Inference-Time Scaling for Generalist Reward Modeling

Zijun Liu[1,2†*], **Peiyi Wang**[1*], **Runxin Xu**[1], **Shirong Ma**[1], **Chong Ruan**[1],
Peng Li[3], **Yang Liu**[2,3], **Yu Wu**[1]
[1]DeepSeek-AI, [2]Dept. of Computer Sci. & Tech., Tsinghua University,
[3]Institute for AI Industry Research (AIR), Tsinghua University
zj-liu24@mails.tsinghua.edu.cn, wangpeiyi9979@gmail.com

Abstract

Reinforcement learning (RL) has been widely adopted in post-training for large language models (LLMs) at scale. Recently, the incentivization of reasoning capabilities in LLMs from RL indicates that *proper learning methods could enable effective inference-time scalability*. A key challenge of RL is to obtain accurate reward signals for LLMs in various domains beyond verifiable questions or artificial rules. In this work, we investigate how to improve reward modeling (RM) with more inference compute for general queries, i.e. the **inference-time scalability of generalist RM**, and further, how to improve the effectiveness of performance-compute scaling with proper learning methods. For the RM approach, we adopt pointwise generative reward modeling (GRM) to enable flexibility for different input types and potential for inference-time scaling. For the learning method, we propose **Self-Principled Critique Tuning** (SPCT) to foster scalable reward generation behaviors in GRMs through online RL, to generate principles adaptively and critiques accurately, resulting in **DeepSeek-GRM** models. Furthermore, for effective inference-time scaling, we use parallel sampling to expand compute usage, and introduce a meta RM to guide voting process for better scaling performance. Empirically, we show that SPCT significantly improves the quality and scalability of GRMs, outperforming existing methods and models in various RM benchmarks without severe biases, and could achieve better performance compared to training-time scaling. DeepSeek-GRM still meets challenges in some tasks, which we believe can be addressed by future efforts in generalist reward systems. The models will be released and open-sourced.

1 Introduction

The remarkable advancements in large language models (LLMs) (DeepSeek-AI, 2024b; OpenAI, 2025b) have catalyzed significant shifts in artificial intelligence research, enabling models to perform tasks

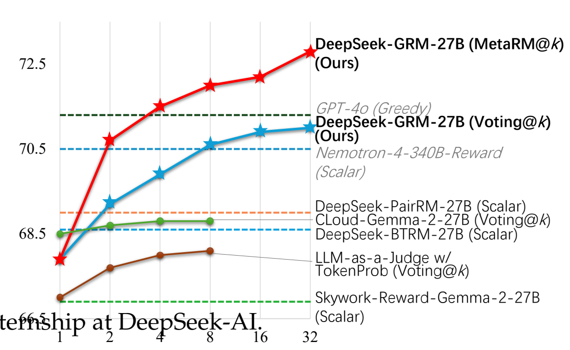


图 1: Inference-time scaling performance with different RMs on all tested RM benchmarks. Results are shown with up to 8 samples for each

*Equal contribution. †Work done during internship at DeepSeek-AI.

Preprint. Under review.

Inference-Time Scaling for Generalist Reward Modeling

Zijun Liu[1,2†*], **Peiyi Wang**[1*], **Runxin Xu**[1], **Shirong Ma**[1], **Chong Ruan**[1],
Peng Li[3], **Yang Liu**[2,3], **Yu Wu**[1]
[1]DeepSeek-AI, [2]Dept. of Computer Sci. & Tech., Tsinghua University,
[3]Institute for AI Industry Research (AIR), Tsinghua University
zj-liu24@mails.tsinghua.edu.cn, wangpeiyi9979@gmail.com

Abstract

强化学习（RL）已被广泛应用于大规模语言模型（LLMs）的后训练中。最近，通过RL激励LLMs推理能力的研究表明，*适当的学习方法可以实现有效的推理时间可扩展性*。RL的一个关键挑战是在各种领域中为LLMs获取准确的奖励信号，而这些领域超出了可验证的问题或人工规则的范畴。在本工作中，我们研究了如何通过更多的推理计算来改进针对通用查询的奖励建模（RM），即**通用RM的推理时间可扩展性**，以及如何通过适当的学习方法进一步提升性能-计算扩展的有效性。对于RM方法，我们采用逐点生成式奖励建模（GRM），以支持不同输入类型的灵活性和推理时间扩展的潜力。对于学习方法，我们提出了**自原则化批评调优**（SPCT），通过在线RL促进GRMs中的可扩展奖励生成行为，从而自适应地生成原则并精确地进行批评，最终得到**DeepSeek-GRM**模型。此外，为了实现有效的推理时间扩展，我们使用并行采样来扩展计算资源的使用，并引入一个元RM来指导投票过程，以获得更好的扩展性能。从经验上来看，我们证明SPCT显著提升了GRMs的质量和可扩展性，在各种RM基准测试中超越了现有的方法和模型，且没有严重的偏差，同时相比训练时间扩展能够取得更好的性能。DeepSeek-GRM在某些任务中仍然面临挑战，但我们相信未来在通用奖励系统方面的努力可以解决这些问题。这些模型将会被发布并开源。

1 Introduction

大规模语言模型（LLMs）的显著进展(DeepSeek-AI, 2024b; OpenAI, 2025b) 推动了人工智能研究的重大转变，使模型能够执行需要理解、生成和细致决策能力的任务。最近，强化学习（RL）作为一种针对LLMs的后训练方法已被广泛采用，并在人类价值观对齐方面取得了显著改进

