RM-Distiller: Exploiting Generative LLM for Reward Model Distillation

Hongli Zhou[1], **Hui Huang**[1], **Wei Liu**[1], **Chenglong Wang**[2], **Xingyuan Bu**[3], **Lvyuan Han**[1], **Fuhai Song**[1], **Muyun Yang**[1*], **Wenhao Jiang**[1], **Hailong Cao**[1] and **Tiejun Zhao**[1]
[1]Faculty of Computing, Harbin Institute of Technology, Harbin, China
[2]School of Computer Science and Engineering, Northeastern University, Shenyang, China
[3]M-A-P
hongli.joe@stu.hit.edu.cn, yangmuyun@hit.edu.cn

Abstract

Reward models (RMs) play a pivotal role in aligning large language models (LLMs) with human preferences. Due to the difficulty of obtaining high-quality human preference annotations, distilling preferences from generative LLMs has emerged as a standard practice. However, existing approaches predominantly treat teacher models as simple binary annotators, failing to fully exploit the rich knowledge and capabilities for RM distillation. To address this, we propose RM-Distiller, a framework designed to systematically exploit the multifaceted capabilities of teacher LLMs: (1) Refinement capability, which synthesizes highly correlated response pairs to create fine-grained and contrastive signals. (2) Scoring capability, which guides the RM in capturing precise preference strength via a margin-aware optimization objective. (3) Generation capability, which incorporates the teacher's generative distribution to regularize the RM to preserve its fundamental linguistic knowledge. Extensive experiments demonstrate that RM-Distiller significantly outperforms traditional distillation methods both on RM benchmarks and reinforcement learning-based alignment, proving that exploiting multifaceted teacher capabilities is critical for effective reward modeling. To the best of our knowledge, this is the first systematic research on RM distillation from generative LLMs.

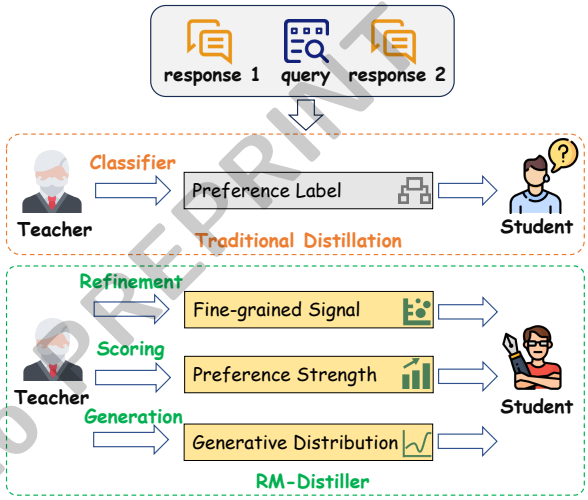

Figure 1: Compared to traditional method, RM-Distiller unlocks the multifaceted potential of teacher LLMs for better distillation.

1 Introduction

Large language models (LLMs) have demonstrated remarkable capabilities across diverse domains, and their effective deployment relies on alignment with human values through Reinforcement Learning from Human Feedback (RLHF) [Bai *et al.*, 2022a]. This paradigm primarily depends on reward modeling, which provides a proxy for human preferences to supervise policy optimization. However, effective reward models (RMs) require to train on large-scale, high-quality human preference data. As task complexity scales, obtaining high-quality human annotations becomes prohibitively expensive and difficult [Touvron *et al.*, 2023].

To mitigate reliance on human feedback, some works proposed to directly use powerful generative LLMs as RMs [Zheng *et al.*, 2023; Huang *et al.*, 2026]. However, despite their strong preference modeling capabilities, their practical use is limited by high API costs and inference latency. Consequently, other works have proposed to distill discriminative RMs from generative LLMs [Bai *et al.*, 2022b; Malik *et al.*, 2026]. These methods typically employ the LLM as an annotator to assign binary preference labels to response pairs. The resulting preference data with triplets of (*instruction*, *responses*, *label*) are then used to train RMs. Distilling RMs from powerful generative LLMs offers a cost-effective and scalable alternative to human annotation. As a result, it has been adopted widely in recent RL practice [Liu *et al.*, 2024; Liu *et al.*, 2025a].

Despite its popularity, this paradigm predominantly utilizes the teacher model's capability to assign binary preference labels, often treating it as a mere binary classifier [Huang *et al.*, 2025], failing to fully leverage the rich capabilities embedded in LLMs. In practice, generative LLMs possess a wide range of capabilities and internal knowledge that are

*Corresponding author.

highly beneficial for preference modeling. The insufficient utilization of these capabilities limits the upper bound of RM performance and subsequent RL-based alignment.

We therefore investigate a crucial yet unexplored question: *How can we more effectively distill generative LLMs into reward models?*

In this paper, we propose RM-Distiller, a framework that exploits generative LLMs to distill RMs, by incorporating their multifaceted capabilities across three dimensions:

1. **Refinement capability:** We leverage the refinement capability of generative LLMs to construct highly contrastive preference pairs for RM training. By performing targeted revisions on rejected responses, the teacher synthesizes highly correlated response pairs that differ only in the attributes determining preference. This compels the RM to discern subtle quality nuances instead of spurious correlations [Jiang *et al.*, 2025].
2. **Scoring capability:** We utilize the scoring capability of generative LLMs to assess response quality and model preference strength. Instead of relying on discrete binary annotation, the teacher provides continuous preference scores that better reflect varying degrees of quality differences between response pairs. This allows the RM to be optimized with a margin-aware objective, capturing more nuanced preference strengths.
3. **Generation capability:** We incorporate the generation capability of generative LLMs to safeguard the foundational linguistic proficiency of the student model. By distilling the teacher's output distribution into the student backbone, we enforce a constraint that prevents the RM from undergoing catastrophic forgetting. This ensures the model retains robustness during regressive RM training [Yang *et al.*, 2024b].

We evaluate RM-Distiller on both RM benchmarks and downstream RL alignment tasks. The results demonstrate that RM-Distiller significantly outperforms existing distillation methods with superior RM accuracy and RL alignment efficacy. Despite the prevailing practice of distilling generative LLMs into RMs, this is the first work that systematically explores the best practice for RM distillation.

