Crossing the Reward Bridge: Expanding Reinforcement Learning with Verifiable Rewards Across Diverse Domains

Yi Su[1,2], **Dian Yu**[2*], **Linfeng Song**[2], **Juntao Li**[1*], **Haitao Mi**[2], **Zhaopeng Tu**[2*], **Min Zhang**[1,3], **and Dong Yu**[2]

[1]Soochow University

[2]Tencent

[3]Key Laboratory of General Artificial Intelligence and Large Models, Soochow University

brunosu@tencent.com; yudian@global.tencent.com

{ljt,minzhang}@suda.edu.cn

Abstract

Reinforcement learning with verifiable rewards (RLVR) has been effective on tasks with structured solutions like math and coding, but its reliance on simple, rule-based verifiers creates a fundamental bottleneck. We find their applicability is surprisingly narrow even in structured domains, a limitation that is compounded at scale: rule-based systems can paradoxically degrade in performance as multi-domain, free-form training data increases. To overcome these challenges, we propose a new RLVR framework that uses a generative verifier to provide soft, probabilistic rewards. Our key insight is that powerful LLMs show high agreement with human evaluators when judging answer correctness given a ground-truth reference, allowing us to automate reward generation without costly human annotation. Our experiments demonstrate the effectiveness of this approach. We show that a compact 7B generative reward model can guide a 7B policy model to decisively outperform models up to 10x its size, including the 72B Qwen2.5-Instruct (by a margin of +8.6%). This effectiveness is robust, holding true across diverse training datasets with answers sourced from experts, web users, and other LLMs, and generalizes strongly to seven out-of-distribution benchmarks. Our work provides a scalable and effective framework for extending RLVR beyond the limitations of pattern-based verification to complex, noisy, real-world domains.

1 Introduction

Reinforcement learning with verifiable rewards (RLVR) is a powerful paradigm for improving the reasoning capabilities of large language models (LLMs)(Luong et al., 2024; Lambert et al., 2024; Guo et al., 2025). This approach has excelled in domains with structured solutions, such as math and coding, where binary correctness can be easily judged. However, its reliance on simple, rule-based verification has limited its application to more complex domains with free-form answers. Our analysis highlights this bottleneck, a limitation that worsens at scale when trained with cross-domain data with free-form answers: we find that as training data increases, the performance of rule-based systems can paradoxically degrade, revealing a critical flaw in conventional RLVR methods.

However, we identify a promising path to broaden RLVR's applicability. Our key observation is that when provided with an expert-written reference, powerful LLMs exhibit remarkable consistency in their correctness judgments. Specifically, we validate that proprietary models like GPT-4o and powerful open-source LLMs show near-perfect agreement (Cohen's κ >0.81), a finding that holds true when compared against human annotators. This insight opens the door to automating reward generation and reducing the dependence on the extensive, domain-specific human annotations traditionally required for reward modeling (Team et al., 2025).

Building on this observation, we extend RLVR to diverse, reasoning-intensive domains such as medicine, chemistry, economics, psychology (see detailed domains in Figure 2). Instead of relying on rigid binary signals, our framework uses a generative verifier to produce soft, probabilistic rewards. These granular reward signals provide a more effective learning signal for the model, especially where correctness may be partial or ambiguous.

Crucially, we demonstrate that a compact, 7B cross-domain reward model can be trained efficiently using only exploration-derived data and teacher model judgments, bypassing costly human annotation. Our experiments show that this enhanced RLVR framework, particularly the use of soft, probabilistic rewards, enables a 7B policy model to outperform not only baselines with rule-based rewards (+7.0%) but also powerful

* Corresponding authors

Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 3872–3892

models up to ten times its size, including the Qwen2.5-72B-Instruct (Team, 2024) (+8.6%) and DeepSeek-R1-Distill-Qwen-32B (Guo et al., 2025) (+9.5%). This effectiveness remains consistent across RL training datasets with varied answer sources, from experts (Yu et al., 2021) to general web users (Yue et al., 2024) and LLMs (Yuan et al., 2025). The resulting performance gains are not only substantial but also generalize robustly to seven out-of-distribution (OOD) reasoning tasks, such as MATH500 (Hendrycks et al., 2021b) and MMLU_Pro (Wang et al., 2024). Our work provides a scalable and effective framework for extending RLVR beyond the limitations of pattern-based verification to complex, noisy, real-world domains.

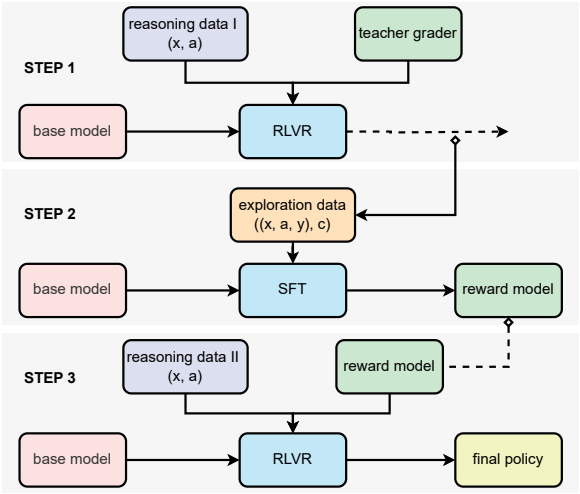

Figure 1: Overview paradigm of RLVR with our cross-domain verifier.

Contributions Our primary contributions are:

- We significantly broaden the applicability of RLVR beyond structured tasks, successfully adapting it for diverse, free-form reasoning domains such as medicine, law, and economics.
- We achieve this using a generative verifier that provides soft, probabilistic rewards. This framework is highly efficient, relying on a compact 7B reward model trained without costly, domain-specific human annotation.
- We demonstrate the effectiveness of our approach through consistent gains on training data from diverse sources (experts, web users, and LLMs) and strong generalization to seven OOD benchmarks.
- We will release our 570K-example multi-domain dataset with expert-written answers and our trained 7B reward model to facilitate future research.

2 Method

Figure 1 presents the overview paradigm of our approach. We consider a setting where each prompt x is paired with an expert-written reference answer a. Such references have proven crucial for providing accurate rewards in RL on reasoning-intensive tasks such as coding and mathematics (Shao et al., 2024), and help mitigate reward hacking (Mroueh, 2025). Ideally, in these domains, a response y can be objectively verified against the reference answer a. In practice, however, this verification may be affected by factors such as imperfect answer extraction and matching when rule-based verifiers are used, as well as noise introduced by automated evaluation systems like a reward model $r_\phi(x, a, y)$.

Nevertheless, we can still use this verifiable reward in a policy gradient algorithm, with REINFORCE (Williams, 1992) as an example:

$$J(\theta) = \mathbb{E}_{(x,a)\sim D}\, \mathbb{E}_{y_i\sim\pi_\theta(\cdot|x)} \Big[r_\phi(x, a, y_i) \Big]. \quad (1)$$

When the generation of an entire response is modeled as a single action (Ahmadian et al., 2024), the gradient becomes (see Section A.4 for details):

$$\nabla_\theta J(\theta) = \mathbb{E}_{(x,a)\sim D}\ \mathbb{E}_{y_i\sim\pi_\theta(\cdot|x)} \Big[r_\phi(x, a, y_i) \nabla_\theta \log \pi_\theta(y_i \mid x) \Big]. \quad (2)$$

2.1 Reward Estimation

Or the binary reward signal, we constrain a generative LLM π_ϕ to output only 0 or 1 through specific instructions in its prompt (see system prompt in Table 6). For notational simplicity, we assume that each response consists of exactly T steps, where each step corresponds to a non-empty line. Following a common practice for reasoning tasks, our system prompt requires the model to output its reasoning followed by a conclusive answer summarized in the final line of the response. For a response y_i, we denote this final line containing the answer as y_i^T. The binary model-based reward function is then defined as:

$$r_\phi(x, a, y_i) = \mathbb{1}\big(c_i = 1\big), \quad (3)$$

where c_i is sampled from $\pi_\phi(\cdot \mid x, a, y_i^T)$, representing π_ϕ's judgment on the correctness of y_i.

While the binary reward relies on a single sampled judgment, we can also define a soft reward function that uses the verifier's probability distribution over the judgment tokens (i.e., 0 or 1):

$$r_\phi(x, a, y_i) = \frac{\pi_\phi(1 \mid x, a, y_i^T)}{\pi_\phi(0 \mid x, a, y_i^T) + \pi_\phi(1 \mid x, a, y_i^T)} \tag{4}$$

As shown in Equations 3 and 4, $r_\phi(x, a, y_i)$ is bounded within $[0, 1]$, ensuring consistency with the widely adopted binary rule-based reward scale in RLVR. We set $r_\phi(x, a, y_i) = 0$ whenever $\pi_\phi(0 \mid x, a, y_i^T) + \pi_\phi(1 \mid x, a, y_i^T) = 0$.