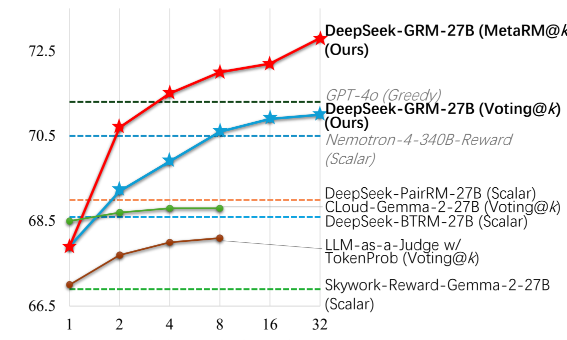


图 1：推理时在所有测试的RM基准上使用不同的RM进行缩放性能。结果显示每种方法最多8个样本，我们的方法进一步缩放到32个样本。非斜体字体表示基于Gemma-2-27B的模型。

*同等贡献。†在DeepSeek-AI实习期间完成的工作。

that require understanding, generation, and nuanced decision-making capabilities. Recently, reinforcement learning (RL) as a post-training method for LLMs has been widely adopted at scale, and results in remarkable improvements in human value alignment (Ouyang et al., 2022; Bai et al., 2022a), long-term reasoning (DeepSeek-AI, 2025; OpenAI, 2025c), and environment adaptation (OpenAI, 2025a) for LLMs. Reward modeling (Gao et al., 2023), as a crucial component in RL, is essential for generating accurate reward signals for LLM responses. Current studies (Lightman et al., 2024; DeepSeek-AI, 2025) also show that, with high-quality and robust rewards in either training or inference time, LLMs can achieve strong performance in specific domains.

However, such high-quality rewards in specific domains are mainly obtained from human-designed environments with clear conditions (Yao et al., 2022; Xie et al., 2024) or from hand-crafted rules for verifiable questions, e.g., part of mathematical problems (Hendrycks et al., 2021; Veeraboina, 2023) and coding tasks (Jimenez et al., 2024; Zhuo et al., 2025). In general domains, reward generation is more challenging, as the criteria for rewards are more diverse and complex, and there are often no explicit reference or ground truth. Generalist reward modeling is thus crucial for improving the performance of LLMs in broader applications, either from post-training perspectives, e.g., RL at scale, or from inference perspectives, e.g., RM-guided search. Furthermore, RM performance should be improved by increasing both the training compute (Gao et al., 2023) and the inference compute.

In practice, challenges arise in making RMs both general and effectively scalable in inference time. Generalist RM demands (1) flexibility for different input types and (2) accurate reward generation in various domains. Moreover, effective inference-time scalability requires the RM (3) to generate higher-quality reward signals with increased inference compute, and (4) to learn scalable behaviors for better performance-compute scaling. Existing research on reward modeling demonstrates several paradigms for reward generation, including scalar (Cobbe et al., 2021; Wang et al., 2024d; Liu et al., 2024), semi-scalar (Ye et al., 2024; Yu et al., 2025b; Zhang et al., 2025a), and generative (Li et al., 2024a; Kim et al., 2024; Vu et al., 2024; Cao et al., 2024; Arabzadeh et al., 2024; Ye et al., 2025; Alexandru et al., 2025; Yu et al., 2025a) approaches, and various scoring patterns, such as pointwise (Kendall & Smith, 1940; Gao et al., 2023; Yuan et al., 2024; Winata et al., 2025; Guo et al., 2025) and pairwise (Park et al., 2024; Zheng et al., 2023; Jiang et al., 2023; Wang et al., 2024c; Liu et al., 2025). These approaches inherently determine the input flexibility and the inference-time scalability of RMs (*(1)&(3)*), as shown in Figure 2. For instance, pairwise RMs only consider the relative preference of paired responses, lacking flexibility to accept single or multiple responses as input; scalar RMs could hardly generate diverse reward signals for the same response, which obstructs getting better rewards through sampling-based inference-time scaling methods (Snell et al., 2025). Also, different learning methods (Wang et al., 2024a; Ankner et al., 2024; Wang et al., 2024c; Mahan et al., 2024) have been proposed to improve the quality of rewards, but

(Ouyang et al., 2022; Bai et al., 2022a)，长期推理能力 (DeepSeek-AI, 2025; OpenAI, 2025c) 以及LLMs的环境适应性 (OpenAI, 2025a)。奖励建模 (Gao et al., 2023) 作为RL中的关键组成部分，对于生成准确的LLM响应奖励信号至关重要。当前的研究 (Lightman et al., 2024; DeepSeek-AI, 2025) 表明，在训练或推理过程中使用高质量且稳健的奖励，LLMs可以在特定领域中实现强大的性能。