Our contributions are as follows:

1. We conduct the first systematic investigation into RM distillation, by introducing a multifaceted distillation strategy RM-Distiller.
2. We propose to exploit three capabilities of the generative LLMs, **refinement**, **scoring**, and **generation**, to provide comprehensive supervision for training the RMs.
3. Extensive experiments demonstrate that RM-Distiller significantly improves RM performance both on RM benchmarks and downstream alignment scenarios.

2 Background

2.1 Preliminary: Reward Modeling

RMs traditionally relied on human preference datasets, typically trained using the Bradley-Terry (BT) model to predict pairwise rankings. The standard objective of RM training is to learn a scalar function $r_\phi(x, y)$ that reflects human preference. Given a preference dataset $\mathcal{D} = \{(x, y_w, y_l)\}$, where y_w and y_l denote the chosen and rejected responses for instruction x, respectively, the RM is optimized by minimizing the negative log-likelihood:

$$\mathcal{L}_{\text{BT}}(\phi) = \log \sigma \left(r_\phi(x, y_w) - r_\phi(x, y_l) \right) \tag{1}$$

where σ is the sigmoid function. While effective for ranking, this loss function only considers the binary order of responses, ignoring the preference strength between them.

2.2 RM Distillation from Generative LLM

Due to the high cost and limited scalability of human oversight [Touvron *et al.*, 2023], the community has increasingly shifted toward getting preference annotation from generative LLM [Liu *et al.*, 2024], where teacher models are employed to evaluate paired responses and identify the preferred ones, thereby constructing preference annotations for training BT models [Liu *et al.*, 2025a]. However, this paradigm often treats the teacher as a black-box classifier, failing to exploit its generative potential or the nuanced preference intensities necessary for reliable alignment.

To address this, recent studies have explored ways to enrich the distillation signal by incorporating more diverse information from teacher models. For instance, SynRM [Ye *et al.*, 2025] integrates natural language critiques into RMs by appending teacher-generated justifications to the responses during both training and inference. Further extending this idea, CLoud [Ankner *et al.*, 2024] explicitly distills the capability of generating critiques into discriminative RMs. Another line of research focuses on data augmentation, utilizing LLMs to synthesize a larger volume of preference data through response perturbation to cover a broader distribution of scenarios [Park *et al.*, 2024; Shen *et al.*, 2025]. While these methodologies enhance supervision or data diversity, they often only exploit a fraction of the teacher model's multifaceted capabilities, failing to fully harness its potential for RM distillation.

3 RM-Distiller

RM-Distiller begins by Initial Preference Pair Construction, which serves as the foundation for RM distillation. Subsequently, it fully exploits the teacher model's multifaceted capabilities through: (1) Refinement capability for Contrastive Refinement, (2) Scoring capability for Strength Guidance, (3) Generation capability for Generative Regularization.

3.1 Initial Preference Pair Construction

Initial Candidate Generation

Given an instruction x from an instruction set $\mathcal{D}_x$, we first choose a set of candidate models $\mathcal{M} = \{m_1, m_2, \ldots, m_K\}$. After that, we collect a response pool $\mathcal{Y}_x = \{y_1, y_2, \ldots, y_K\}$ such that each $y_j \sim m_j(x)$. The candidate models are chosen to differ in model families and sizes, ensuring responses to each instruction are with varying styles and quality levels.

Initial Preference Annotation

To construct the initial preference data, for each instruction x, we randomly sample two distinct responses (y_a, y_b) from

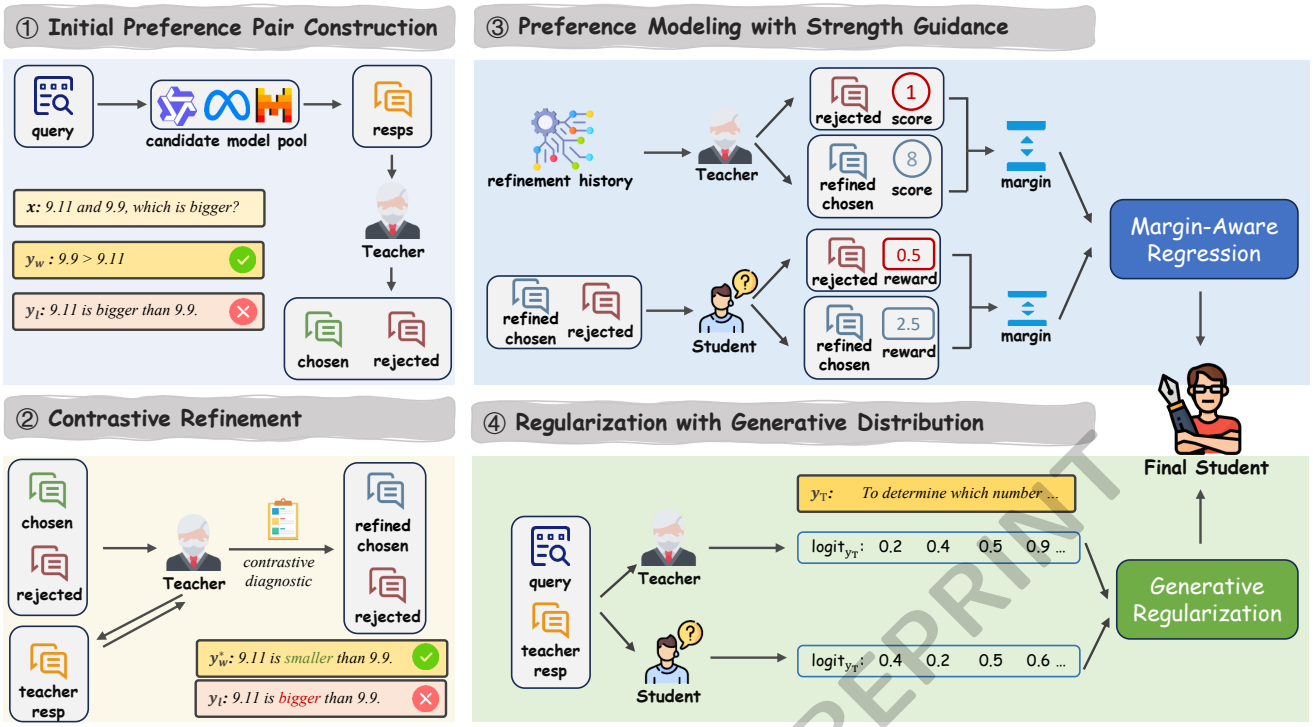

Figure 2: The illustration of RM-Distiller. We first initialize preference pairs from diverse candidate models, and then synthesize highly correlated response pairs. During the training phase, the student model is guided by continuous preference margins and generative regularization.