2.2 Reward Model Training

When considering generative verifiers, a natural choice is to use an off-the-shelf, aligned LLM as the reward model π_ϕ, inspired by prior work that employs LLMs as judges (Zheng et al., 2023). However, we observe a noticeable performance gap on downstream tasks when using LLMs of different sizes. For example, the 72B reward model achieves 3.5% improvement compared to its 7B counterpart (see training details and Figure 4 in Sections 3 and 4). To address this trade-off, we explore training a moderately sized reward model (e.g., 7B) for diverse-domain use, aiming to balance performance and efficiency.

Since there are no ground-truth reward labels, for each (x, a, y) triple, we prompt a fixed LLM to obtain the binary judgments $c \in \{0, 1\}$, indicating whether y matches the reference answer a. During the RL phase, we collect the data $\{((x, a, y), c)\}$ from the exploration stages and use it to fine-tune our reward models with the cross-entropy loss over c. Unlike relying on a fixed LLM to generate y, the improving actor policy produces responses with varying performance and potential formatting noise, which may enhance the robustness of the trained reward models.

3 Experimental Setting

3.1 Data

Multi-Subject Data Since no publicly available large-scale, free-form dataset with objective reference answers exists across general domains, we use ExamQA (Yu et al., 2021), a multi-subject multiple-choice question answering (QA) dataset originally

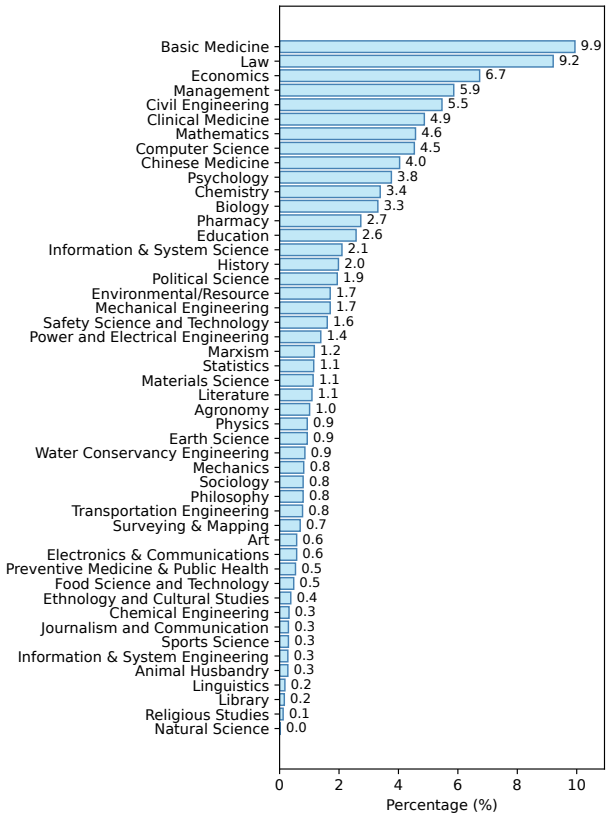

Figure 2: Distribution of subject occurrences in the test set of ExamQA (excluding unclassified).

written in Chinese. Spanning 48+ first-level subjects, ExamQA contains 638K **college-level** examples, with questions and answers written by domain experts for examination purposes. Unlike resources such as web forum data (Yue et al., 2024), where answer quality and objectivity can vary, ExamQA offers standardized, curriculum-aligned answers that are accurate, concise, and pedagogically sound, ideal for RLVR training and evaluation. We remove the distractor options and convert each instance into a free-form QA pair, then use GPT-4o-mini to translate the data into English.

For evaluation, we randomly sample 6,000 questions from ExamQA as the test set, while the remaining questions are used as the training pool. Since subject labels are not provided for each QA pair, we use GPT-4o-mini to classify them into one of 48 subjects or mark them as unclassified if uncertain. The detailed classification prompt is provided in Table 7. Excluding unclassified instances (15.8% of the test data), the most frequent subjects include basic medicine, law, economics, management, civil engineering, mathematics, computer science and technology, psychology, and chemistry, as shown in Figure 2. For ease of analysis, we further cate-

gorize these subjects into four broad fields (STEM, social sciences, humanities, and applied sciences) as detailed in Table 8. See examples in Table 11.

Data for Training the Reward Model To generate training data for our reward model, we first select 40K examples from the ExamQA training set. Using the methodology in Section 2.2, we employ Qwen2.5-7B (Team, 2024) to conduct RL training. We use the RLOO (Kool et al., 2019; Ahmadian et al., 2024) algorithm and generate four online samples for each prompt. We use Qwen2.5-72B-Instruct as the teacher reward model for hard (binary) label determination. This model was chosen as our teacher due to its strong performance as a zero-shot verifier and its high agreement with both GPT-4o and human evaluators, as established in our evaluation (Section 3.3). By preserving all input-output pairs, this process yields 160K distilled training samples from Qwen2.5-72B-Instruct for reward model training. We fine-tune Qwen2.5-7B-Instruct (Team, 2024) using SFT on the resulting data to train our reward models.

To verify the training approach's validity, we exclude these 40K original samples from the final training dataset. This strict separation ensures that the reward model never encounters any data used in previous training stages, thereby guaranteeing evaluation objectivity.

3.2 Model

We use Qwen2.5-7B (Team, 2024) as the base model for all our RL experiments. We use different RL algorithms to validate the robustness and effectiveness of our method, including REINFORCE (Williams, 1992; Ahmadian et al., 2024), RLOO (Kool et al., 2019; Ahmadian et al., 2024), and GRPO (Shao et al., 2024). Following Stiennon et al. (2020); Ouyang et al. (2022); Hu (2025), we also consider a Kullback-Leibler divergence penalty between the policy model and the reference policy distributions to mitigate bias in the reward model and batch reward normalization inspired by prior studies such as GRPO (Shao et al., 2024) and REINFORCE++ (Hu, 2025). However, in our experiments, these two modifications do not lead to clear performance gains.

We consider two types of rewards:

- **Binary:** a reward $\in \{0, 1\}$, either from conventional rule-based methods (exact match on the final answer) or predicted by a reward model.
- **Soft:** the soft reward in the range $[0, 1]$ produced by a reward model.

We also compare our trained reward model, RM-7B, with its teacher, Qwen2.5-72B-Instruct (Team, 2024), which functions as a larger generative RM. See the training hyper-parameters of RL distilled data collection, reward model training, and the main experiments in the Appendix (Table 5).

3.3 Evaluation

We first investigate majority voting using a strong open-source LLM, Qwen2.5-72B-Instruct (Team, 2024), as the reward model π_ϕ. The evaluation process follows the prompting template provided in Table 6. Given a prompt x and a reference answer a, we generate m evaluation samples and determine the correctness of a response y via majority vote. Specifically, a response is considered correct if at least half of the sampled judgments classify it as such: $\sum_{j=1}^{m} \mathbb{1}\left[\pi_\phi^{(j)}(x, y^T, a) = 1\right] \geq \frac{m}{2}$. To assess this method's reliability, we compare the majority-voted judgments from Qwen2.5-72B-Instruct against those of GPT-4o, one of the most capable proprietary LLMs, which provides a single evaluation per response. Agreement is measured using Cohen's Kappa (κ). As shown in Figure 5, the two evaluation methods exhibit almost perfect agreement ($0.81 \leq \kappa \leq 1.00$), with κ exceeding 0.87 for multi-subject college-level problems and 0.85 for pre-college mathematics.

We further validate the reliability of the automated evaluations by comparing them against human annotations on randomly selected 500 instances, where both LLMs exhibit strong agreement with human judgments (more details in Appendix A.2 and Appendix A.3). This high level of agreement remains consistent across different values of m, indicating that results are robust to the number of sampled judgments. We also provide a granular view of verifier reliability across domains (Applied Sciences, Humanities, STEM, Social Sciences, and Unclassified) (Table 15). Based on this observation, we adopt $m = 1$ in all subsequent evaluations to improve efficiency without compromising evaluation quality.