然而，在特定领域中，这种高质量的奖励主要来自于具有明确条件的人工设计环境 (Yao et al., 2022; Xie et al., 2024) 或者为可验证问题手工制定的规则，例如部分数学问题 (Hendrycks et al., 2021; Veeraboina, 2023) 和编程任务 (Jimenez et al., 2024; Zhuo et al., 2025)。而在通用领域中，奖励生成更具挑战性，因为奖励的标准更加多样化和复杂，并且通常没有明确的参考或真实标签。因此，通用奖励建模对于从后训练角度（如大规模RL）或推理角度（如RM引导的搜索）提升LLMs在更广泛应用中的性能至关重要。此外，应通过增加训练计算量和推理计算量来进一步提升RM的性能 (Gao et al., 2023)。在实际应用中，使奖励模型（RMs）既通用又能在推理时间上有效扩展存在挑战。通用型RM需要具备 (1) 适应不同输入类型的能力和 (2) 在各种领域生成准确奖励的能力。此外，有效的推理时间可扩展性要求RM (3) 随着推理计算量的增加生成更高质量的奖励信号，并且 (4) 学习可扩展的行为以实现更好的性能-计算扩展。现有的奖励建模研究展示了几种奖励生成范式，包括标量方法 (Cobbe et al., 2021; Wang et al., 2024d; Liu et al., 2024), 半标量方法 (Ye et al., 2024; Yu et al., 2025b; Zhang et al., 2025a) 和生成方法 (Li et al., 2024a; Kim et al., 2024; Vu et al., 2024; Cao et al., 2024; Arabzadeh et al., 2024; Ye et al., 2025; Alexandru et al., 2025; Yu et al., 2025a)，以及各种评分模式，例如逐点评分 (Kendall & Smith, 1940; Gao et al., 2023; Yuan et al., 2024; Winata et al., 2025; Guo et al., 2025) 和成对评分 (Park et al., 2024; Zheng et al., 2023; Jiang et al., 2023; Wang et al., 2024c; Liu et al., 2025)。这些方法本质上决定了RM的输入灵活性和推理时间上的可扩展性 (*(1)&(3)*)，如图 2 所示。例如，成对RM仅考虑配对响应的相对偏好，缺乏接受单个或多个响应作为输入的灵活性；标量RM很难为相同的响应生成多样化的奖励信号，这阻碍了通过基于采样的推理时间扩展方法获得更好的奖励 (Snell et al., 2025)。此外，已经提出了不同的学习方法 (Wang et al., 2024a; Ankner et al., 2024; Wang et al., 2024c; Mahan et al., 2024) 来提高奖励的质量，但很少有方法关注推理时间上的可扩展性并研究所学奖励生成行为与RM推理时间扩展的有效性之间的联系，导致性能改进有限 (*(2)&(4)*). 当前的研究 (DeepSeek-AI, 2025) 表明，适当的训练方法可以实现有效的推理时间扩展，这引发了一个问题：*我们能否设计一种旨在为通用奖励建模启用有效推理时间扩展的学习方法？*在本工作中，我们研究了不同的RM方法，发现点对点生成式奖励建模（GRM）可以在纯语言表示中统一单个、成对和多个响应的评分，克服挑战*(1)*。我们探索了某些原则可以指导GRMs在适当标准下的奖励生成，从而提高奖励的质量，这启发我们*通过扩展高质量原则的生成和准确批评,可能实现RM的推理时间可扩展性*。基于这一初步研究，我们提出了一种新的学习方法，即**自原则批评调优**（SPCT），以促进GRMs在推理时间上的有效可扩展行为。通过利用基于规则的在线强化学习，SPCT使GRMs能够根据输入查询和响应自适应地提出原则和批评，从而在一般领域内获得更好的结果奖励（挑战*(2)*）。随后，我们提出了**DeepSeek-GRM-27B**，它基于Gemma-2-27B (Team, 2024)通过SPCT进行后训练。对于推理时间扩展，我们通过多次采样来增加计算使用量。通过并行采样，DeepSeek-GRM可以生成不同的原则集及其相应的批评，并最终投票决定最终奖励。**通过更大规模的**

few of them focus on inference-time scalability and study the interconnection between the learned reward generation behaviors and the effectiveness of inference-time scaling of RMs, resulting in marginal performance improvement (*(2)&(4)*). Current research (DeepSeek-AI, 2025) indicates that effective inference-time scalability could be enabled by proper learning methods, which raises the question: *Can we design a learning method aiming to enable effective inference-time scaling for generalist reward modeling?*

In this work, we investigate in different approaches for RM, and found that pointwise generative reward modeling (GRM) could unify the scoring of single, paired, and multiple responses within pure language representation, overcoming challenge *(1)*. We explored that certain principles could guide reward generation within proper criteria for GRMs, improving the quality of rewards, which inspired us that *inference-time scalability of RM might be achieved by scaling the generation of high-quality principles and accurate critiques*. Based on this preliminary, we propose a novel learning method, **Self-Principled Critique Tuning** (SPCT), to foster effective inference-time scalable behaviors in GRMs. By leveraging rule-based online RL, SPCT enables GRMs to learn to adaptively posit principles and critiques based on the input query and responses, leading to better outcome rewards in general domains (challenge *(2)*). We then come up with **DeepSeek-GRM-27B**, which is post-trained with SPCT based on Gemma-2-27B (Team, 2024). For inference-time scaling, we expand compute usage by sampling multiple times. By sampling in parallel, DeepSeek-GRM could generate different sets of principles and according critiques, and then vote for the final reward. **With larger-scale sampling, DeepSeek-GRM could judge more accurately upon principles with higher diversity, and output rewards with finer granularity**, which resolves challenge *(3)&(4)*. Furthermore, We train a meta RM besides voting for better scaling performance. Empirically, we show that SPCT significantly improves the quality and scalability of GRMs, outperforming existing methods and models in multiple comprehensive RM benchmarks without severe domain biases. We also compared the inference-time scaling performance of DeepSeek-GRM-27B with larger models up to 671B parameters, and found it could achieve better performance compared to training-time scaling on model sizes. Though the current method meets challenges in efficiency and specific tasks, with efforts beyond SPCT, we believe GRMs with enhanced scalability and efficiency could serve as a versatile interface for generalist reward systems, advancing the frontiers of LLM post-training and inference.