$\mathcal{Y}_x$. We then employ the generative teacher LLM f_T to determine the relative quality between (y_a, y_b). This results in an annotated preference pair (y_w, y_l), where y_w and y_l denote the chosen and rejected responses, respectively.

At this stage, we obtain preference triplets (x, y_w, y_l), which can be directly used to train RMs under the BT objective, as in standard RM distillation. However, this will result in the insufficient utilization of teacher models' capabilities. Therefore, we introduce the following methods to further leverage the teacher's supervision signal.

3.2 Contrastive Refinement

The efficacy of RMs depends heavily on the contrastive quality of the preference pairs. In standard distillation practices, pairs are often sampled independently, therefore exhibit significant discrepancies in length, formatting, or linguistic style [Jiang *et al.*, 2025], which results in the RMs to rely on superficial shortcut instead of intrinsic quality attributes [Zhou *et al.*, 2024]. To eliminate these spurious correlations and force the model to identify fine-grained preference boundaries, we utilize the refinement capability of the teacher model to synthesize highly correlated contrasts, as shown in Figure 3.

The refinement process begins by analyzing the gap between the rejected response y_l and the teacher-generated response y_T, where y_T serves as a high-quality reference. We first prompt the teacher model f_T to perform a contrastive diagnostic, where it articulates specific reasons for the rejection and outlines a path for improvement. Following this diagnostic, f_T is instructed to minimally modify y_l into a refined chosen version y_w^* that matches the quality of y_T while strictly

Contrastive Refinement Example

Instruction
How to get the last element of a list `L` in Python?

Rejected Response (y_l)
You can use `L[0]` to access the last element of a list.

Refined Chosen Response (y_w^*)
You can use `L[-1]` to access the last element of a list.

Teacher Response (y_T)
In Python, negative indexing allows you to access elements from the end, with `-1` representing the last item.

Figure 3: A concrete example of Contrastive Refinement.

preserving the original structure and phrasing. By maintaining a minimal edit distance, we ensure that y_w^* and y_l share a nearly identical semantic backbone, differing only in the attributes that define superiority.

3.3 Preference Modeling with Strength Guidance

Margin-Aware Regression

In the traditional BT framework, the RM is trained as a binary classifier to maximize the probability $P(y_w \succ y_l)$. This objective implicitly assuming a uniform preference strength across all response pairs. However, in reality, the quality difference between responses varies significantly [Wang *et al.*, 2024]. Therefore, we leverage the scoring capability of LLMs to explicitly capture and represent the strength of preferences

to provide fine-grained supervision signals.

First, we prompt the teacher model f_T to assign scalar quality scores $s_T(x, y)$ for chosen and rejected responses y_w and y_l, respectively. After that, we propose Margin-Aware Regression, to guide the student model to mimic the teacher's evaluative sensitivity by aligning the predicted reward difference with the teacher-generated score margin. Specifically, we define margin-aware loss as follows:

$$\mathcal{L}_{\text{margin}} = \Big((r_\phi(x, y_w) - r_\phi(x, y_l)) - (s_T(x, y_w) - s_T(x, y_l))\Big)^2 \quad (2)$$

With this objective, the student model explicitly learns to align not only the relative order but also the exact magnitude of preferences, thereby being able to capture fine-grained variations in preference strength, distinguishing between subtle improvements and absolute errors. While some recent studies have attempted to introduce a margin as a lower bound [Touvron *et al.*, 2023], they still fail to capture the nuanced intensity of preference as in this work.

Self-Calibrated Scoring

As established in Section 3.2, we obtained the refined response y_w^* through Contrastive Refinement as the chosen response. To assign more precise quality scores, we introduce Self-Calibrated Scoring. Specifically, we explicitly provide the teacher with the previously assigned score of the rejected response $s_T(x, y_l)$ as a fixed anchor, and task it with scoring the refined response y_w^* on the same scale. As a result, the score $s_T(x, y_w^*)$ reflects a calibrated increase proportional to the specific corrections applied during refinement. This relative evaluation protocol yields more stable and learnable preference margins for downstream reward modeling.

Moreover, during this process, we reuse the refinement dialogue history as the context for scoring, exposing the scorer to the teacher-produced contrastive diagnosis and its refinement outcome. By conditioning the scoring on the explicit refinement history, Self-Calibrated Scoring transforms absolute quality assessment into a relative adjustment.

Preference Pair Filtering

To ensure high data quality, we filter the synthesized preference pairs to retain sufficiently strong contrasts based on two criteria: (1) a substantial score margin and (2) a minimum edit distance between y_w^* and y_l:

$$\begin{aligned}\mathcal{D}_{\text{refine}} = \{(x, y_w^*, y_l, s_T(x, y_w^*), s_T(x, y_l) \mid \\ \text{dist}_{\text{edit}}(y_w^*, y_l) > \tau_e \wedge \Delta s > \tau_s\}\end{aligned} \quad (3)$$

where τ_s and τ_e denote the minimum thresholds for the score margin and the edit distance, respectively. Filtering for edit distance ensures that the refinement has introduced meaningful changes, while score margin ensures that the teacher recognizes a sufficient quality improvement. This ensures effective contrast between preference pairs for RM training.

3.4 Regularization with Generative Distribution

Generative Regularization

Solely optimizing for discriminative signals can lead to reward over-optimization, where the model gradually loses its foundational understanding of language, which is essential for generalizability [Yang *et al.*, 2024b]. Therefore, we propose Generative Regularization to preserve the generative knowledge of the backbone model during distillation.

