork-Reward-Preference-80K-v0.2` dataset limited performance on RM Bench. Achieving robust results across multiple benchmarks typically requires careful dataset balancing, consistent with best practices in recent studies (Muennighoff et al., 2025; Ye et al., 2025).

Due to these factors, DUAL RM may not perform optimally in certain scenarios. However, it is important to note that in real-world applications, building a data flywheel is essential. Preference data should be continuously collected to correct and enhance model performance. DUAL RM offers a practical mechanism to sustain a data flywheel: as more preference data is collected, the DisRM becomes stronger, which in turn further enhances GenRM, enabling continuous improvement in performance.

Acknowledgments

We sincerely thank the anonymous reviewers for their valuable feedback. This work was supported by the CCF–Tencent Rhino-Bird Open Research Fund, the National Natural Science Foundation of China (NSFC No. 62576232), and the Key Laboratory of General Artificial Intelligence and Large Models in Provincial Universities, Soochow University.

References

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

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, and 1 others. 2025. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*.

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

Ralph Allan Bradley and Milton E Terry. 1952. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39(3/4):324–345.

Maosong Cao, Alexander Lam, Haodong Duan, Hongwei Liu, Songyang Zhang, and Kai Chen. 2024. Compassjudger-1: All-in-one judge model helps model evaluation and evolution. *arXiv preprint arXiv:2410.16256*.

Xiusi Chen, Gaotang Li, Ziqi Wang, Bowen Jin, Cheng Qian, Yu Wang, Hongru Wang, Yu Zhang, Denghui Zhang, Tong Zhang, and 1 others. 2025a. Rm-r1: Reward modeling as reasoning. *arXiv preprint arXiv:2505.02387*.

Yutong Chen, Jiandong Gao, and Ji Wu. 2025b. Towards revealing the effectiveness of small-scale fine-tuning in r1-style reinforcement learning. *arXiv preprint arXiv:2505.17988*.

Zhe Chen, Weiyun Wang, Yue Cao, Yangzhou Liu, Zhangwei Gao, Erfei Cui, Jinguo Zhu, Shenglong Ye, Hao Tian, Zhaoyang Liu, and 1 others. 2024. Expanding performance boundaries of open-source multimodal models with model, data, and test-time scaling. *arXiv preprint arXiv:2412.05271*.

Hanze Dong, Wei Xiong, Bo Pang, Haoxiang Wang, Han Zhao, Yingbo Zhou, Nan Jiang, Doyen Sahoo, Caiming Xiong, and Tong Zhang. 2024. Rlhf workflow: From reward modeling to online rlhf. *arXiv preprint arXiv:2405.07863*.

Bofei Gao, Zefan Cai, Runxin Xu, Peiyi Wang, Ce Zheng, Runji Lin, Keming Lu, Dayiheng Liu, Chang Zhou, Wen Xiao, and 1 others. 2024. Llm critics help catch bugs in mathematics: Towards a better mathematical verifier with natural language feedback. *arXiv preprint arXiv:2406.14024*.

Leo Gao, John Schulman, and Jacob Hilton. 2022. Scaling laws for reward model overoptimization. *Preprint*, arXiv:2210.10760.

Anisha Gunjal, Anthony Wang, Elaine Lau, Vaskar Nath, Yunzhong He, Bing Liu, and Sean Hendryx. 2025. Rubrics as rewards: Reinforcement learning beyond verifiable domains. *arXiv preprint arXiv:2507.17746*.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, and 1 others. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*.

Yanping Huang and Rajesh PN Rao. 2011. Predictive coding. *Wiley Interdisciplinary Reviews: Cognitive Science*, 2(5):580–593.

Zhuoran Jin, Hongbang Yuan, Kejian Zhu, Jiachun Li, Pengfei Cao, Yubo Chen, Kang Liu, and Jun Zhao. 2025. Omni-reward: Towards generalist omni-modal reward modeling with free-form preferences. *arXiv preprint arXiv:2510.23451*.

Nathan Lambert, Valentina Pyatkin, Jacob Morrison, Lester James Validad Miranda, Bill Yuchen Lin, Khyathi Chandu, Nouha Dziri, Sachin Kumar, Tom Zick, Yejin Choi, and 1 others. 2025. Rewardbench: Evaluating reward models for language modeling. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 1755–1797.