In general, our main contribution is as follows.

1. We propose a novel approach, **Self-Principled Critique Tuning** (SPCT), to foster effective inference-time scalability for generalist reward modeling, resulting in **DeepSeek-GRM** models. And we further introduce a meta RM to effectively improve the inference-time scaling performance of DeepSeek-GRM beyond voting.

2. We empirically show SPCT significantly improves the quality and inference-time scalability of GRMs over existing methods and several strong public models.

3. We also applied the SPCT training schedule on LLMs with larger sizes and found that inference-time scaling could outperform model size scaling in training time.

Preprint. Under review.

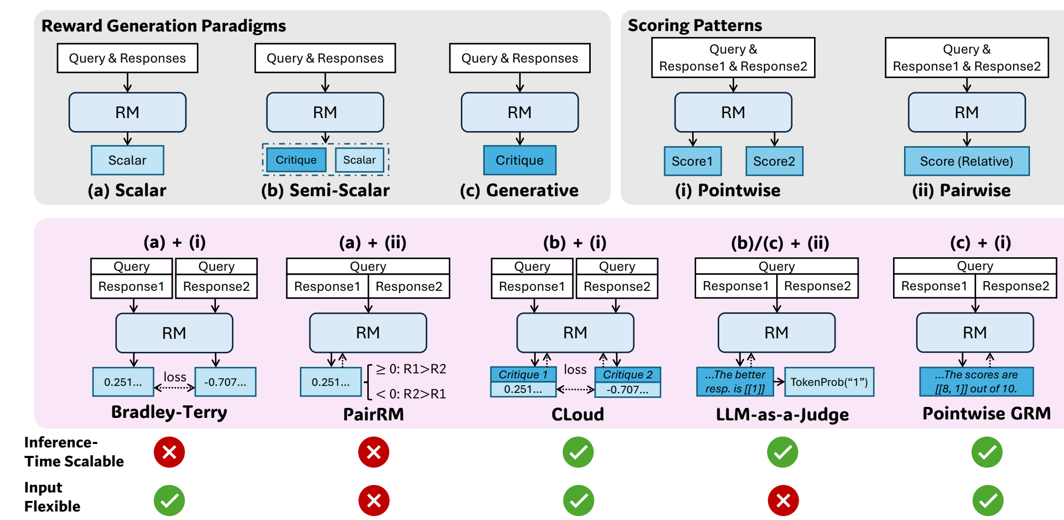


图 2: 不同的奖励生成范式，包括 (a) 标量、(b) 半标量和 (c) 生成式方法，以及不同的评分模式，包括 (i) 点估计和 (ii) 成对比较方法。我们列出了每种方法的代表性技术，并对应说明了其推理时间的可扩展性（是否可以通过多次采样获得更好的奖励）和输入灵活性（是否支持对单个或多个响应进行评分）。

采样,DeepSeek-GRM可以根据更高多样性的原则做出更准确的判断,并输出更精细粒度的奖励，从而解决了挑战*(3)&(4)*。此外，我们还训练了一个元RM以辅助投票，以实现更好的扩展性能。实证研究表明，SPCT显著提高了GRMs的质量和可扩展性，在多个全面的RM基准测试中优于现有方法和模型，且没有严重的领域偏差。我们还将DeepSeek-GRM-27B与多达671B参数的大模型进行了推理时间扩展性能比较，发现其相对于训练时间扩展在模型大小上可以取得更好的性能。尽管当前方法在效率和特定任务上仍面临挑战，但我们相信，通过超出SPCT的努力，具有增强可扩展性和效率的GRMs可以作为通用奖励系统的多功能接口，推动LLM后训练和推理的前沿。

总的来说，我们的主要贡献如下：

1. 我们提出了一种新的方法，**自原则批评调整**（SPCT），以促进通用奖励建模在推理时间的有效可扩展性，从而得到**DeepSeek-GRM**模型。我们进一步引入了一个元奖励模型（meta RM），以有效提升DeepSeek-GRM在推理时间的扩展性能，超越简单的投票机制。
2. 我们通过实验证明，与现有方法和几个强大的公开模型相比，SPCT 显著提高了 GRMs 的质量和推理时间的可扩展性。
3. 我们还将SPCT训练计划应用于更大规模的LLM上，发现推理时间扩展在训练时间内可以超越模型规模扩展的表现。

2 Preliminaries

2.1 Comparisons of Different RM approaches

如图 2所示，RM 方法主要由奖励生成范式和评分模式决定，这从根本上影响了 RM 的推理时间可扩展性和输入灵活性。对于**奖励生成范式**，我们区分了三种主要方法：标量、半标量和生成式。标量方法为给定的查询和响应分配标量值，而半标量方法生成文本判断（称为“批评”）以及标量奖励值。生成式方法仅生成作为文本奖励的批评，从中可以提取奖励值。对于**评分模式**，我们区分了两种主要方法：逐点和成对。逐点方法为每个响应分配单独的分数，而成对方法从所有候选中选择一个最佳响应。