4 Experimental Results

4.1 Main Results on Multi-Subject Tasks

Table 1 shows the results on the multi-subject dataset, which is notably challenging with even

Method	Reward Model	Score Type	Multi-Subject					
			STEM	Social	Humanities	Applied	Others	Avg
Qwen2.5-7B$_{\text{Instruct}}$ (Team, 2024)			18.5	15.0	15.8	14.7	14.6	16.0
Phi-4-14B (Abdin et al., 2024)			20.6	15.3	19.1	14.2	14.0	16.7
Llama3.1-70B$_{\text{Instruct}}$ (Grattafiori et al., 2024)			19.8	16.4	21.5	16.8	15.8	17.7
Qwen2.5-72B$_{\text{Instruct}}$ (Team, 2024)			25.2	20.1	28.7	20.5	21.0	22.6
DeepSeek-R1-Qwen-32B (Guo et al., 2025)			23.2	21.8	26.7	20.5	18.5	21.7
Vanilla (Qwen2.5-7B)			16.3	14.9	15.2	13.3	14.8	15.0
+ SFT			24.6	22.8	25.7	20.9	22.6	23.1
REINFORCE	Rule-based	binary	25.3	26.6	27.7	21.1	20.7	24.2
	Qwen2.5-**72**B$_{\text{Instruct}}$	binary	27.9	27.9	30.7	24.4	23.2	26.6
		soft	32.2	**32.8**	**36.0**	24.9	**27.9**	30.3
	RM-**7**B	binary	29.0	29.1	28.4	23.8	24.8	27.3
		soft	**32.7**	**32.8**	35.6	**28.6**	27.4	**31.2**
RLOO	Rule-based	binary	28.2	27.9	27.4	22.4	24.5	26.3
	Qwen2.5-**72**B$_{\text{Instruct}}$	binary	29.4	30.5	33.7	24.6	26.1	28.4
		soft	**32.9**	31.4	34.7	**27.7**	26.8	**30.6**
	RM-**7**B (ours)	binary	29.3	29.0	33.3	25.8	25.6	28.1
		soft	31.0	**32.0**	**35.6**	27.0	**27.0**	30.0
GRPO	Rule-based	binary	24.9	22.5	23.8	20.7	20.2	22.6
	Qwen2.5-**72**B$_{\text{Instruct}}$	binary	26.3	26.6	28.7	24.7	22.2	25.5
		soft	28.1	27.4	**32.0**	25.8	24.0	27.0
	RM-**7**B (ours)	binary	26.7	23.9	27.4	23.8	24.3	25.0
		soft	**31.4**	**31.6**	31.4	**29.1**	**27.5**	**30.3**

Table 1: Performance comparison of different methods on the multi-subject tasks in ExamQA.

Model	Mathematical Benchmarks						General Benchmarks			
	AMC	GSM8K	MATH	Mine.	Olym.	Avg.	GPQA	GPQA^{Sup}	MMLU^{Pro}	Avg.
Vanilla (Qwen2.5-7B)	25.3	81.9	52.6	15.4	19.4	38.9	22.1	18.1	40.3	26.8
+Rule-based Reward	37.3	88.6	65.6	26.8	29.5	49.6	26.6	22.7	45.9	31.7
+Qwen2.5-72B$_{\text{Instruct}}$	32.5	87.1	68.0	25.7	32.9	49.2	27.5	25.5	50.0	**34.3**
+RM-7B (ours)	37.3	88.9	67.0	28.7	31.0	**50.6**	26.6	25.5	50.0	34.0

Table 2: Adaptation results of policies on out-of-distribution mathematical and general-domain benchmarks. Policies are trained on our multi-domain data using either rule-based (binary) or model-based (soft) rewards.

strong open-source models achieving relatively low accuracies. We argue that incorporating such tasks into the RLVR setting is crucial for fostering deeper investigation and accelerating progress in this promising area.

RLVR with model-based soft rewards delivers the largest overall gains across all domains. Using our RM-7B as the reward model and the REINFORCE trainer, the *soft* reward setting reaches 31.2% average accuracy, surpassing RLVR with binary rule-based rewards by +7.0%, the strongest supervised baseline (SFT, 23.1%) by +8.1%, and the vanilla 7B model by +16.2%. Gains are consistent across all subject clusters (STEM :+7.4%, Social :+6.2%, Humanities :+7.9%, Applied :+7.5%, Others :+6.7%) over rule-based reward settings, confirming that the proposed RLVR extension scales beyond strictly structured tasks and supports our first contribution claim.

Soft scoring consistently outperforms hard binary scoring. Across all trainers (REINFORCE, RLOO, and GRPO) and reward models (RM-7B

and Qwen2.5-72B), switching to soft rewards provides a consistent performance boost of +2.2% to +5.0%, with the largest gains in less structured Social and Humanities categories. These observations corroborate our second contribution: soft, model-based rewards provide richer, more informative feedback that is especially valuable in free-form answer settings.

Extended RLVR decisively outperforms strong, much larger, instruction-tuned models. The best 7B RLVR model surpasses Qwen2.5-72B-Instruct (22.6%) by +8.6% and even outperforms DeepSeek-R1-Qwen-32B (21.7%) by +9.5%, establishing a new state of the art among open-source models on this multi-domain benchmark. The margin is largest in Humanities (+6.9%) and STEM (+7.5%), underscoring the practical relevance of our approach for knowledge-intensive applications.

A compact cross-domain reward model rivals or even surpasses a 10× larger teacher. Despite being trained on noisy exploration data and having only 7B parameters, RM-7B matches or exceeds its 72B teacher in three of the six macro metrics under REINFORCE (Avg :+0.9%) and outperforms it clearly under GRPO (Avg :+3.3%). This validates our third contribution, demonstrating the feasibility of distilling reliable, cross-domain reward functions into modestly sized networks without additional human annotation.

All three RL algorithms benefit from model-based rewards. REINFORCE attains the highest overall score (31.2%), yet both RLOO (30.0%) and GRPO (30.3%) also substantially outperform SFT baselines, indicating that the gains stem primarily from the quality of the reward rather than a specific policy-optimization recipe. This robustness under different algorithms further highlights the scalability of the proposed RLVR framework.

4.2 Generalization to Other Datasets

To evaluate if a reward model trained on expert-written academic data can generalize to other styles of free-form content, we test it on two additional multi-domain free-form datasets: NaturalReasoning (NR) (Yuan et al., 2025) and WebInstruct (Web) (Yue et al., 2024). Examples for each dataset is provided in Appendix (Table 12 and 13). We compare policies trained with rule-based binary or model-based soft rewards. For each data, we randomly sample 30K examples for training and 5K for evaluation. The model-based reward setting yields over 10% improvement on both datasets (Table 3), showing strong generalization to noisy settings where answers may be written by web users (Web) or generated by LLMs (NR).

RL Algorithm	Reward Model	NR	Web
RLOO	Rule-based	29.4	33.9
	RM-7B (ours)	39.8	44.0
REINFORCE	Rule-based	27.1	34.7
	RM-7B (ours)	35.0	42.3
GRPO	Rule-based	28.1	33.1
	RM-7B (ours)	**44.0**	**49.3**

Table 3: Accuracy (%) on multi-subject, long-form QA datasets . Soft rewards from RM-7B generalize across RL algorithms, consistently outperforming baselines with rule-based rewards.

We further evaluate the policies trained on multi-subject data using RLOO with different reward functions, without applying any additional fine-tuning. These evaluations are conducted across several out-of-distribution (OOD), reasoning-intensive benchmarks, which are grouped into two primary categories: (1) *Mathematical Benchmarks*: AMC 23, GSM8K (Cobbe et al., 2021), MATH500 (Hendrycks et al., 2021b), Minerva (Lewkowycz et al., 2022), and Olympiads (He et al., 2024); and (2) *General Reasoning Benchmarks*: GPQA Diamond (Rein et al., 2024), Super GPQA (Du et al., 2025), and MMLU_Pro (Wang et al., 2024). The results presented in Table 2 indicate that policies optimized using our model-based rewards consistently exhibit robust generalization capabilities across diverse reasoning tasks.

Policies trained with soft, model-based rewards transfer best to out-of-distribution benchmarks. Table 2 shows that a single policy learned on our multi-subject corpus with RM-7B rewards attains the highest average accuracy on five OOD math sets (50.6%) and nearly ties the best score on three challenging general-knowledge sets (34.0%). Relative to the vanilla 7B baseline, this corresponds to +11.7% and +7.2% absolute improvements, respectively. Crucially, the same policy even edges out (Math Avg: +1.4%) or matches (General Avg) the policy trained with the 10× larger Qwen2.5-72B reward model, confirming that reliable cross-domain knowledge can be distilled into a compact verifier.

Our approach consistently offers better generalization than rule-based rewards across diverse

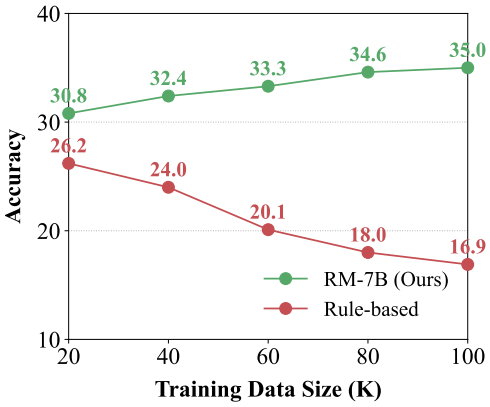

Figure 3: Impact of training data size on ExamQA.