Specifically, we retain the student's language model (LM) head and use the teacher f_T as a linguistic anchor. For each teacher response y_T, we supervise the student's LM head using a combination of negative log-likelihood (NLL) and Kullback-Leibler (KL) divergence distillation:

$$\begin{aligned}\mathcal{L}_{\text{reg}} = \frac{1}{L}\sum_{t=1}^{L}\Big(&- \alpha \cdot \log P_\phi(y_T^t | x, y_T^{<t}) \\ &+ \beta \cdot \sum_{v \in \mathcal{V}} P_T(v | x, y_T^{<t}) \log \frac{P_T(v | x, y_T^{<t})}{P_\phi(v | x, y_T^{<t})}\Big)\end{aligned} \quad (4)$$

where $y_T = \{y_T^1, y_T^2, \ldots, y_T^L\}$ denotes the teacher response of length L, and $\mathcal{V}$ denotes the vocabulary. P_ϕ and P_T represent the next-token probability distributions of the student and the teacher, respectively. α and β are coefficients that control the strength of the NLL loss and the KL divergence. Notably, if the teacher model is closed-source or if the teacher and student do not share a common vocabulary, we set $\beta = 0$.

Total Objective Function

Combining our proposed Margin-Aware Regression and the Generative Regularization loss, the total optimization objective of RM-Distiller is formulated as a weighted sum:

$$\mathcal{L}_{\text{total}} = (1 - \alpha - \beta) \cdot \mathcal{L}_{\text{margin}} + \mathcal{L}_{\text{reg}} \quad (5)$$

This multi-task objective synergizes precise preference modeling with linguistic anchoring. Specifically, $\mathcal{L}_{\text{margin}}$ enables the model to capture precise quality differences, while $\mathcal{L}_{\text{reg}}$ prevents representation collapse by grounding the shared backbone in the teacher's generative distribution. This dual supervision ensures semantic robustness, leading to better generalization and stability during downstream RLHF.

4 Experiments

In this section, we evaluate RM-Distiller through a series of experiments to assess its reward modeling performance.

4.1 Experimental Setup

Data Settings

For training RMs, we utilize a 10,000-sample subset randomly drawn from the Skywork-Preference-v0.2 dataset [Liu *et al.*, 2024]. Our method is evaluated across two scenarios:

- **Unlabeled Scenario** where no prior preference annotations are available, and we only utilize instructions from the dataset and employ candidate models to generate responses. This setup mirrors practical distillation settings with no access to high-quality human annotations, making LLM-based distillation a more practical alternative.
- **Labeled Scenario** where response pairs are associated with high-quality ground-truth preference annotations, and we retain the original response pairs from the dataset and keep only those samples for which the teacher's preference judgments are consistent with the original labels.

Student	Teacher	Method	RewardBench					RM-Bench			
			Chat	Hard	Safety	Reason	Avg.	Easy	Normal	Hard	Avg.
LLM-as-a-Judge											
Claude-3.5-Sonnet-20240620			96.4	74.0	81.6	84.7	84.2	73.8	63.4	45.9	61.0
Llama-3.1-70B-Instruct [Grattafiori *et al.*, 2024]			97.2	70.2	82.8	86.0	84.1	74.7	67.8	54.1	65.5
GPT-4o [Hurst *et al.*, 2024]			96.1	70.7	87.8	75.1	82.4	83.8	70.9	48.2	67.6
Qwen3-14B [Yang *et al.*, 2025]			96.4	69.2	88.0	82.0	83.9	83.6	71.7	52.1	69.1
Qwen2.5-3B-Instruct [Yang *et al.*, 2024a]			88.3	43.2	55.1	42.7	57.3	69.6	49.4	34.3	51.1
Open-Source Reward Models											
Eurus-RM-7B [Yuan *et al.*, 2025]			98.0	65.6	81.4	86.3	81.6	87.2	70.2	40.2	65.9
GRAM-LLaMA3.2-3B-RewardModel [Wang *et al.*, 2025]			90.6	83.9	79.8	80.2	83.6	84.2	65.2	40.0	63.2
JudgeLRM-7B [Chen *et al.*, 2025]			92.9	56.4	78.2	73.6	75.2	73.2	66.2	54.8	64.7
Training on the Same Preference Dataset in the Unlabeled Scene											
Qwen2.5-3B-Instruct	GPT-4o	BT Classifier	93.3	37.7	69.3	81.0	70.3	**91.9**	62.3	13.6	55.9
		Margin BT [Touvron *et al.*, 2023]	93.6	33.6	73.5	80.8	70.4	90.4	61.0	17.5	56.3
		SteerLM [Dong *et al.*, 2023]	94.4	46.5	74.3	82.8	74.5	90.2	59.8	17.1	55.7
		SynRM [Ye *et al.*, 2025]	**96.1**	47.6	78.7	79.1	75.4	91.0	**66.2**	25.1	60.7
		RMBoost [Shen *et al.*, 2025]	91.3	35.1	67.8	74.8	67.3	86.3	54.9	17.6	52.9
		CLoud [Ankner *et al.*, 2024]	93.6	37.1	62.6	63.6	64.2	87.1	59.2	18.9	55.1
		LSAM [Wang *et al.*, 2024]	94.1	37.3	73.5	81.4	71.6	89.3	58.8	22.2	56.8
		RM-Distiller	95.5	**51.1**	**84.1**	**87.4**	**79.5**	86.4	64.8	**32.2**	**61.1**
	Qwen3-14B	BT Classifier	88.0	43.4	72.4	83.5	71.8	89.8	60.7	19.8	56.7
		Margin BT [Touvron *et al.*, 2023]	94.7	46.3	79.9	70.4	72.8	90.7	63.6	26.3	60.2
		SteerLM [Dong *et al.*, 2023]	93.6	45.0	79.1	78.5	74.0	89.9	64.1	19.9	58.0
		SynRM [Ye *et al.*, 2025]	84.9	**57.0**	79.1	**91.1**	78.0	**92.7**	**66.7**	20.0	59.8
		RMBoost [Shen *et al.*, 2025]	92.7	43.6	73.7	90.6	75.2	84.8	60.3	25.9	57.0
		CLoud [Ankner *et al.*, 2024]	91.1	39.7	74.1	67.5	68.1	87.7	58.4	20.3	55.5
		LSAM [Wang *et al.*, 2024]	92.5	39.9	72.7	84.9	72.5	88.8	60.4	24.6	57.9
		RM-Distiller	**95.0**	55.0	**81.8**	89.0	**80.2**	87.2	66.3	**34.6**	**62.7**
Training on the Same Preference Dataset in the Labeled Scene											
Qwen2.5-3B-Instruct	—	BT Classifier	77.4	71.3	86.5	60.0	73.8	81.7	64.5	43.2	63.2
	GPT-4o	Margin BT [Touvron *et al.*, 2023]	88.6	70.8	87.3	67.8	78.6	80.3	66.8	**48.2**	65.1
		SteerLM [Dong *et al.*, 2023]	91.6	65.4	80.4	63.5	75.2	85.6	65.8	37.4	62.9
		SynRM [Ye *et al.*, 2025]	81.0	**82.5**	86.9	78.6	82.2	80.9	66.0	40.3	62.4
		RMBoost [Shen *et al.*, 2025]	85.8	73.7	86.6	74.9	80.3	79.3	59.0	38.8	59.1
		CLoud [Ankner *et al.*, 2024]	69.3	66.0	72.6	66.9	68.7	68.3	57.1	45.4	57.0
		LSAM [Wang *et al.*, 2024]	89.4	66.5	88.1	63.8	77.0	85.2	63.8	36.7	61.9
		RM-Distiller	**92.7**	64.3	**90.7**	**86.9**	**83.6**	**87.0**	**68.9**	42.0	**65.9**
	Qwen3-14B	Margin BT [Touvron *et al.*, 2023]	88.6	72.8	86.6	71.6	79.9	81.0	67.8	**49.1**	66.0
		SteerLM [Dong *et al.*, 2023]	89.7	69.5	**87.2**	67.1	78.4	84.6	68.3	43.8	65.6
		SynRM [Ye *et al.*, 2025]	78.2	**80.7**	85.1	74.8	79.7	73.6	61.5	48.5	61.2
		RMBoost [Shen *et al.*, 2025]	82.7	73.7	82.2	88.2	81.7	84.8	60.3	25.9	57.0
		CLoud [Ankner *et al.*, 2024]	83.0	67.3	75.0	62.2	71.9	77.3	61.9	44.9	61.4
		LSAM [Wang *et al.*, 2024]	**93.0**	69.7	84.1	70.4	79.3	**90.7**	66.0	35.2	63.9
		RM-Distiller	91.3	69.7	85.5	**91.3**	**84.5**	89.5	**69.3**	40.5	**66.4**