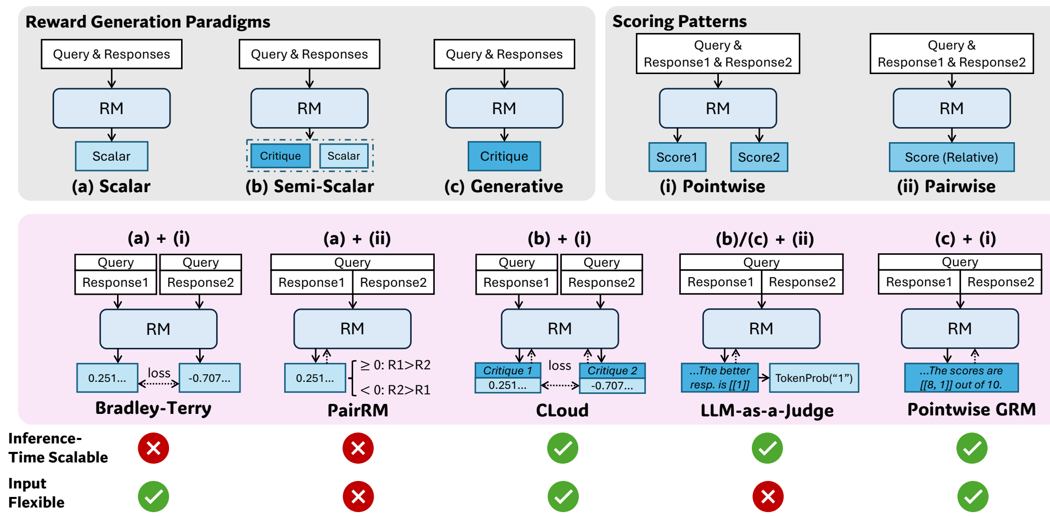


图 2: Different paradigms for reward generation, including (a) scalar, (b) semi-scalar, and (c) generative approaches, and different scoring patterns, including (i) pointwise and (ii) pairwise approaches. We list the representative methods for each approach, and corresponding inference-time scalability (whether better rewards could be obtained from multiple sampling) and input flexibility (whether supports rating single and multiple responses).

2 Preliminaries

2.1 Comparisons of Different RM approaches

As shown in Figure 2, RM approaches are mainly determined by reward generation paradigms and scoring patterns, which inherently affect the inference-time scalability and the input flexibility of the RM. For **reward generation paradigms**, we distinguish three main approaches: scalar, semi-scalar, and generative. The scalar approach assigns scalar values to the given query and responses, while the semi-scalar approach generates textual judgement, termed "critique", and the scalar reward value as well. The generative approach only generates critiques as the textual reward, from which the reward value could be extracted. For **scoring patterns**, we distinguish two main approaches: pointwise and pairwise. The pointwise approach assigns an individual score to each response, while the pairwise approach selects a single best response from all candidates.

To expand compute usage in inference time, we focus on sampling-based methods, which generate multiple sets of rewards for the same query and responses, and then aggregate the final reward. Thus, the *inference-time scalability* of RMs is determined by whether different rewards could be obtained from multiple sampling, where scalar RMs would fail in most cases due to the invariant generation of rewards; and the *input flexibility* is defined by whether the RM supports rating single, paired, and multiple responses, where pairwise RMs could hardly rate single responses and usually require extra techniques (Jiang et al., 2023; Liu et al., 2025) to handle multiple responses. The formulation of pointwise GRMs is:

$$\{S_i\}_{i=1}^n = f_{\text{point}}(\mathcal{R}, \{y_i\}_{i=1}^n) = f_{\text{extract}}(\boldsymbol{C}), \quad \mathcal{R} = \boldsymbol{C} \sim r_\theta\left(x, \{y_i\}_{i=1}^n\right), S_i \in \mathbb{R}, \tag{1}$$

where x is the query, y_i is the i-th response, r_θ is the reward function parameterized by θ, $\mathcal{R}$ is the reward, $\boldsymbol{C}$ is the critique, S_i is the individual score of y_i, and $f_{\text{extract}}(\cdot)$ extracts the

为了在推理时扩展计算使用，我们专注于基于采样的方法，这些方法为相同的查询和响应生成多组奖励，然后聚合最终奖励。因此，RM 的*推理时间可扩展性*取决于是否可以通过多次采样获得不同的奖励，在大多数情况下，标量 RM 会由于奖励生成的不变性而失败；而*输入灵活性*则由 RM 是否支持对单个、成对和多个响应进行评分来定义，其中成对 RM 很难对单个响应进行评分，并且通常需要额外的技术 (Jiang et al., 2023; Liu et al., 2025) 来处理多个响应。逐点 GRMs 的公式为：

$$\{S_i\}_{i=1}^n = f_{\text{point}}(\mathcal{R}, \{y_i\}_{i=1}^n) = f_{\text{extract}}(\boldsymbol{C}), \quad \mathcal{R} = \boldsymbol{C} \sim r_\theta\left(x, \{y_i\}_{i=1}^n\right), S_i \in \mathbb{R}, \tag{1}$$

其中 x 是查询，y_i 是第 i 个响应，r_θ 是由 θ 参数化的奖励函数，$\mathcal{R}$ 是奖励，$\boldsymbol{C}$ 是批评值，S_i 是 y_i 的个体得分，$f_{\text{extract}}(\cdot)$ 从生成结果中提取奖励。通常，奖励是离散的，在这项工作中，我们默认设定 $S_i \in \mathbb{N}, 1 \leq S_i \leq 10$。详细分析见附录 C.1。