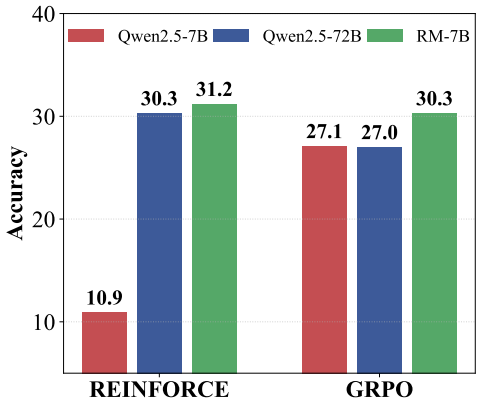

Figure 4: Impact of RMs on policy accuracy (ExamQA).

OOD benchmarks, reinforcing the advantages of nuanced feedback. Across both math and general OOD datasets, policies trained with soft rewards from our RM-7B generally outperform the policy trained with binary rule-based rewards. Specifically, RM-7B surpasses Rule-based by 1.0% on mathematical benchmarks, and by 2.3% on general benchmarks. This pattern highlights that the more nuanced feedback provided by soft, model-based rewards not only benefits in-domain performance but also translates to improved adaptability and robustness on unseen tasks, aligning with our second contribution regarding the superiority of soft rewards for generalization and scalability.

4.3 Analysis

In this section, we provide some insights into how our approach improves performance.

Our approach demonstrates superior scalability over rule-based rewards. Scalability remains a critical challenge in RL. A key question is whether model performance continues to improve as RL training progresses and the data grows. To examine this, we conduct experiments using our trained reward model against rule-based reward while progressively scaling the dataset. We randomly sampled 100K samples from our training corpus as the scaling set and evaluate performance on the same multi-subject test set.

The results in Figure 3 clearly show that as the amount of training data increases from 20K to 100K, the performance of our RM-7B reward model consistently improves, rising from 30.8% to 35.0%. In contrast, the rule-based reward function exhibits a declining trend, dropping from 26.2% to 16.9% across the same data scale. This divergence highlights the fundamental limitations of rule-based reward mechanisms, which fail to adapt effectively as tasks and data scale, especially in complex multi-subject settings. Meanwhile, the learned reward model robustly captures more nuanced feedback and generalizes better with larger datasets. The results support the effectiveness of our method in real-world conditions where data availability increases over time and reward interpretation needs to be dynamic and context-aware.

Larger reward models improve policy quality, but our RM-7B offers a strong alternative. The choice of a teacher model for generating reward signals strongly impacts the performance of the RL policy, as shown in Figure 4. Generally, employing larger, well-aligned instruction-following models as reward sources is expected to yield better policy outcomes. For instance, policies trained with rewards from the Qwen2.5-72B-Instruct model substantially outperform those trained with rewards from the smaller Qwen2.7-7B-Instruct model when using REINFORCE (30.3% vs. 10.9%) and RLOO (30.6% vs. 0.3%). This general trend is often because smaller aligned LLMs, like Qwen2.7-7B-Instruct, can exhibit underconfidence; they may assign unduly low probabilities to correct responses, resulting in noisy or uninformative reward signals even when their discrete, binary judgments might be accurate. Remarkably, our RM-7B model, despite being considerably smaller than Qwen2.5-72B, demonstrates highly competitive or even superior performance. Specifically, RM-7B achieves a score of 31.2% against Qwen2.5-72B's 30.3% with REINFORCE, performs competitively with RLOO (30.0% for RM-7B vs. 30.6% for Qwen2.5-72B), and notably outperforms with GRPO (30.3% for RM-7B vs. 27.0% for Qwen2.5-72B). This strong performance from a more compact model under-

scores the success of our specific training phase in developing a robust and efficient reward model capable of effectively guiding RLVR to achieve high-quality policy outcomes.

5 Related Work

Reward Estimation in RLVR For reasoning tasks such as math reasoning, a solution is typically considered correct if it arrives at correct final answer (Cobbe et al., 2021). This is because reliably assessing the correctness of individual steps remains an open challenge. Similarly, the correctness of solutions to coding problems is typically accessed based on whether all test cases pass (Austin et al., 2021; Hendrycks et al., 2021a; Gehring et al., 2024). Therefore, previous RLVR studies have mainly focused on math and coding tasks.

In most previous studies (Zelikman et al., 2022; Gandhi et al., 2024; Zhang et al., 2024b; Lambert et al., 2024; Guo et al., 2025; Ma et al., 2025a; Yu et al., 2025), given access to the reference answer a, the correctness label z for a response y to a prompt x is typically a binary value. z can also take on a value in the range $[0, 1]$ to reflect varying degrees of correctness (Luong et al., 2024; Li et al., 2024; Ma et al., 2025b; Xie et al., 2025; Chen et al., 2025). Labels are assigned by a deterministic function $z = f(x, y, a)$, which operates based on predefined rules (e.g., exact match). These rules can also be combined with tools, such as a Python library, for verification (Xiong et al., 2025; Luo et al., 2025). This method is particularly effective when the answer type is fixed and easily matchable, such as a numerical value or a multiple-choice option. Each response is rated individually, without considering any preference information.

Besides using closed-source LLMs such as GPT-4o as verifiers (Chen et al., 2024), recent studies have also explored training reference-based reward models for mathematical reasoning (Team et al., 2025). However, these models are confined to a single domain and still require large-scale training data (e.g., 800k instances for math).

Generative Reward Modeling Using next-token prediction for reward modeling has attracted great interest in recent years (Lightman et al., 2023; Zheng et al., 2023; Tian et al., 2024; Zhang et al., 2024a), as it enables LLMs to fully leverage their generative capabilities, not only to produce accurate rewards but also to provide rationales that justify their judgments. In this work, we explore applying generative, reference-based verifiers to reinforcement learning and investigate their effectiveness across a variety of domains, an area that remains largely underexplored.

Furthermore, we explore training generative reward models without the need for annotated or synthetic step-by-step rationales (Team et al., 2025; Zhang et al., 2024a) to justify the final assessment. Specifically, we leverage the confidence of generative verifiers to provide stable and informative reward signals, enhancing the robustness of RL training in the presence of noise and ambiguity.

Verifiable Reasoning Data Most RLVR studies focus on narrow tasks (Liu and Zhang, 2025; Xie et al., 2025) such as math word problem solving, code generation, and logic puzzles, where short, structured reference answers allow for simple, rule-based verification. For example, SimpleRL (Zeng et al., 2025) and Tulu (Lambert et al., 2024) use math datasets GSM8K (Cobbe et al., 2021) and MATH (Hendrycks et al., 2021b), in which each reference answer typically consists of fewer than two words. However, this reliance on well-structured data constrains the scale and diversity of resources that can be used for RLVR across broader domains. This work explores RLVR using multi-domain reasoning-intensive data, where the free-form reference answers are written by domain experts for unbiased evaluation (Yu et al., 2021). We also consider two additional multi-domain datasets, WebInstruct (Yue et al., 2024) and NaturalReasoning (Yuan et al., 2025), which provide reference answers either extracted from pre-training corpora written by web users or generated by LLMs, respectively. While the two datasets contain longer answers with varied or uniform styles, the quality and objectivity of references may differ. Nevertheless, we observe trends consistent with our main observations. Additional details are provided in the Appendix A.6.

6 Conclusions

This work extends Reinforcement Learning with Verifiable Rewards (RLVR) to diverse domains that require nuanced, free-form answers. By replacing rigid, rule-based verifiers with a generative model that provides soft, probabilistic rewards, our framework achieves significant gains in robustness, scalability, and generalization. We demonstrate that an effective cross-domain reward model can be distilled into a compact 7B model without costly

human annotation. The results are compelling: our 7B policy decisively outperforms powerful, well-aligned open-source models up to 10 times its size on challenging multi-domain reasoning tasks.

Limitations

On the Use of Format-Based Rewards This work does not incorporate format-based rewards (Guo et al., 2025; Xie et al., 2025). We revisit the role of format-related constraints and rewards in this context. Prior work often uses pattern-based functions for scoring, making it critical to guide LLMs to enclose final answers in a machine-parsable format. These extracted answers are then compared to reference answers for verification and evaluation. However, recent studies show that rigid format-based rewards may hinder exploration, especially when training from base models (Zeng et al., 2025). In contrast, by reintroducing a reward model in RLVR without imposing any format constraints on reference answers or model outputs, we reduce the need for extensive human effort in data standardization and pattern design. We view this as part of a broader shift toward format-agnostic reward modeling in RLVR, which invites further research on balancing structural guidance and expressive flexibility.