Nathan Lambert, Valentina Pyatkin, Jacob Morrison, LJ Miranda, Bill Yuchen Lin, Khyathi Chandu, Nouha Dziri, Sachin Kumar, Tom Zick, Yejin Choi, Noah A. Smith, and Hannaneh Hajishirzi. 2024. Rewardbench: Evaluating reward models for language modeling. *Preprint*, arXiv:2403.13787.

Yuran Li, Jama Hussein Mohamud, Chongren Sun, Di Wu, and Benoit Boulet. 2025. Leveraging llms as meta-judges: A multi-agent framework for evaluating llm judgments. *arXiv preprint arXiv:2504.17087*.

Xiaobo Liang, Haoke Zhang, Juntao Li, Kehai Chen, Qiaoming Zhu, and Min Zhang. 2025. Generative reward modeling via synthetic criteria preference learning. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 26755–26769, Vienna, Austria. Association for Computational Linguistics.

Chris Yuhao Liu, Liang Zeng, Yuzhen Xiao, Jujie He, Jiacai Liu, Chaojie Wang, Rui Yan, Wei Shen, Fuxiang Zhang, Jiacheng Xu, and 1 others. 2025. Skywork-reward-v2: Scaling preference data curation via human-ai synergy. *arXiv preprint arXiv:2507.01352*.

Yantao Liu, Zijun Yao, Rui Min, Yixin Cao, Lei Hou, and Juanzi Li. 2024. Rm-bench: Benchmarking reward models of language models with subtlety and style. *arXiv preprint arXiv:2410.16184*.

Dakota Mahan, Duy Van Phung, Rafael Rafailov, Chase Blagden, Nathan Lile, Louis Castricato, Jan-Philipp Fränken, Chelsea Finn, and Alon Albalak. 2024. Generative reward models. *arXiv preprint arXiv:2410.12832*.

Beren Millidge, Anil Seth, and Christopher L Buckley. 2021. Predictive coding: a theoretical and experimental review. *arXiv preprint arXiv:2107.12979*.

Niklas Muennighoff, Zitong Yang, Weijia Shi, Xiang Lisa Li, Li Fei-Fei, Hannaneh Hajishirzi, Luke Zettlemoyer, Percy Liang, Emmanuel Candès, and Tatsunori B Hashimoto. 2025. s1: Simple test-time scaling. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 20286–20332.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, and 1 others. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744.

Archiki Prasad, Weizhe Yuan, Richard Yuanzhe Pang, Jing Xu, Maryam Fazel-Zarandi, Mohit Bansal, Sainbayar Sukhbaatar, Jason Weston, and Jane Yu. 2024. Self-consistency preference optimization. *arXiv preprint arXiv:2411.04109*.

Swarnadeep Saha, Omer Levy, Asli Celikyilmaz, Mohit Bansal, Jason Weston, and Xian Li. 2024. Branch-solve-merge improves large language model evaluation and generation. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 8352–8370.

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

Zhihong Shao, Yuxiang Luo, Chengda Lu, ZZ Ren, Jiewen Hu, Tian Ye, Zhibin Gou, Shirong Ma, and Xiaokang Zhang. 2025. Deepseekmath-v2: Towards self-verifiable mathematical reasoning. *arXiv preprint arXiv:2511.22570*.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, and 1 others. 2024. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*.

Guangming Sheng, Chi Zhang, Zilingfeng Ye, Xibin Wu, Wang Zhang, Ru Zhang, Yanghua Peng, Haibin Lin, and Chuan Wu. 2024. Hybridflow: A flexible and efficient rlhf framework. *arXiv preprint arXiv: 2409.19256*.

Charlie Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. 2024. Scaling llm test-time compute optimally can be more effective than scaling model parameters. *arXiv preprint arXiv:2408.03314*.

Hao Sun, Yunyi Shen, and Jean-Francois Ton. 2025. Rethinking bradley-terry models in preference-based reward modeling: Foundations, theory, and alternatives. *Preprint*, arXiv:2411.04991.

Richard S Sutton, Andrew G Barto, and 1 others. 1998. *Reinforcement learning: An introduction*, volume 1. MIT press Cambridge.

Binghai Wang, Rui Zheng, Lu Chen, Yan Liu, Shihan Dou, Caishuang Huang, Wei Shen, Senjie Jin, Enyu Zhou, Chenyu Shi, and 1 others. 2024a. Secrets of rlhf in large language models part ii: Reward modeling. *arXiv preprint arXiv:2401.06080*.