2.2 Boosting Reward Quality with Principles

通用奖励模型（RM）需要在特定领域之外生成高质量的奖励 (Hendrycks et al., 2021; Jimenez et al., 2024)，其中奖励的标准更加多样化和复杂，并且通常没有明确的参考或 ground truth。为此，对于通用领域，我们采用原则来指导奖励生成，而不是人工规则。大型语言模型（LLM）的原则最初是在宪法人工智能（Constitutional AI）中引入的 (Bai et al., 2022b; Sharma et al., 2025)，这些是手工制定的标准，用于引导 LLM 或策划分类器构建安全的数据管道。通过这些原则，通用奖励模型（GRMs）的奖励生成转变为

$$\mathcal{R} = \boldsymbol{C} \sim r_\theta\left(x, \{y_i\}_{i=1}^n, \{p_i\}_{i=1}^m\right), \tag{2}$$

其中$\{p_i\}_{i=1}^m$表示原则。我们进行了一项初步实验，以检查适当原则对奖励质量的影响，使用了Reward Bench的Chat Hard子集 (Lambert et al., 2024) 和 PPE基准的IFEval子集 (Frick et al., 2025)。

我们使用GPT-4o-2024-08-06生成原则，然后对每个样本进行四次逐点奖励计算。接着，我们筛选出与真实值一致的奖励对应的原则。我们测试了不同的大语言模型（LLMs），分别使用它们自动生成的原则和经过筛选的原则，并将这些结果与没有原则指导的默认设置进行了比较。结果如表1所示。我们发现，自动生成的原则几乎不能提升性能，而经过筛选的原则可以显著提高奖励质量。这表明，在正确召集的标准下，适当的原则能够更好地引导奖励生成。详细信息见附录 D。

Method	Chat Hard	IFEval
GPT-4o-2024-08-06	76.1	56.0
w/ Self-Gen. Principles	75.9	55.6
w/ Filtered Principles	77.8	57.5
Gemma-2-27B-it	59.1	56.1
w/ Self-Gen. Principles	64.0	55.8
w/ Filtered Principles	68.0	57.3

表 1: 关于原则对奖励质量影响的初步实验。DeepSeek-GRM-27B的默认设置包括自生成原则。

3 Self-Principled Critique Tuning (SPCT)

受初步结果的启发，我们为逐点GRMs开发了一种新颖的方法，用于学习生成自适应且高质量的原则，这些原则能够有效指导批评的生成，我们将其称为**自原则批评调整**（SPCT）。如图 3所示，SPCT包含两个阶段：拒绝性微调作为冷启动，以及基于规则的在线强化学习，

rewards from generation results. Usually, the rewards are discrete, and in this work, we assign $S_i \in \mathbb{N}, 1 \leq S_i \leq 10$ by default. Detailed analysis is provided in Appendix C.1.

2.2 Boosting Reward Quality with Principles

Generalist RM requires to generate high-quality rewards beyond specific domains (Hendrycks et al., 2021; Jimenez et al., 2024), where the criteria for rewards are more diverse and complex, and there are often no explicit reference or ground truth. To this end, for general domains, we adopt principles to guide reward generation in place of artificial rules. Principles for LLMs are first introduced in Constitutional AI (Bai et al., 2022b; Sharma et al., 2025), which are handicraft criteria that guide the LLMs or curated classifiers to construct safe data pipelines. With principles, the reward generation of GRMs changes to

$$\mathcal{R} = \boldsymbol{C} \sim r_\theta\left(x, \{y_i\}_{i=1}^n, \{p_i\}_{i=1}^m\right), \tag{2}$$

where $\{p_i\}_{i=1}^m$ denotes the principles. We conduct a preliminary experiment to examine the influence of proper principles on reward quality, with the Chat Hard subset of Reward Bench (Lambert et al., 2024) and the IFEval subset of the PPE benchmark (Frick et al., 2025).

We used GPT-4o-2024-08-06 to generate the principles and then pointwise rewards four times for each sample. And we filtered the principles whose according rewards are aligned with the ground truth. We tested different LLMs with principles generated by themselves and the filtered principles, and compared them with the default setting with no principle guidance. The results are shown in Table 1. We found that the self-generated principles barely improve performance, but the filtered principles could significantly boost the reward quality. This indicates that proper principles better guide reward generation under correctly summoned criteria. Details are depicted in Appendix D.

Method	Chat Hard	IFEval
GPT-4o-2024-08-06	76.1	56.0
w/ Self-Gen. Principles	75.9	55.6
w/ Filtered Principles	77.8	57.5
Gemma-2-27B-it	59.1	56.1
w/ Self-Gen. Principles	64.0	55.8
w/ Filtered Principles	68.0	57.3

表 1: Preliminary experiments on the influence of principles on reward quality. The default setting of DeepSeek-GRM-27B includes self-generated principles.

3 Self-Principled Critique Tuning (SPCT)

Inspired from the preliminary results, we developed a novel approach for pointwise GRMs to learn generating adaptive and high-quality principles that could effectively guide the generation of critiques, termed **Self-Principled Critique Tuning** (SPCT). As shown in Figure 3, SPCT consists of two phases: rejective fine-tuning, as the cold start, and rule-based online RL, reinforcing generalist reward generation by advancing the generated principles and critiques. SPCT fosters these behaviors in GRMs for inference-time scaling as well.