To CoT or Not to CoT for Verifiers in RLVR In this work, we simplify the verification task by instructing a generative reward model to output either 1 or 0, without requiring chain-of-thought (CoT) reasoning (Nye et al., 2021; Wei et al., 2022). While CoT has proven useful in both reference-based (Team et al., 2025) and reference-free (Zhang et al., 2024a) settings, it remains an open question how essential in-depth rationales are for assessing semantic equivalence between reference answers and model responses in the same language, particularly when focusing on the conclusive part of each response. We explored using RL to train the RM to encourage more "talkative" behavior (Team et al., 2025) in the absence of CoT data for supervision. However, this underperforms compared to using SFT for distillation. This also raises a broader question in process reward modeling (Lightman et al., 2023): how should rewards be assigned in the absence of direct supervision for intermediate steps, regardless of the segmentation strategy?

Scalability of Data and Model Sizes Due to computational constraints, all training in our experiments is performed on 7B-parameter models. Nonetheless, we observe consistent trends across both Qwen and DeepSeek-7B (Shao et al., 2024) models (Table 10 in the Appendix). Extending RLVR to larger models and more comprehensive multi-subject or multi-domain datasets is a natural next step, and we anticipate that scaling studies will play a critical role in advancing the field.

Scope of Reference-Based Judgments of LLMs Our observation of high agreement in binary reference-based judgments among LLMs focuses on relatively short free-form reference answers (approximately four words) for college-level multi-subject questions, and moderately long reference answers (30–50 words) for pre-college mathematical tasks (Section 3.3). While our study focuses on specific answer lengths and task domains, large-scale evaluation of human–LLM agreement remains an open and important challenge for the broader research community. In addition, broadening this scope to longer, more complex reference answers, particularly in diverse domains and advanced tasks such as IMO-style proofs, represents a promising direction for future work.

Acknowledgments

We want to thank all the anonymous reviewers for their valuable comments. We thank Min Zhang and Dong Yu for their advising. This work was supported by the National Science Foundation of China (NSFC No. 62576232), Key Laboratory of General Artificial Intelligence and Large Models in Provincial Universities, Soochow University.

References

Marah Abdin, Jyoti Aneja, Harkirat Behl, Sébastien Bubeck, Ronen Eldan, Suriya Gunasekar, Michael Harrison, Russell J Hewett, Mojan Javaheripi, Piero Kauffmann, and 1 others. 2024. Phi-4 technical report. *arXiv preprint arXiv:2412.08905*.

Arash Ahmadian, Chris Cremer, Matthias Gallé, Marzieh Fadaee, Julia Kreutzer, Olivier Pietquin, Ahmet Üstün, and Sara Hooker. 2024. Back to basics: Revisiting reinforce style optimization for learning from human feedback in llms. *arXiv preprint arXiv:2402.14740*.

Jacob Austin, Augustus Odena, Maxwell Nye, Maarten Bosma, Henryk Michalewski, David Dohan, Ellen Jiang, Carrie Cai, Michael Terry, Quoc Le, and 1 others. 2021. Program synthesis with large language models. *arXiv preprint arXiv:2108.07732*.

Junying Chen, Zhenyang Cai, Ke Ji, Xidong Wang, Wanlong Liu, Rongsheng Wang, Jianye Hou, and Benyou Wang. 2024. Huatuogpt-o1, towards medical complex reasoning with llms. *arXiv preprint arXiv:2412.18925*.

Mingyang Chen, Tianpeng Li, Haoze Sun, Yijie Zhou, Chenzheng Zhu, Fan Yang, Zenan Zhou, Weipeng Chen, Haofen Wang, Jeff Z Pan, and 1 others. 2025. Learning to reason with search for llms via reinforcement learning. *arXiv preprint arXiv:2503.19470*.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, and 1 others. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

Xinrun Du, Yifan Yao, Kaijing Ma, Bingli Wang, Tianyu Zheng, King Zhu, Minghao Liu, Yiming Liang, Xiaolong Jin, Zhenlin Wei, and 1 others. 2025. Supergpqa: Scaling llm evaluation across 285 graduate disciplines. *arXiv preprint arXiv:2502.14739*.

Kanishk Gandhi, Denise HJ Lee, Gabriel Grand, Muxin Liu, Winson Cheng, Archit Sharma, and Noah Goodman. 2024. Stream of search (sos): Learning to search in language. In *First Conference on Language Modeling*.

Jonas Gehring, Kunhao Zheng, Jade Copet, Vegard Mella, Taco Cohen, and Gabriel Synnaeve. 2024. Rlef: Grounding code llms in execution feedback with reinforcement learning. *arXiv preprint arXiv:2410.02089*.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, and 1 others. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

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*.

Chaoqun He, Renjie Luo, Yuzhuo Bai, Shengding Hu, Zhen Leng Thai, Junhao Shen, Jinyi Hu, Xu Han, Yujie Huang, Yuxiang Zhang, and 1 others. 2024. Olympiadbench: A challenging benchmark for promoting agi with olympiad-level bilingual multimodal scientific problems. *arXiv preprint arXiv:2402.14008*.

Dan Hendrycks, Steven Basart, Saurav Kadavath, Mantas Mazeika, Akul Arora, Ethan Guo, Collin Burns, Samir Puranik, Horace He, Dawn Song, and 1 others. 2021a. Measuring coding challenge competence with apps. *arXiv preprint arXiv:2105.09938*.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. 2021b. Measuring mathematical problem solving with the math dataset. *Preprint*, arXiv:2103.03874.

Jian Hu. 2025. Reinforce++: A simple and efficient approach for aligning large language models. *arXiv preprint arXiv:2501.03262*.

Wouter Kool, Herke van Hoof, and Max Welling. 2019. Buy 4 reinforce samples, get a baseline for free!

Nathan Lambert, Jacob Morrison, Valentina Pyatkin, Shengyi Huang, Hamish Ivison, Faeze Brahman, Lester James V Miranda, Alisa Liu, Nouha Dziri, Shane Lyu, and 1 others. 2024. T\" ulu 3: Pushing frontiers in open language model post-training. *arXiv preprint arXiv:2411.15124*.

Aitor Lewkowycz, Anders Andreassen, David Dohan, Ethan Dyer, Henryk Michalewski, Vinay Ramasesh, Ambrose Slone, Cem Anil, Imanol Schlag, Theo Gutman-Solo, and 1 others. 2022. Solving quantitative reasoning problems with language models. *Advances in Neural Information Processing Systems*, 35:3843–3857.

Long Li, Xuzheng He, Haozhe Wang, Linlin Wang, and Liang He. 2024. How do humans write code? large models do it the same way too. *arXiv preprint arXiv:2402.15729*.

Hunter Lightman, Vineet Kosaraju, Yura Burda, Harri Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. 2023. Let's verify step by step. *arXiv preprint arXiv:2305.20050*.

Jiawei Liu and Lingming Zhang. 2025. Code-r1: Reproducing r1 for code with reliable rewards.

Michael Luo, Sijun Tan, Justin Wong, Xiaoxiang Shi, William Y. Tang, Manan Roongta, Colin Cai, Jeffrey Luo, Tianjun Zhang, Li Erran Li, Raluca Ada Popa, and Ion Stoica. 2025. Deepscaler: Surpassing o1-preview with a 1.5b model by scaling rl. Notion Blog.

Trung Quoc Luong, Xinbo Zhang, Zhanming Jie, Peng Sun, Xiaoran Jin, and Hang Li. 2024. Reft: Reasoning with reinforced fine-tuning. *arXiv preprint arXiv:2401.08967*.

Ruotian Ma, Peisong Wang, Cheng Liu, Xingyan Liu, Jiaqi Chen, Bang Zhang, Xin Zhou, Nan Du, and Jia Li. 2025a. S2r: Teaching llms to self-verify and self-correct via reinforcement learning. *arXiv preprint arXiv:2502.12853*.

Zexiong Ma, Chao Peng, Pengfei Gao, Xiangxin Meng, Yanzhen Zou, and Bing Xie. 2025b. Sorft: Issue resolving with subtask-oriented reinforced fine-tuning. *arXiv preprint arXiv:2502.20127*.

Youssef Mroueh. 2025. Reinforcement learning with verifiable rewards: Grpo's effective loss, dynamics, and success amplification. *arXiv preprint arXiv:2503.06639*.

Maxwell Nye, Anders Johan Andreassen, Guy Gur-Ari, Henryk Michalewski, Jacob Austin, David Bieber, David Dohan, Aitor Lewkowycz, Maarten Bosma, David Luan, and 1 others. 2021. Show your work: Scratchpads for intermediate computation with language models.

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.

David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R Bowman. 2024. Gpqa: A graduate-level google-proof q&a benchmark. In *First Conference on Language Modeling*.

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

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

Kimi Team, Angang Du, Bofei Gao, Bowei Xing, Changjiu Jiang, Cheng Chen, Cheng Li, Chenjun Xiao, Chenzhuang Du, Chonghua Liao, and 1 others. 2025. Kimi k1. 5: Scaling reinforcement learning with llms. *arXiv preprint arXiv:2501.12599*.