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

Yibin Wang, Zhimin Li, Yuhang Zang, Chunyu Wang, Qinglin Lu, Cheng Jin, and Jiaqi Wang. 2025a. Unified multimodal chain-of-thought reward model through reinforcement fine-tuning. *arXiv preprint arXiv:2505.03318*.

Yibin Wang, Yuhang Zang, Hao Li, Cheng Jin, and Jiaqi Wang. 2025b. Unified reward model for multimodal understanding and generation. *arXiv preprint arXiv:2503.05236*.

Chenxi Whitehouse, Tianlu Wang, Ping Yu, Xian Li, Jason Weston, Ilia Kulikov, and Swarnadeep Saha. 2025. J1: Incentivizing thinking in llm-as-a-judge via reinforcement learning. *arXiv preprint arXiv:2505.10320*.

Tianhao Wu, Weizhe Yuan, Olga Golovneva, Jing Xu, Yuandong Tian, Jiantao Jiao, Jason Weston, and Sainbayar Sukhbaatar. 2024. Meta-rewarding language models: Self-improving alignment with llm-as-a-meta-judge. *Preprint*, arXiv:2407.19594.

Tianhao Wu, Weizhe Yuan, Olga Golovneva, Jing Xu, Yuandong Tian, Jiantao Jiao, Jason E Weston, and Sainbayar Sukhbaatar. 2025. Meta-rewarding language models: Self-improving alignment with llm-as-a-meta-judge. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 11548–11565.

Rui Yang, Ruomeng Ding, Yong Lin, Huan Zhang, and Tong Zhang. 2024. Regularizing hidden states enables learning generalizable reward model for llms. *Advances in Neural Information Processing Systems*, 37:62279–62309.

Yixin Ye, Zhen Huang, Yang Xiao, Ethan Chern, Shijie Xia, and Pengfei Liu. 2025. Limo: Less is more for reasoning. *arXiv preprint arXiv:2502.03387*.

Ziyi Ye, Xiangsheng Li, Qiuchi Li, Qingyao Ai, Yujia Zhou, Wei Shen, Dong Yan, and Yiqun Liu. 2024. Beyond scalar reward model: Learning generative judge from preference data. *arXiv preprint arXiv:2410.03742*.

Yue Yu, Zhengxing Chen, Aston Zhang, Liang Tan, Chenguang Zhu, Richard Yuanzhe Pang, Yundi Qian, Xuewei Wang, Suchin Gururangan, Chao Zhang, Melanie Kambadur, Dhruv Mahajan, and Rui Hou. 2025. Self-generated critiques boost reward modeling for language models. *Preprint*, arXiv:2411.16646.

Lifan Yuan, Ganqu Cui, Hanbin Wang, Ning Ding, Xingyao Wang, Jia Deng, Boji Shan, Huimin Chen, Ruobing Xie, Yankai Lin, and 1 others. 2024a. Advancing llm reasoning generalists with preference trees. *arXiv preprint arXiv:2404.02078*.

Weizhe Yuan, Richard Yuanzhe Pang, Kyunghyun Cho, Xian Li, Sainbayar Sukhbaatar, Jing Xu, and Jason E Weston. 2024b. Self-rewarding language models. In *Forty-first International Conference on Machine Learning*.

Lunjun Zhang, Arian Hosseini, Hritik Bansal, Mehran Kazemi, Aviral Kumar, and Rishabh Agarwal. 2024. Generative verifiers: Reward modeling as next-token prediction. *arXiv preprint arXiv:2408.15240*.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, and 1 others. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in neural information processing systems*, 36:46595–46623.

Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, Zheyan Luo, Zhangchi Feng, and Yongqiang Ma. 2024. Llamafactory: Unified efficient fine-tuning of 100+ language models. *arXiv preprint arXiv:2403.13372*.

Jinguo Zhu, Weiyun Wang, Zhe Chen, Zhaoyang Liu, Shenglong Ye, Lixin Gu, Hao Tian, Yuchen Duan, Weijie Su, Jie Shao, and 1 others. 2025. Internvl3: Exploring advanced training and test-time recipes for open-source multimodal models. *arXiv preprint arXiv:2504.10479*.