3.1 Unpinning Principles from Understanding to Generation

From preliminary experiments in Section 2.2, we found that proper principles could guide reward generation within certain criteria, which is critical for high-quality rewards. How-

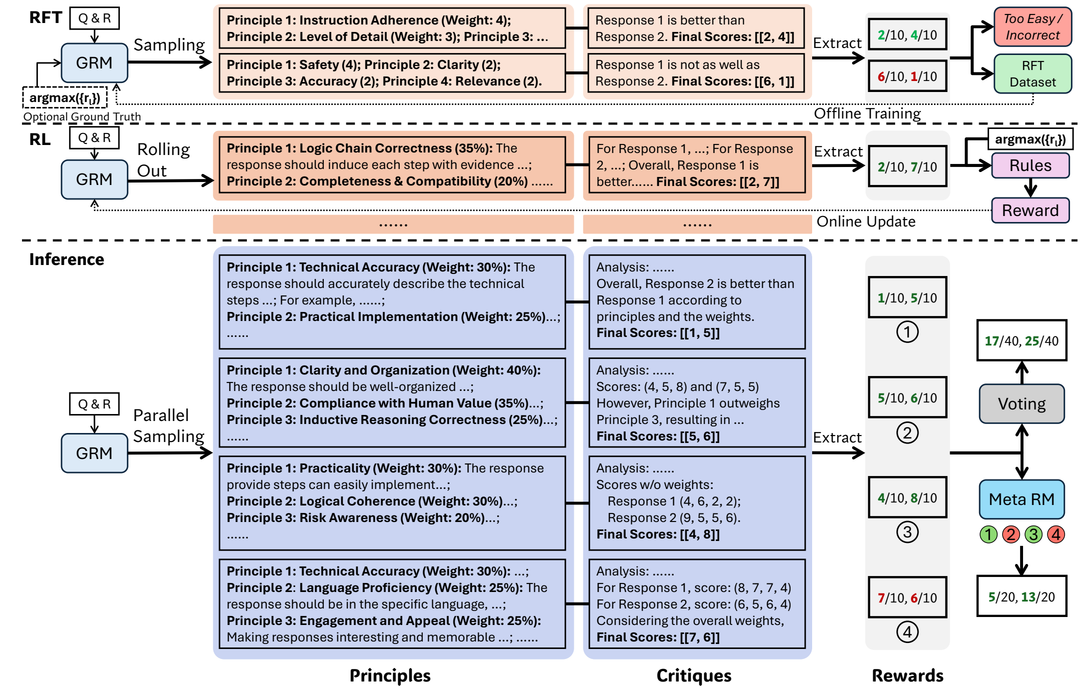


图 3: SPCT的插图，包括拒绝微调、基于规则的强化学习（RL）以及推理期间相应的可扩展行为。通过大规模生成的原则，推理时间的扩展是通过朴素投票或元RM引导的投票实现的，这导致在扩展的价值空间内获得更细粒度的结果奖励。

通过推进生成的原则和批评来增强通用奖励生成。SPCT还促进了GRMs在推理时间上的扩展行为。

3.1 Unpinning Principles from Understanding to Generation

从第 2.2节的初步实验中，我们发现适当的准则可以在特定标准内引导奖励生成，这对于高质量的奖励至关重要。然而，为通用奖励模型（RM）大规模生成有效的原则仍然具有挑战性。为了解决这一挑战，我们提出将原则从理解转向生成，即视原则为奖励生成的一部分，而不是预处理步骤。

形式上，当原则预先定义时，原则按照Equation 2引导奖励生成。生成式奖励模型（GRMs）可以自行生成原则，然后基于这些原则生成批评内容，其形式化表达为

$$\{p_i\}_{i=1}^m \sim p_\theta\left(x, \{y_i\}_{i=1}^n\right), \quad \mathcal{R} = \boldsymbol{C} \sim r_\theta\left(x, \{y_i\}_{i=1}^n, \{p_i\}_{i=1}^m\right), \tag{3}$$

其中，p_θ 是由 θ 参数化的主生成函数，它与奖励生成 r_θ 共享相同的模型。**这种转变使得原则可以根据输入的查询和响应自适应地生成,并与奖励生成过程对齐,同时通过在 GRM 上的后训练可以进一步提升原则及其对应批评的质量和颗粒度。** 随着大规模生成的原则，GRM 有可能在更合理的标准下以及以更精细的颗粒度输出奖励，这对推理时间的扩展同样至关重要。