Qwen Team. 2024. Qwen2.5: A party of foundation models.

Ye Tian, Baolin Peng, Linfeng Song, Lifeng Jin, Dian Yu, Lei Han, Haitao Mi, and Dong Yu. 2024. Toward self-improvement of llms via imagination, searching, and criticizing. In *Advances in Neural Information Processing Systems*, volume 37, pages 52723–52748. Curran Associates, Inc.

Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, and 1 others. 2024. Mmlu-pro: A more robust and challenging multi-task language understanding benchmark. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, and 1 others. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.

Ronald J Williams. 1992. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8:229–256.

Tian Xie, Zitian Gao, Qingnan Ren, Haoming Luo, Yuqian Hong, Bryan Dai, Joey Zhou, Kai Qiu, Zhirong Wu, and Chong Luo. 2025. Logic-rl: Unleashing llm reasoning with rule-based reinforcement learning. *arXiv preprint arXiv:2502.14768*.

Wei Xiong, Hanning Zhang, Chenlu Ye, Lichang Chen, Nan Jiang, and Tong Zhang. 2025. Self-rewarding correction for mathematical reasoning. *Preprint*, arXiv:2502.19613.

Dian Yu, Kai Sun, Dong Yu, and Claire Cardie. 2021. Self-teaching machines to read and comprehend with large-scale multi-subject question-answering data. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 56–68, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Tiantian Fan, Gaohong Liu, Lingjun Liu, Xin Liu, and 1 others. 2025. Dapo: An open-source llm reinforcement learning system at scale. *arXiv preprint arXiv:2503.14476*.

Weizhe Yuan, Jane Yu, Song Jiang, Karthik Padthe, Yang Li, Dong Wang, Ilia Kulikov, Kyunghyun Cho, Yuandong Tian, Jason E Weston, and 1 others. 2025. Naturalreasoning: Reasoning in the wild with 2.8 m challenging questions. *arXiv preprint arXiv:2502.13124*.

Xiang Yue, Tuney Zheng, Ge Zhang, and Wenhu Chen. 2024. Mammoth2: Scaling instructions from the web. *arXiv preprint arXiv:2405.03548*.

Eric Zelikman, Yuhuai Wu, Jesse Mu, and Noah Goodman. 2022. Star: Bootstrapping reasoning with reasoning. *Advances in Neural Information Processing Systems*, 35:15476–15488.

Weihao Zeng, Yuzhen Huang, Qian Liu, Wei Liu, Keqing He, Zejun Ma, and Junxian He. 2025. Simplerl-zoo: Investigating and taming zero reinforcement learning for open base models in the wild. *Preprint*, arXiv:2503.18892.

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

Yuxiang Zhang, Yuqi Yang, Jiangming Shu, Yuhang Wang, Jinlin Xiao, and Jitao Sang. 2024b. Openrft: Adapting reasoning foundation model for domain-specific tasks with reinforcement fine-tuning. *arXiv preprint arXiv:2412.16849*.

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.

A Appendix

A.1 Template

Table 6 shows the template for the grading task. Table 7 shows the template for the classification task. Table 8 shows the classification of subjects into STEM (Science, Technology, Engineering, and Mathematics), Social Sciences, Humanities, and Applied Sciences.

A.2 Discussions about RLVR for Mathematical Tasks

Mathematics Data To ensure high-quality reference answers, we use a large-scale dataset of 773k Chinese QA pairs, collected under authorized licenses from educational websites. This dataset covers three educational levels: elementary, middle, and high school. Unlike well-structured yet small-scale benchmarks such as MATH (Hendrycks et al., 2021b) and GSM8K (Cobbe et al., 2021), our reference answers are inherently free-form, often interwoven with rationales or involving several sub-questions yet lacking clear structural patterns. As a result, rule-based reward functions that rely on clean, well-structured answers for verification struggle to process these unstructured reference answers effectively. We use GPT-4o-mini to translate questions and their corresponding responses into English. We randomly sample 3,000 QA pairs from each level and reserve them for testing. The average length of reference answers in the test set is 33.7, 36.3, and 53.9 words for elementary, middle, and high school levels, respectively. These are much longer than those in the GSM8K (1 word) and MATH (1.3 words) test sets. We use the same 7B reward model introduced in Section 3.1. We will release the data.

In our experiments, we find GRPO is not stable when the number of the samples is small (e.g., four). For the results in Table 9, eight responses are sampled from the policy during exploration for all GRPO experiments, compared to four for the other two methods. As a result, the comparison between GRPO and the other algorithms is not entirely fair.

A.3 Agreement

Note that for each instance, we obtain only a single judgment from GPT-4o. As shown in Figure 5, the standard deviation of κ across different values of m is below 0.004, indicating the agreement between the two evaluation methods is highly stable with respect to the number of sampled judgments. For multi-subject tasks, we observe only marginal gains as m increases, suggesting that additional samples may offer diminishing returns in terms of reliability. These results are based on evaluations of 6,000 multi-subject test instances (Section 3.1) and 9,000 math-specific test instances (Section A.2).

To further assess the reliability of LLM-based evaluation, we compare GPT-4o and Qwen2.5-72B-Instruct (with $m = 1$) against human annotations on a randomly selected subset of 100 instances from the multi-subject test set. As shown in Table 4, both LLM-based evaluations exhibit strong agreement with human judgments: GPT-4o achieves a Cohen's κ of 0.882, while Qwen2.5-72B-Instruct reaches 0.857. These results indicate that automated evaluations are well-aligned with human preferences, at least for short reference answers across domains, supporting their use as scalable, high-quality alternatives to manual evaluation.

Evaluation Method	m	κ vs. Human (↑)
GPT-4o	–	0.882
Qwen2.5-72B-Instruct	1	0.857
	2	0.810
	3	0.832
	4	0.857
	5	0.857
	6	0.857
	7	0.832
	8	0.832
	9	0.832
	10	0.832

Table 4: Agreement with human annotations, measured using Cohen's κ.

Hyperparameter	Reward Training		Main Experiments	
	RL	SFT	RL	SFT
micro_train_batch_size	8	4	8	4
train_batch_size	128	128	128	128
micro_rollout_batch_size	16	–	16	–
rollout_batch_size	128	–	128	–
n_samples_per_prompt	4	–	4/8 (GRPO)	–
max_samples	40000	1600000	30000	30000
max_epochs	1	1	1	1
prompt_max_len	1024	–	1024	–
generate_max_len	1024	–	1024	–
max_len	–	4096	–	4096
actor_learning_rate	5e-7	–	5e-7	–
init_kl_coef	0.01	–	0.01	–

Table 5: Training hyper parameters. Other hyper parameters are the default configuration in OpenRLHF.

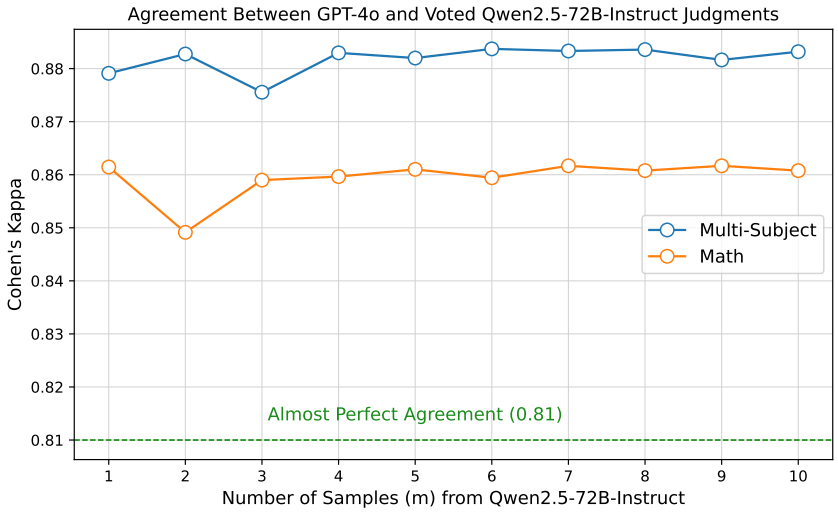

Figure 5: Agreement between GPT-4o and the majority vote of m sampled judgments from Qwen2.5-72B-Instruct, measured using Cohen's Kappa.

A.4 REINFORCE

$$\begin{aligned}\nabla_\theta \mathbb{E}_{y\sim\pi_\theta(\cdot|x)}[r(x,a,y)] &= \sum_y \nabla_\theta \pi_\theta(y \mid x)\cdot r(x,a,y) \\ &= \sum_y \pi_\theta(y \mid x)\nabla_\theta \log \pi_\theta(y \mid x)\cdot r(x,a,y) \\ &= \mathbb{E}_{y\sim\pi_\theta(\cdot|x)}[\nabla_\theta \log \pi_\theta(y \mid x)\cdot r(x,a,y)].\end{aligned} \tag{5}$$

A.5 Hyper-Parameters

Table 5 shows the hyper parameters of our experiments.