Table 1: Results of different RM distillation methods on RewardBench and RM-Bench.

We employ RewardBench [Lambert *et al.*, 2025] and RM-Bench [Liu *et al.*, 2025b] as our primary evaluation benchmarks, which cover diverse subsets.

Models

We employ Qwen2.5-3B-Instruct [Yang *et al.*, 2024a] as the backbone for the student RM. To facilitate distillation, we use GPT-4o [Hurst *et al.*, 2024] and Qwen3-14B [Yang *et al.*, 2025] in its non-thinking mode as the teacher models. For constructing the response pool in the unlabeled scenario, we leverage a set of candidate models including Llama-3.1-8B-Instruct [Grattafiori *et al.*, 2024], Mistral-7B-Instruct-v0.3 [Jiang *et al.*, 2023], Gemma-2-9b-it [Team *et al.*, 2024] and Vicuna-7b-v1.5 [Zheng *et al.*, 2023].

Baselines

We compare RM-Distiller against several RM distillation methods including the vanilla BT Classifier, alongside its extensions Margin BT [Touvron *et al.*, 2023] and LSAM [Wang *et al.*, 2024]. We also evaluate against methods that leverage teacher-generated critiques, revisions or attributes, including SynRM [Ye *et al.*, 2025], CLoud [Ankner *et al.*, 2024], RM-Boost [Shen *et al.*, 2025], and SteerLM [Dong *et al.*, 2023].

We also include a range of external RMs as reference points. These comprise discriminative models such as Eurus-RM-7B [Yuan *et al.*, 2025], as well as off-the-shelf generative models including Claude-3.5-Sonnet and Llama-3.1-70B-Instruct, along with fine-tuned generative models including GRAM-LLaMA3.2-3B-RewardModel [Wang *et al.*,

Policy	Student RM	Teacher	Dataset	Method	Algorithm	AlpacaEval 2.0		FollowBench		CF-Bench		
						LC	Len	HSR	SSR	CSR	ISR	PSR
Llama-3-8B	—	—	UltraChat	Baseline	SFT	8.86	1,017	41.04	57.39	0.51	0.15	0.22
	Qwen2.5-3B -Instruct	GPT-4o	HH-RLHF	BT Classifier	PPO	14.32	2,045	46.85	61.20	0.57	0.18	0.26
					GRPO	17.89	2,310	48.33	62.15	0.61	0.22	0.30
					DAPO	18.25	2,005	48.59	62.48	0.62	0.22	0.31
				RM-Distiller	PPO	16.90	1,927	48.42	62.05	0.60	0.23	0.29
					GRPO	20.10	2,342	49.95	63.14	**0.64**	**0.24**	**0.33**
					DAPO	**20.52**	2,487	**50.09**	**63.31**	0.63	0.23	0.32
			ShareGPT	BT Classifier	PPO	13.80	2,584	46.20	60.95	0.56	0.17	0.25
					GRPO	17.92	2,360	47.90	61.88	0.61	0.21	0.30
					DAPO	17.45	2,200	48.15	62.10	0.60	0.21	0.29
				RM-Distiller	PPO	16.24	2,293	47.78	61.77	0.59	0.19	0.27
					GRPO	19.85	2,090	49.45	62.90	0.62	0.22	0.31
					DAPO	**20.15**	2,335	**49.62**	**63.15**	**0.63**	**0.23**	**0.32**

Table 2: Performance of policy model supervised by different RM distillation methods.