A Preliminary

A.1 The Porblem of "*Matryoshka Doll*"

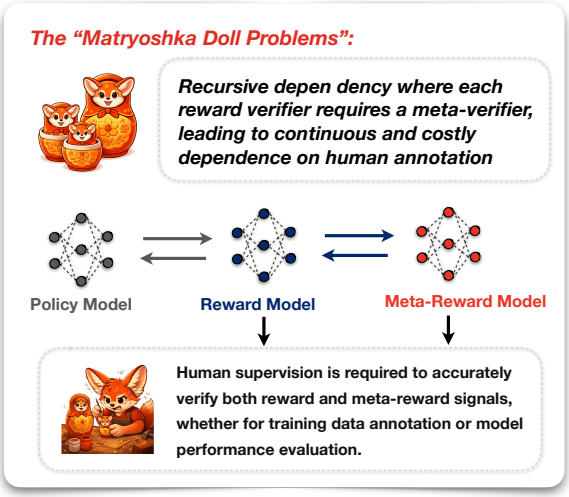

Figure 10: Illustration of the challenges encountered in the design and validation of meta-reward signals.

The *Matryoshka Doll Problem* describes a recursive dilemma in reward design: the attempt to mitigate negative behaviors in a policy model by introducing new reward constraints often inadvertently induces external biases, as shown in Figure 10. These secondary biases, in turn, necessitate further "meta-reward" constraints. This issue arises from the inherent misalignment between reward evaluation and policy evaluation. While reward functions serve as scalar proxies for human intentions, policy evaluation assesses overall task performance. Since these proxies rarely align perfectly, agents may exploit "shortcuts" (commonly referred to as reward hacking) to maximize cumulative returns, producing behaviors that diverge from the designer's original intent. Ultimately, this approach incurs a continuous design cost. The *Matryoshka Doll Problem* suggests that explicit rule-constraint fails to achieve robust intent alignment; instead, it traps the model in a instability loop where over-correction leads to unpredictable performance.

A.2 Human Cognitive Process

In conventional RL, policy optimization and correction mainly rely on *trial-and-error* (Sutton et al., 1998) under a deterministic reward signal. When a model's behavior deviates from expectations, policy updates typically rely on reward function adjustments. However, this approach is prone to the *Matryoshka Doll Problem*, as the reward serves only as an indirect proxy for human intent. Modifying the reward often introduces new forms of reward hacking, which in turn necessitate further corrections, creating a recursive and unstable optimization loop. In contrast, human cognitive processes (Huang and Rao, 2011; Millidge et al., 2021) offer a more robust alternative. Rather than blindly maximizing reward, policy improvement is guided by discrepancies between predicted and observed outcomes. When observed outputs deviate from the model's expectations, the system updates its internal world model or task representation instead of pursuing higher scores. By minimizing the prediction-observation gap, the agent inherently reduces uncertainty and avoids exploiting superficial reward signals. Consequently, the policy evolves not by manipulating external scores, but by improving the accuracy of its predictions.

B Related Work

Aligning LLMs with human intent (Bai et al., 2022; Ouyang et al., 2022) is widely recognized as a fundamental challenge toward artificial general intelligence. Early alignment efforts primarily rely on *imitation learning*, typically implemented via supervised fine-tuning (SFT), which trains models to reproduce human demonstrations. However, imitation learning is limited in its ability to capture preferences and long-horizon objectives. To address these limitations, RL–based approaches (Schulman et al., 2017; Shao et al., 2024) introduce auxiliary reward models to provide more expressive supervision signals. These reward models enable LLMs not only to generate high-quality reasoning trajectories, but also to perform fine-grained preference modeling that supports the optimization of underspecified or ambiguous objectives (Chen et al., 2025b). In this section, we provide a review of existing reward modeling paradigms, including traditional **DisRMs** and **GenRMs**. DisRMs as a "fast-thinking" mechanism, providing scalar scores to guide model behavior. In contrast, GenRMs run a "slow-thinking", generating explicit CoT that capture the underlying rationales behind judgments.