A.6 Application to Other Multi-Domain Datasets

In addition to ExamQA, we experiment with two free-form, multi-domain datasets: WebInstruct (Yue et al., 2024), with lengthy reference answers extracted from pre-training corpora written by web users, and NaturalReasoning (Yuan et al., 2025), with similarly long answers generated by LLMs. We present results in Section 4.2 and provide example instances in Table 12 and 13.

A.7 Back-Translation Quality Analysis for ExamQA

To assess potential semantic drift introduced by machine translation of the ExamQA dataset from Chinese to English, we conducted a back-translation analysis on a random sample of 5,000 instances. Specifically, we used GPT-4o-mini to translate the English questions and answers back into Chinese and computed semantic similarity between the original Chinese texts and the back-translated versions using Sentence-BERT (paraphrase-multilingual-MiniLM-L12-v2). The results, broken down by subject category, are shown in Table 14.

A.8 Human Agreement by Subject Category

To provide a granular view of verifier reliability across domains, we conduct human evaluation on 500 randomly selected test instances and computed agreement metrics by subject category. Cohen's Kappa (κ) and accuracy between human judgments and two LLM verifiers (Qwen2.5-72B-Instruct and GPT-4o) are reported in Table 15.

A.9 Soft Reward Distribution Analysis

We analyzed the distribution of soft rewards generated by RM-7B during training on the MATH dataset (first 3,000 samples). As shown in Table 16, while the majority of rewards are near 0 or 1 (reflecting clear correctness/incorrectness), approximately 9.8% of rewards fall in the intermediate range $[0.1, 0.9]$, providing nuanced learning signals for ambiguous cases.

```
Given a problem, determine whether the final answer in the provided (incomplete)
     solution process matches the reference answer.
The reference answer may be one single option character (e.g., A, B, C, D), a
     numerical value, an expression, or a list of answers if multiple questions
     are involved.
**The reference answer may be in Chinese or another language, but your evaluation
      should be language-agnostic.**

Your task:
- Compare the final output of the solution process with the reference answer.
- If they **match exactly**, output **YES**.
- If they **do not match**, output **NO**.
- If the solution process is unclear, incomplete, or ambiguous, assume it is
     incorrect and output **NO**.

Your output must be strictly **'YES'** or **'NO'**, with no additional words,
     punctuation, or explanation.

---

**Question:**
{question}

**Solution Process (Final Step Only):**
{response}

**Reference Answer:**
{reference}

**Output:**
```

Table 6: Template for the grading task.

```
Based on the content of 'Question' and 'Answer' classify the subject into one of
     the following categories.

Return only the corresponding subject ID. If classification is uncertain, return
     999.

**Question:**
{question}

**Answer:**
{answer}

110     Mathematics
120     Information Science and System Science
130     Mechanics
140     Physics
150     Chemistry
170     Earth Science
180     Biology
190     Psychology
210     Agronomy
230     Animal Husbandry and Veterinary Science
310     Basic Medicine
320     Clinical Medicine
330     Preventive Medicine and Public Health
350     Pharmacy
360     Chinese Medicine and Chinese Materia Medica
413     Information and System Science Related Engineering and Technology
416     Natural Science Related Engineering and Technology
420     Surveying and Mapping Science and Technology
430     Materials Science
460     Mechanical Engineering
470     Power and Electrical Engineering
510     Electronics and Communications Technology
520     Computer Science and Technology
530     Chemical Engineering
550     Food Science and Technology
560     Civil Engineering
570     Water Conservancy Engineering
580     Transportation Engineering
610     Environmental/Resource Science and Technology
620     Safety Science and Technology
630     Management
710     Marxism
720     Philosophy
730     Religious Studies
740     Linguistics
750     Literature
760     Art
770     History
790     Economics
810     Political Science
820     Law
840     Sociology
850     Ethnology and Cultural Studies
860     Journalism and Communication
870     Library, Information, and Documentation
880     Education
890     Sports Science
910     Statistics
999     Unclassified
```

Table 7: Template for the classification task, with subject names and IDs referenced from (Yu et al., 2021).

Category	Subject IDs
STEM	110 (Mathematics), 120 (Information Science and System Science), 130 (Mechanics), 140 (Physics), 150 (Chemistry), 170 (Earth Science), 180 (Biology), 430 (Materials Science), 460 (Mechanical Engineering), 470 (Power and Electrical Engineering), 510 (Electronics and Communications Technology), 520 (Computer Science and Technology), 530 (Chemical Engineering), 560 (Civil Engineering), 570 (Water Conservancy Engineering), 580 (Transportation Engineering), 610 (Environmental/Resource Science and Technology), 620 (Safety Science and Technology), 910 (Statistics)
Social Sciences	190 (Psychology), 790 (Economics), 810 (Political Science), 820 (Law), 840 (Sociology), 850 (Ethnology and Cultural Studies), 860 (Journalism and Communication), 870 (Library, Information, and Documentation), 880 (Education), 890 (Sports Science), 630 (Management)
Humanities	710 (Marxism), 720 (Philosophy), 730 (Religious Studies), 740 (Linguistics), 750 (Literature), 760 (Art), 770 (History)
Applied Sciences	210 (Agronomy), 230 (Animal Husbandry and Veterinary Science), 310 (Basic Medicine), 320 (Clinical Medicine), 330 (Preventive Medicine and Public Health), 350 (Pharmacy), 360 (Chinese Medicine and Chinese Materia Medica), 413 (Information and System Science Related Engineering and Technology), 416 (Natural Science Related Engineering and Technology), 420 (Surveying and Mapping Science and Technology), 550 (Food Science and Technology)

Table 8: Classification of subjects into STEM (Science, Technology, Engineering, and Mathematics), Social Sciences, Humanities, and Applied Sciences.

Method	Reward Model	Score Type	MATH			
			Elementary	Middle	High	Avg
Qwen2.5-72B$_{\text{Instruct}}$ (Team, 2024)			44.2	57.7	40.3	47.4
DeepSeek-R1-Qwen-32B (Guo et al., 2025)			27.6	34.8	17.4	26.6
Vanilla (Qwen2.5-7B)			43.1	53.9	33.2	43.4
+SFT			53.6	50.5	32.9	45.7
REINFORCE++	Rule-based	binary	58.5	66.5	46.7	57.2
	Qwen2.5-**72**B$_{\text{Instruct}}$	binary	**64.4**	**72.1**	51.6	**62.7**
		soft	62.5	71.2	**53.1**	62.3
	RM-**7**B	binary	63.8	71.7	51.9	62.5
		soft	62.9	70.7	53.0	62.2
RLOO	Rule-based	binary	58.2	67.0	50.2	58.5
	Qwen2.5-**72**B$_{\text{Instruct}}$	binary	63.0	70.8	51.1	61.6
		soft	**63.8**	71.0	52.4	62.4
	RM-**7**B (ours)	binary	63.4	**71.8**	**53.8**	**63.0**
		soft	63.3	71.7	53.6	62.9
GRPO	Rule-based	binary	60.6	67.4	48.7	58.9
	Qwen2.5-**72**B$_{\text{Instruct}}$	binary	64.4	72.5	54.8	63.9
		soft	65.0	72.2	52.8	63.3
	RM-**7**B (ours)	binary	**65.7**	**72.8**	**56.0**	**64.8**
		soft	**65.7**	72.2	54.2	64.0

Table 9: Performance comparison of different methods on math tasks.