Method	RewardBench Average	RM-Bench Average
GPT-4o as Teacher and Qwen2.5-3B-Instruct as Student		
BT Classifier	70.3	55.9
+ Margin-Aware Regression	72.7	57.3
+ Contrastive Refinement	76.8	58.8
+ Generative Regularization	**79.5**	**61.1**
Qwen3-14B as Teacher and Qwen2.5-3B-Instruct as Student		
BT Classifier	71.8	56.7
+ Margin-Aware Regression	75.7	59.5
+ Contrastive Refinement	78.7	61.5
+ Generative Regularization	**80.2**	**62.7**

Table 3: Ablation study of RM-Distiller components.

2025] and JudgeLRM-7B [Chen *et al.*, 2025].

4.2 Main Results

As shown in Table 1, RM-Distiller consistently and significantly outperforms traditional distillation methods across both unlabeled and labeled settings, whether using a closed-source or open-source model as a teacher. Notably, RM-Distiller also surpasses several powerful external baselines, including both frontier generative models and open-source RMs built on significantly larger foundation models or trained with more extensive datasets.

These performance gains are driven by the effective synergy of three key capabilities leveraged from the teacher model. Specifically, the teacher's refinement and scoring abilities enable the RM to identify fine-grained differences in response pairs, thereby enhancing the perceptual capability of nuanced quality aspects. Furthermore, the teacher's generation capability helps the RM maintain a foundational understanding of language, allowing it to better comprehend what constitutes a good response.

4.3 Ablation Study

To further verify the effectiveness of RM-Distiller, we conduct an ablation study by incrementally adding each module, corresponds to each capability of the teacher. As shown in Table 3, each proposed module contributes significantly to the overall performance gains. Across both RewardBench and RM-Bench, we observe a steady increase in average accuracy as we incrementally integrate Margin-Aware Regression, Contrastive Refinement, and Generative Regularization. This confirms that leveraging the teacher's multifaceted capabilities is essential for high-fidelity RM distillation on both evaluation benchmarks.

4.4 Performance on RLHF

To evaluate the effectiveness of the distilled RMs in downstream alignment, we conduct RLHF experiments using PPO [Schulman *et al.*, 2017], GRPO [Shao *et al.*, 2024], and DAPO [Yu *et al.*, 2025]. For all RL experiments, we adopt Llama-3-8B [Grattafiori *et al.*, 2024] as the base policy model. We first perform SFT on the base model using the UltraChat dataset [Ding *et al.*, 2023] as a cold-start. We then use prompts from the HH-RLHF [Bai *et al.*, 2022a] and ShareGPT [Zheng *et al.*, 2023] datasets as training instructions during RL. Alignment effectiveness are evaluated on both general and complex instruction following benchmarks, including AlpacaEval 2.0 [Dubois *et al.*, 2024], FollowBench [Jiang *et al.*, 2024] and CFBench [Zhang *et al.*, 2025][1].

As shown in Table 2, RM-Distiller consistently outperforms the BT Classifier across all datasets and RL algorithms, indicating more effective reward signal for RLHF. Our distilled RM significantly enhances both general alignment and complex instruction following, demonstrating its ability to discern nuanced quality differences. Furthermore, the consistent gains under length-controlled evaluations confirm that these improvements reflect genuine enhancements in response quality rather than a mere verbosity bias.

[1]Since these benchmarks rely on LLM-based evaluation, we adopt GPT-4o-0806 as the judge model.

Model	Sample Num	Acc.
Skywork-Reward-V2-8B [Liu *et al.*, 2025a]	40,000K	81.3
Skywork-Reward-8B-v0.2 [Liu *et al.*, 2024]	80K	75.9
URM-LLaMA-3.1-8B [Lou *et al.*, 2024]	100K	76.7
Llama-3-OffsetBias-8B [Park *et al.*, 2024]	70K	83.2
Tulu-3-8B-RM-RB2 [Malik *et al.*, 2026]	350K	83.2
BT-Qwen2.5-3B-Instruct	10K	72.2
RM-Distiller-Qwen2.5-3B-Instruct	10K	**85.4**

Table 4: Results of different methods and previous RMs on the Arabic preference test set.

Student	Teacher	Method	EvalBiasBench
GPT-4o [Hurst *et al.*, 2024]			70.0
Qwen3-14B [Yang *et al.*, 2025]			75.0
Qwen2.5-3B-Instruct [Yang *et al.*, 2024a]			44.4
Qwen2.5-3B-Instruct	GPT-4o	BT Classifier	30.0
		RM-Distiller	**57.5**
	Qwen3-14B	BT Classifier	47.5
		RM-Distiller	**63.8**

Table 5: Results of different methods on EvalBiasBench.

4.5 Fast Adaptation Capability

Fast Adaptation refers to the ability to rapidly provide reliable reward signals for novel, domain-specific tasks, which is a vital determinant of RM's efficacy. RM-Distiller offers an efficient and practical solution, enabling the rapid distillation of specialized RMs with only a small amount of targeted data. To evaluate this capability, we choose the Arabic language domain, where we utilize an Arabic dataset[2], which has significant difference with general preference modeling[3]. We utilize Qwen2.5-3B-Instruct as the student model and Qwen3-14B as the teacher model.

As shown in Table 4, RM-Distiller not only demonstrates a substantial margin over the BT model but also surpasses currently top-performing generalist RMs. This empirical evidence reveals that generalist RMs struggle to achieve true universality across diverse specific domains. In contrast, RM-Distiller provides an efficient mechanism to adapt and distill reliable reward signals into specialized scenarios with only a small group of instruction data, enabling fast adaptation of reward modeling to a new domain.

4.6 Generalizability

A robust RM should generalize to challenging scenarios without exploiting spurious shortcuts. Poor generalizability often leads to reward hacking, where the model exploits unintended biases to achieve deceptively high scores without fully following the instructions [Zhou *et al.*, 2026]. In this section, we employ EvalBiasBench [Park *et al.*, 2024] and IFBench [Peng *et al.*, 2025] as our evaluation benchmarks.