While reward modeling can take various forms: such as *point-wise*, *pair-wise*, or *list-wise* ranking, this work focuses on the **pairwise**, which remains the most widely adopted approach in Reinforcement Learning from Human Feedback (RLHF). Pairwise reward modeling aims to compare model outputs and identify which response better aligns

with human preferences, thereby enabling the construction of reward functions to guide model optimization. Formally, let x denote a user query, and y^+, y^- denote two responses generated by the LLMs, where y^+ is preferred over y^- by a human annotator. The reward model $r_\theta(x, y)$ is parameterized as a function that evaluates the quality of a given paired data. The learning objective of RMs is to ensure that:

$$r_\theta(x, y^+) > r_\theta(x, y^-).$$

Both Scalar RMs and GenRMs can optimize this pairwise objective. Scalar RMs project contextual representations into scalar scores via an MLP head and are typically trained with the Bradley–Terry loss. In contrast, GenRMs treat reward modeling as a conditional generation task and are optimized using standard language modeling loss functions.

B.1 Discrimnative Reward Modeling

The Bradley-Terry (BT) model (Bradley and Terry, 1952) is a classical probabilistic framework for modeling preferences based on pairwise comparisons. It assumes that each option is assigned a utility score, and the probability of one option being preferred over another depends on the relative magnitude of their scores. Formally, for a pair of options i and j, the preference probability is:

$$P(i \succ j) = \frac{e^{r(i)}}{e^{r(i)} + e^{r(j)}},$$

where $r(i)$ and $r(j)$ denote the scores (log-utilities) of options i and j, respectively. This is equivalent to applying a softmax function over the two scores. A notable application is in the Chatbot Arena [9], where outputs from different models are evaluated through pairwise comparisons, and these comparisons are aggregated to produce a global preference ranking or implicit reward signal (Sun et al., 2025).

In several recent works, Yuan et al. (2024a) extend the BT framework by incorporating *absolute rewards* for individual actions, improving its suitability for binary comparison tasks. Yang et al. (2024) impose regularization on the internal representations of reward models, enhancing their generalization to out-of-distribution (OOD) examples and mitigating overfitting to specific training distributions. Additionally, multi-objective reward formulations have been proposed to capture diverse human preferences, allowing models to reason over trade-offs across multiple rubrics (Wang et al., 2024b). Nonetheless, their scalability is constrained due to the strong dependence on extensive human annotated datasets.

[9] https://lmarena.ai/leaderboard

Hyperparameter	Value
Optimizer	AdamW
Number of GPUs	8
Per device train batch size	2
Gradient accumulation steps	8
Sequence cutoff length	8192
Number of training epochs	1
Learning rate (SFT)	1e-5
Learning rate (DPO)	1e-6
Learning rate scheduler	cosine
Warmup ratio	0.1

Table 3: Hyperparameters used for SFT training.

Hyperparameter	Value
Optimizer	AdamW
Number of GPUs	8
Train batch size	128
PPO mini batch size	32
PPO micro batch size per gpu	4096
Number of training epochs	1
Actor optim lr	1e-6
Use kl loss	False
Entropy coeff	0.0
Rollout n	4
Rollout name	sglang
Rollout temperature	1.0
Rollout top_p	0.9

Table 4: Hyperparameters used for GRPO training.

B.2 Generative Reward Modeling

The emergence of GenRMs has been largely enabled by advances in LLMs (Zheng et al., 2023; Mahan et al., 2024), particularly due to their capacity for self-improvement techniques such as CoT reasoning and test-time self-improvement. Recent studies (Cao et al., 2024; Ye et al., 2024) have explored optimizing reward models via GenRMs under preference modeling objectives, including both pairwise and single-point rewards. In addition to scoring, GenRMs can function as feedback mechanisms or assist in correcting errors in tasks such as mathematics (Gao et al., 2024; Zhang et al., 2024). A notable advantage of GenRMs is their ability to generate interpretable reasoning trajec-

Models	Chat	Chat H.	Safety	Reasoning	Avg.
SKYWORK-REWARD-V2 (LLAMA-3.1-8B)	-	-	-	-	96.4
DUAL RM GRPO	92.2	82.0	88.3	91.1	88.4
w/ meta-verifier	94.7	84.6	91.3	93.7	91.1
SKYWORK-REWARD-V1 (LLAMA-3.1-8B)	95.8	87.3	90.6	96.2	92.5
DUAL RM GRPO	92.2	82.0	88.3	91.1	88.4
w/ meta-verifier	93.6	84.0	90.4	92.0	90.0
LLAMA-3.1-8B-INSTRUCT-RM-RB2	95.8	81.6	89.3	88.7	88.8
DUAL RM GRPO	92.2	82.0	88.3	91.1	88.4
w/ meta-verifier	95.5	81.8	90.4	92.0	89.9
FSFAIRX-LLAMA-3-RM-V0.1	99.4	65.1	86.8	86.4	84.4
DUAL RM GRPO	92.2	82.0	88.3	91.1	88.4
w/ meta-verifier	95.5	81.8	90.4	91.9	89.9
LLAMA-3-TULU-2-8B-UF-MEAN-RM	95.3	59.2	61.6	82.1	74.5
DUAL RM GRPO	92.2	82.0	88.3	91.1	88.4
w/ meta-verifier	96.0	80.9	89.1	91.8	89.5