Method	Reward	Score Type	Math				Multi-Subject					
			E	M	H	Avg	STEM	Social	Humanities	Applied	Others	Avg
Qwen2.5-72B-Instruct	–	–	44.2	57.7	40.3	47.4	25.2	20.1	28.7	20.5	21.0	22.6
DeepSeek-R1-Distill-Qwen-32B	–	–	27.6	34.8	17.4	26.6	23.2	21.8	26.7	20.5	18.5	21.7
Base	–	–	43.1	53.9	33.2	43.4	16.3	14.9	15.2	13.3	14.8	15.0
SFT	–	–	53.6	50.5	32.9	45.7	24.6	22.8	25.7	20.9	22.6	23.1
DeepSeek-Math-7B	rule based	binary	39.1	54.4	54.2	49.3	10.8	8.8	12.2	11.6	6.3	9.8
	Qwen2.5-72B$_{\text{Instruct}}$	binary	39.7	54.9	54.4	49.7	7.1	6.2	5.6	3.0	5.7	5.7
		soft	39.7	54.9	54.2	49.6	11.2	10.1	14.2	11.9	7.0	10.5
	RM-7B (ours)	binary	27.1	25.3	16.8	23.1	11.2	7.9	6.3	5.2	8.8	8.5
		soft	34.6	42.5	44.7	40.6	11.5	10.0	13.2	12.0	6.9	10.6
Qwen2.5-7B-Instruct	rule based	binary	57.2	67.2	49.4	57.9	22.5	18.2	18.2	18.9	16.8	19.5
	Qwen2.5-72B$_{\text{Instruct}}$	binary	63.6	72.1	54.0	63.3	26.0	22.6	24.8	20.1	22.6	23.3
		soft	63.6	71.1	54.3	63.0	28.0	28.5	29.4	25.4	22.6	26.8
	RM-7B (ours)	binary	63.6	71.9	54.6	63.4	24.4	22.3	24.4	21.3	21.0	22.7
		soft	63.8	71.6	53.0	62.8	24.9	23.5	26.4	21.7	22.0	23.5
Qwen2.5-7B	rule based	binary	58.2	67.0	50.2	58.5	28.2	27.9	27.4	22.4	24.5	26.3
	Qwen2.5-72B$_{\text{Instruct}}$	binary	63.0	70.8	51.1	61.6	29.4	30.5	33.7	24.6	26.1	28.4
		soft	**63.8**	71.0	52.4	62.4	**32.9**	31.4	34.7	**27.7**	26.8	**30.6**
	RM-7B (ours)	binary	63.4	**71.8**	**53.8**	**63.0**	29.3	29.0	33.3	25.8	25.6	28.1
		soft	63.3	71.7	53.6	62.9	31.0	**32.0**	**35.6**	27.0	**27.0**	30.0

Table 10: Performance comparison of different base models under RLOO. E: elementary. M: middle. H: high.

coarse	fine	question	answer
Social Sciences	Psychology	Setting up an activity for students to 'bomb' each other with compliments belongs to ( ).	Self-awareness guidance
STEM	Civil Engineering	A gravity retaining wall meets the Rankine earth pressure conditions, $H = 3\,\mathrm{m}$, top width $2\,\mathrm{m}$, bottom width $3\,\mathrm{m}$, fill $c = 0$, $\phi = 30^\circ$, $\gamma = 18.0\,\mathrm{kN/m^3}$, the base friction coefficient is 0.4, the anti-sliding stability safety factor K_s and the anti-tilting stability safety factor K_t are respectively ()	2.67; 1.73
Humanities	Philosophy	Laozi pointed out in the 'Tao Te Ching', 'Without leaving the door, one knows the world; without peering through the window, one knows the way of heaven. The farther one goes, the less one knows. Therefore, the sage knows without traveling, sees without looking, and achieves without doing.' Laozi's view here	denies the decisive role of practice in understanding
Applied Sciences	Agronomy	Under light, the physiological processes that can occur in the mesophyll cells and vascular bundle sheath cells of wheat (C3) are	Production of ATP and [H]

Table 11: Example question and reference answer pairs in ExamQA.

question	answer
I'm uncertain if my calculation is correct. Given the probability distribution function $P(X \geq c) = (1-c)^5$, I originally had $c = 0.01$ for the left-hand side (LHS). However, I'm now thinking that the LHS should be $1 - 0.01 = 0.99$ due to the shape of the curve. Could someone please clarify which is the proper LHS for this expression? Thank you.	Your revised understanding is correct. The left-hand side (LHS) should indeed be $1 - c$, not just c. Thus, for $c = 0.01$, the LHS would be 0.99. To verify the correctness of the expression $P(X \geq c) = (1 - c)^5$, consider the following checks: 1. When $c = 0$, the probability $P(X \geq 0)$ should be 1, which is true because $(1-0)^5 = 1$. 2. When $c = 1$, the probability $P(X \geq 1)$ should be 0, which is also true because $(1 - 1)^5 = 0$. These checks confirm that the expression $P(X \geq c) = (1 - c)^5$ is correct.
Is 3y = 9x a direct variation? If so, what is the constant of variation?	A direct variation has the form $y = kx$, where k is the constant of variation. Rearranging $3y = 9x$ gives $y = \frac{9}{3}x = 3x$, so it matches $y = kx$ with $k = 3$. Therefore, $3y = 9x$ is a direct variation, and the constant of variation is 3.
How should I connect the backplane for an MM5452 with a 4-digit parallel 7-segment LCD display (specifically a Varitronix VI-422), considering the MM5452 datasheet mentions a BP pin connection, but the display doesn't have a BP pin label?	The MM5452 datasheet refers to the backplane (BP) connection, which is essential for the proper functioning of the display. In the case of the Varitronix VI-422 display, the backplane connection corresponds to the COM pins, which are labeled Pin 1 and Pin 40. These common pins serve as the backplane, as they provide the reference for the alternating current (AC) signal applied to the segments to make them visible. Connect the MM5452's backplane pins to the display's COM pins to establish the necessary connection.
How can we prove that any linearly independent set of vectors in a finite-dimensional vector space cannot have more vectors than a basis of the vector space?	To prove this, consider an arbitrary linearly independent set of vectors with $n + 1$ elements, where n is the number of vectors in a basis of the vector space. Express each vector in this set as a linear combination of the basis vectors. Substituting these expressions into the equation defining linear independence, we obtain a homogeneous system of linear equations with more unknowns (the coefficients of the linear combinations) than equations. By the properties of homogeneous systems, this system must have a nontrivial solution, implying that the vectors in the linearly independent set are linearly dependent, contradicting the assumption of linear independence. Therefore, any linearly independent set cannot have more vectors than a basis.

Table 12: Example question and reference answer pairs in WebInstruct.

question	answer
What is the maximum possible order for an element in S_5, and how can it be determined by analyzing the disjoint cycle structures and their least common multiples?	The largest order would be the least common multiple of 3 and 2, which is 6.
Explain how the energy from sunlight contributes to the organization of chemical molecules into primordial biomass, and discuss the role of entropy in the process of evolution. Provide a detailed analysis of the relationship between the Second Law of Thermodynamics and the emergence of complex life forms on Earth.	The energy from sunlight provides the energy to enable the organization of chemical molecules into primordial biomass, but it does not directly cause the organization.
Find the second derivative of $y = 2x^3e^{4x}$	$y'' = 6x^2e^{4x} + 8x^3e^{4x} + 24x^3e^{4x} + 32x^4e^{4x}$
Prove algebraically the combinatorial identity $\binom{3n}{n} = \sum_{r=0}^{n} \binom{n}{r}\binom{2n}{n-r}$ using the Binomial Theorem. Show all steps clearly, starting from the expansion of $(1+x)^{3n}$ and explain how the coefficients of x^n on both sides of the equation lead to the desired identity.	$\boxed{\binom{3n}{n} = \sum_{r=0}^{n} \binom{n}{r}\binom{2n}{n-r}}$

Table 13: Example question and reference answer pairs in NaturalReasoning.

Category	Sample Size	Avg. Cosine Similarity		Avg. MSE	
		Question	Answer	Question	Answer
Applied Sciences	1063	0.9389	0.9486	0.0038	0.0036
Humanities	248	0.9518	0.9564	0.0031	0.0030
STEM	1580	0.9615	0.9645	0.0027	0.0027
Social Sciences	1315	0.9631	0.9536	0.0027	0.0033
Unclassified	794	0.9572	0.9530	0.0030	0.0032
Overall	5000	0.9560	0.9560	0.0030	0.0031

Table 14: Semantic similarity between original Chinese and back-translated Chinese texts for ExamQA. High cosine similarity (>0.95) and low MSE (0.003) indicate strong semantic preservation.

Category	Sample Size	Cohen's κ		Accuracy	
		Qwen	GPT-4o	Qwen	GPT-4o
Applied Sciences	98	0.726	0.767	0.891	0.908
Humanities	21	0.978	1.000	0.990	1.000
STEM	163	0.868	0.860	0.942	0.939
Social Sciences	129	0.850	0.800	0.937	0.915
Unclassified	89	0.753	0.861	0.918	0.955
Overall	500	0.827	0.834	0.929	0.932

Table 15: Agreement between LLM verifiers and human annotators by subject category. Results confirm strong cross-domain alignment (Qwen refers to Qwen2.5-72B-Instruct).

Score Range	Count	Percentage (%)
[0.0 – 0.1)	2215	73.83
[0.1 – 0.2)	58	1.93
[0.2 – 0.3)	39	1.30
[0.3 – 0.4)	22	0.73
[0.4 – 0.5)	24	0.80
[0.5 – 0.6)	21	0.70
[0.6 – 0.7)	25	0.83
[0.7 – 0.8)	31	1.03
[0.8 – 0.9)	46	1.53
[0.9 – 1.0]	519	17.30
Total	3000	100.00

Table 16: Distribution of soft rewards on MATH training data. Intermediate rewards (9.8% of total) enable finer-grained policy updates.