[2]https://huggingface.co/datasets/FreedomIntelligence/Arabic-preference-data-RLHF

[3]We randomly sample 10,000 instructions as the start point for distillation. For evaluation, we extract a test set of 1,000 samples, which are filtered by GPT-5 to ensure a reliable ground truth.

Student	Teacher	Method	IFBench
GPT-4o [Hurst *et al.*, 2024]			50.8
Qwen3-14B [Yang *et al.*, 2025]			54.2
Qwen2.5-3B-Instruct [Yang *et al.*, 2024a]			36.9
Qwen2.5-3B-Instruct	GPT-4o	BT Classifier	50.2
		RM-Distiller	**57.9**
	Qwen3-14B	BT Classifier	52.5
		RM-Distiller	**57.7**

Table 6: Results of different methods on IFBench.

Teacher	Method	RewardBench Average	RM-Bench Average
GPT-4o	BT Classifier	50.3	48.1
	RM-Distiller	**76.1**	**59.9**
Qwen3-14B	BT Classifier	64.6	51.9
	RM-Distiller	**76.4**	**60.2**

Table 7: Results of different methods under 1K instructions.

As shown in Table 5 and 6, RM-Distiller demonstrates a superior capacity to generalize to out-of-distribution tasks. In EvalBiasBench, while the BT Classifier collapses by relying on superficial patterns, RM-Distiller exhibits significantly higher robustness. This indicates the student model learns to discern semantic boundaries rather than spurious correlations. In IFBench, RM-Distiller not only outperforms the BT Classifier baseline but even surpasses the direct prompting performance of the teacher. This demonstrates that RM-Distiller effectively captures fine-grained constraints within instructions, verifying its generalizability to unseen tasks.

4.7 Data Efficiency

Data efficiency is critical in RM distillation, as high-quality instruction data is often scarce when dealing with specialized domains or rapid model iteration. To evaluate this, we compare RM-Distiller against the BT baseline using a reduced training set of only 1,000 instructions.

As illustrated in Table 7, when the training set is reduced to only 1,000 instructions, the performance of the student model trained with BT model deteriorates sharply. In contrast, RM-Distiller exhibits exceptional data efficiency, maintaining a high level of discriminative accuracy. By extracting significantly more supervisory information from each sample, RM-Distiller effectively distills the teacher's underlying preferences. This high-density training signal enables the development of reliable RMs with minimal data, offering a practical solution for real-world alignment tasks.

5 Conclusion

In this work, we present the first systematic research about how to distill generative LLMs into discriminative RMs. By exploiting the multifaceted capabilities of generative LLMs, our proposed RM-Distiller provides the student model with rich, reliable supervision signals, enabling high RM performance without additional inference overhead.

Acknowledgments

This work was supported by the National Key R&D Program of China (Grant No. 2024YFC3809100), the National Natural Science Foundation of China (Grant No. 62276077, 62376075, 62376076), the Department of Science and Technology of Heilongjiang Province (Grant No. ZL2025F004), and the MOE Key Laboratory of Cognitive Intelligence and Content Security (Grant No. 10120251103, Harbin Institute of Technology).

Contribution Statement

Hongli Zhou, Hui Huang and Wei Liu contributed equally.

References

[Ankner *et al.*, 2024] Zachary Ankner, Mansheej Paul, Brandon Cui, Jonathan Daniel Chang, and Prithviraj Ammanabrolu. Critique-out-loud reward models. In *Pluralistic Alignment Workshop at NeurIPS 2024*, 2024.

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

[Bai *et al.*, 2022b] Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, et al. Constitutional ai: Harmlessness from ai feedback. *arXiv preprint arXiv:2212.08073*, 2022.

[Chen *et al.*, 2025] Nuo Chen, Zhiyuan Hu, Qingyun Zou, Jiaying Wu, Qian Wang, Bryan Hooi, and Bingsheng He. Judgelrm: Large reasoning models as a judge. *arXiv preprint arXiv:2504.00050*, 2025.

[Ding *et al.*, 2023] Ning Ding, Yulin Chen, Bokai Xu, Yujia Qin, Shengding Hu, Zhiyuan Liu, Maosong Sun, and Bowen Zhou. Enhancing chat language models by scaling high-quality instructional conversations. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 3029–3051, 2023.

[Dong *et al.*, 2023] Yi Dong, Zhilin Wang, Makesh Sreedhar, Xianchao Wu, and Oleksii Kuchaiev. SteerLM: Attribute conditioned SFT as an (user-steerable) alternative to RLHF. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, 2023.

[Dubois *et al.*, 2024] Yann Dubois, Percy Liang, and Tatsunori Hashimoto. Length-controlled alpacaeval: A simple debiasing of automatic evaluators. In *First Conference on Language Modeling*, 2024.

[Grattafiori *et al.*, 2024] Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

[Huang *et al.*, 2025] Hui Huang, Xingyuan Bu, Hongli Zhou, Yingqi Qu, Jing Liu, Muyun Yang, Bing Xu, and Tiejun Zhao. An empirical study of LLM-as-a-judge for LLM evaluation: Fine-tuned judge model is not a general substitute for GPT-4. In *Findings of the Association for Computational Linguistics: ACL 2025*, 2025.

[Huang *et al.*, 2026] Hui Huang, Yancheng He, Hongli Zhou, Rui Zhang, Wei Liu, Weixun Wang, Jiaheng Liu, and Wenbo Su. Think-j: Learning to think for generative llm-as-a-judge. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(37):31158–31166, 2026.

[Hurst *et al.*, 2024] Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*, 2024.

[Jiang *et al.*, 2023] Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. Mistral 7b. *arXiv preprint arXiv:2310.06825*, 2023.

[Jiang *et al.*, 2024] Yuxin Jiang, Yufei Wang, Xingshan Zeng, Wanjun Zhong, Liangyou Li, Fei Mi, Lifeng Shang, Xin Jiang, Qun Liu, and Wei Wang. FollowBench: A multi-level fine-grained constraints following benchmark for large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics*, pages 4667–4688, 2024.

[Jiang *et al.*, 2025] Yuxin Jiang, Bo Huang, Yufei Wang, Xingshan Zeng, Liangyou Li, Yasheng Wang, Xin Jiang, Lifeng Shang, Ruiming Tang, and Wei Wang. Bridging and modeling correlations in pairwise data for direct preference optimization. In *International Conference on Learning Representations*, pages 85386–85405, 2025.