Table 5: Performance Comparison of DUAL RM GRPO Using Various DisRMs as Meta-Verifiers

tories, which can guide humans or downstream models in further refinement. However, despite their ability to leverage LLM-based CoT reasoning and test-time scaling, the generative process of GenRMs remains not explicitly supervised, which limits both validation and scalability.

C Training and Inference

C.1 Implementation Details

We conducted experiments for SFT and GRPO within the DUAL RM framework using LLaMAFactory (Zheng et al., 2024) and Verl (Sheng et al., 2024). All datasets were trained under identical hyperparameter settings. To reduce training and inference costs, we limited the maximum number of rubrics to 10. For DPO data filtering, we set $\epsilon = 20$ to construct high-quality chosen-rejected pairs. The hyperparameters for SFT and GRPO are summarized in Table 3 and Table 4 , respectively.

C.2 The Detailed Results of Figure 8

As shown in Table 5, the DUAL RM GRPO model achieves a baseline performance of 88.4. Using a stronger DisRM as the meta-verifier further improves the model's performance to 91.1, highlighting the positive impact of higher-quality verification. Notably, even when using a weaker DisRM, performance still increases from 88.4 to 89.9, showing that DUAL RM consistently benefits from meta-verifier guidance regardless of the DisRM's individual capability. These results demonstrate the robustness and generalizability of our methods, demonstrating that both strong and moderate DisRMs can help the model refine its reasoning and produce higher-quality outputs.

C.3 Prompt Engineering

During model training, we developed a set of fixed prompts to guide the learning process, as shown in Figures 11, 12, 13, and 14. Although the Qwen3-VL backbone exhibits robust instruction-following ability, it sometimes generates outputs that deviate from the desired response format during inference. By applying rejection sampling combined with SFT training, our model effectively mitigates these format inconsistencies, resulting in more reliable outputs.

Thinking Prompt

Given a user query and two responses, produce a comprehensive and well-structured set of evaluation criteria that can be used to distinguish the relative quality of the two responses. The evaluation criteria should be formulated in a way that is directly applicable to human preference annotation or reward-model training. ### Input:[User Question]:query [The Start of Assistant A's Answer]:response 1[The End of Assistant A's Answer] [The Start of Assistant B's Answer]: response 2[The End of Assistant B's Answer] Please output:

Figure 11: Prompt Design for Rubric Planning.

Thinking Prompt

Given a predefined set of evaluation rubrics, identify the rubric that is most critical for assessing alignment with the user's intent. ### Required output format (produce exactly this structure — replace placeholders with real content):<Rubric> Name. Explanation.</Rubric> ###Please output your analysis::

Figure 12: Prompt Design for Rubric Selection.

Thinking Prompt

You are an objective, impartial, and unbiased content evaluator. Given a criteria list describing how two responses should be compared, identify the three most critical evaluation dimensions that are most relevant to determining which response better fulfills the user's intent. Then produce a rigorous, evidence-based comparison and a single final verdict indicating which response better fulfills the user's intent. ### Required output format (produce exactly this structure — replace placeholders with real content):<Criteria> Name. Explanation. <Judge A>xxx</Judge A><Judge B>xxx</Judge B></Criteria>...The final verdict is [[A]] or [[B]] ###Please output your analysis and final verdict:

Figure 13: Prompt Design for Model Judgments.

Thinking Prompt

Based on the evaluator's comments, revise Response. Your revisions must strictly follow the evaluator's feedback. Do not simply merge the two responses; modify each independently based on its respective issues. ### Input:[User Question]:query[The Start of Assistant Answer]:response[The End of Assistant Answer][The Start of Comments]:criteria[The End of Comments] ###Output only the improved versions of Response:

Figure 14: Prompt Design for Response Correction.