[Lambert *et al.*, 2025] Nathan Lambert, Valentina Pyatkin, Jacob Morrison, LJ Miranda, Bill Yuchen Lin, Khyathi Chandu, Nouha Dziri, Sachin Kumar, Tom Zick, Yejin Choi, et al. RewardBench: Evaluating reward models for language modeling. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 1755–1797, 2025.

[Liu *et al.*, 2024] Chris Yuhao Liu, Liang Zeng, Jiacai Liu, Rui Yan, Jujie He, Chaojie Wang, Shuicheng Yan, Yang Liu, and Yahui Zhou. Skywork-reward: Bag of tricks for reward modeling in llms. *arXiv preprint arXiv:2410.18451*, 2024.

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

[Liu *et al.*, 2025b] Yantao Liu, Zijun Yao, Rui Min, Yixin Cao, Lei Hou, and Juanzi Li. Rm-bench: Benchmarking reward models of language models with subtlety and style. In *International Conference on Learning Representations*, 2025.

[Lou *et al.*, 2024] Xingzhou Lou, Dong Yan, Wei Shen, Yuzi Yan, Jian Xie, and Junge Zhang. Uncertainty-aware reward model: Teaching reward models to know what is unknown. *arXiv preprint arXiv:2410.00847*, 2024.

[Malik *et al.*, 2026] Saumya Malik, Valentina Pyatkin, Sander Land, Jacob Morrison, Noah A. Smith, Hannaneh Hajishirzi, and Nathan Lambert. Rewardbench 2: Advancing reward model evaluation. In *International Conference on Learning Representations*, 2026.

[Park *et al.*, 2024] Junsoo Park, Seungyeon Jwa, Ren Meiying, Daeyoung Kim, and Sanghyuk Choi. OffsetBias: Leveraging debiased data for tuning evaluators. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 1043–1067, 2024.

[Peng *et al.*, 2025] Hao Peng, Yunjia Qi, Xiaozhi Wang, Zijun Yao, Bin Xu, Lei Hou, and Juanzi Li. Agentic reward modeling: Integrating human preferences with verifiable correctness signals for reliable reward systems. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics*, pages 15934–15949, 2025.

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

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

[Shen *et al.*, 2025] Jiaming Shen, Ran Xu, Yennie Jun, Zhen Qin, Tianqi Liu, Carl Yang, Yi Liang, Simon Baumgartner, and Michael Bendersky. RMBoost: Reward model training with preference-conditional multi-aspect synthetic data generation. In *Scaling Self-Improving Foundation Models without Human Supervision*, 2025.

[Team *et al.*, 2024] Gemma Team, Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, Léonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ramé, et al. Gemma 2: Improving open language models at a practical size. *arXiv preprint arXiv:2408.00118*, 2024.

[Touvron *et al.*, 2023] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.

[Wang *et al.*, 2024] Binghai Wang, Rui Zheng, Lu Chen, Zhiheng Xi, Wei Shen, Yuhao Zhou, Dong Yan, Tao Gui, Qi Zhang, and Xuanjing Huang. Reward modeling requires automatic adjustment based on data quality. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 4041–4064, 2024.

[Wang *et al.*, 2025] Chenglong Wang, Yang Gan, Yifu Huo, Yongyu Mu, Qiaozhi He, Murun Yang, Bei Li, Tong Xiao, Chunliang Zhang, Tongran Liu, and Jingbo Zhu. GRAM: A generative foundation reward model for reward generalization. In *International Conference on Machine Learning*, pages 62916–62936. PMLR, 2025.

[Yang *et al.*, 2024a] An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, et al. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2024.

[Yang *et al.*, 2024b] Rui Yang, Ruomeng Ding, Yong Lin, Huan Zhang, and Tong Zhang. Regularizing hidden states enables learning generalizable reward model for llms. In *Advances in Neural Information Processing Systems*, volume 37, pages 62279–62309, 2024.

[Yang *et al.*, 2025] An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

[Ye *et al.*, 2025] Zihuiwen Ye, Fraser David Greenlee, Max Bartolo, Phil Blunsom, Jon Ander Campos, and Matthias Gallé. Improving reward models with synthetic critiques. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 4506–4520, 2025.

[Yu *et al.*, 2025] Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Juncai Liu, et al. Dapo: An open-source llm reinforcement learning system at scale. In *Advances in Neural Information Processing Systems*, volume 38, pages 113222–113244, 2025.

[Yuan *et al.*, 2025] Lifan Yuan, Ganqu Cui, Hanbin Wang, Ning Ding, Xingyao Wang, Boji Shan, Zeyuan Liu, Jia Deng, Huimin Chen, Ruobing Xie, et al. Advancing llm reasoning generalists with preference trees. In *International Conference on Learning Representations*, 2025.

[Zhang *et al.*, 2025] Tao Zhang, ChengLIn Zhu, Yanjun Shen, Wenjing Luo, Yan Zhang, Hao Liang, Tao Zhang, Fan Yang, Mingan Lin, Yujing Qiao, et al. CFBench: A comprehensive constraints-following benchmark for LLMs. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics*, 2025.

[Zheng *et al.*, 2023] Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. Judging llm-as-a-judge with mt-bench and chatbot arena. In *Advances in Neural Information Processing Systems*, volume 36, pages 46595–46623, 2023.

[Zhou *et al.*, 2024] Hongli Zhou, Hui Huang, Yunfei Long, Bing Xu, Conghui Zhu, Hailong Cao, Muyun Yang, and Tiejun Zhao. Mitigating the bias of large language model evaluation. In *Proceedings of the 23rd Chinese National Conference on Computational Linguistics*, pages 1310–1319, 2024.

[Zhou *et al.*, 2026] Hongli Zhou, Hui Huang, Rui Zhang, Kehai Chen, Bing Xu, Conghui Zhu, Tiejun Zhao, and Muyun Yang. Toward robust llm-based judges: Taxonomic bias evaluation and debiasing optimization. *arXiv preprint arXiv:2603.08091*, 2026.