3.2 Rule-Based Reinforcement Learning

为了同时优化 GRMs 中的原则生成和批评生成，我们提出了 SPCT，它集成了 拒绝微调 和 基于规则的强化学习(RL)。前者作为冷启动。

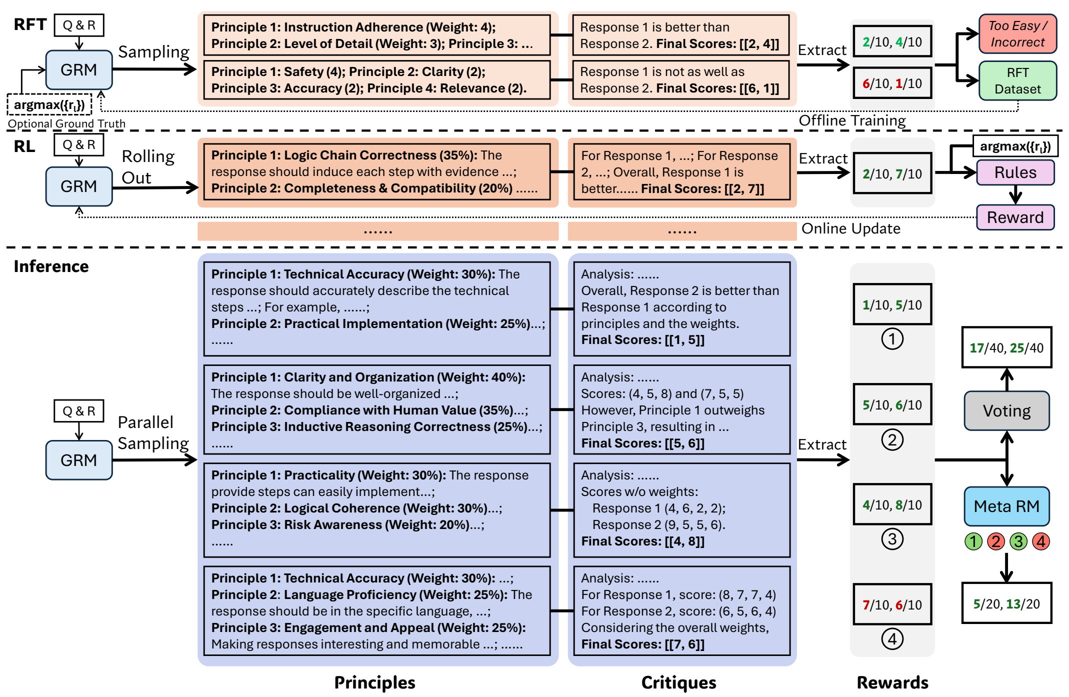


图 3: Illustration of SPCT, including rejective fine-tuning, rule-based RL, and corresponding scalable behaviors during inference. The inference-time scaling is achieved via naive voting or meta RM guided voting with principles generated at scale, resulting in finer-grained outcome rewards within a expanded value space.

ever, it remains challenging to generate effective principles for generalist RM at scale. To address this challenge, we propose to unpin principles from understanding to generation, i.e. view principles as a part of reward generation instead of a preprocessing step.

Formally, principles guide the generation of rewards following Equation 2, when principles are pre-defined. GRMs could generate principles themselves, and then generate critiques based on the principles, formalized as

$$\{p_i\}_{i=1}^m \sim p_\theta\left(x, \{y_i\}_{i=1}^n\right), \quad \mathcal{R} = \boldsymbol{C} \sim r_\theta\left(x, \{y_i\}_{i=1}^n, \{p_i\}_{i=1}^m\right), \tag{3}$$

where p_θ is the principle generation function parameterized by θ, that shares the same model with reward generation r_θ. **This shift enables to principles to be generated based on the input query and responses, adaptively aligning reward generation process, and the quality and granularity of the principles and corresponding critiques could be further improved with post-training on the GRM.** With the principles generated at scale, the GRM could potentially output rewards within more reasonable criteria and with finer granularity, which is crucial for inference-time scaling as well.

3.2 Rule-Based Reinforcement Learning

To optimize principle and critique generation in GRMs simultaneously, we propose SPCT, which integrates rejective fine-tuning and rule-based RL. The former serves as a cold start.

Rejective Fine-Tuning (Cold Start) The core idea of the rejective fine-tuning stage is to accomodate the GRM to generate principles and critiques with correct format and for

Preprint. Under review.

拒绝微调（冷启动） 拒绝微调阶段的核心思想是使 GRM 适应生成格式正确且适用于各种输入类型的原则和批评。与之前的工作 (Vu et al., 2024; Cao et al., 2024; Alexandru et al., 2025) 不同，这些工作混合了不同格式的单个、成对和多响应的 RM 数据，而我们在第 2.1 节中介绍的逐点 GRM 可以灵活地为任何数量的相同格式的响应生成奖励。在数据构建方面，除了通用指令数据外，我们还从包含各种响应数量的 RM 数据中采样轨迹，给定查询及其响应，使用预训练的 GRM 进行采样。对于每个查询及其对应的响应，采样执行 N_{RFT} 次。拒绝策略也得到了统一，即拒绝预测奖励与真实值不一致（错误）的轨迹，并且拒绝所有 N_{RFT} 条轨迹均正确的查询和响应（过于简单）。形式上，设 r_i 表示查询 x 的第 i 个响应 y_i 的真实奖励，预测的逐点奖励 $\{S_i\}_{i=1}^n$ 是正确的如果

$$\begin{cases} \forall i \neq j, \quad S_j > S_i, \quad j = \arg\max_l \{r_l\}_{l=1}^n, & \text{if } n \geq 2, \\ S_1 = r_1, & \text{if } n = 1. \end{cases} \tag{4}$$

保证了真实奖励中仅包含一个最大值。然而，类似于之前的研究工作 (Zhang et al., 2025a)，我们发现预训练的GRM在有限的采样配额内，难以正确生成部分查询及其对应响应的奖励。因此，我们可选地将$\arg\max_l \{r_l\}_{l=1}^n$附加到GRM的提示中，称为提示采样，期望预测的奖励能与真实值对齐，除此之外还有非提示采样。对于提示采样，每个查询及其对应的响应只采样一次，并且只有在结果不正确时才拒绝轨迹。超越以往研究 (Li et al., 2024a; Mahan et al., 2024)，我们观察到提示采样的轨迹有时会简化生成的批评，特别是在推理任务中，这表明在线强化学习（RL）对于GRM的必要性和潜在优势。

基于规则的强化学习 GRM进一步通过基于规则的在线强化学习进行微调。具体来说，我们使用了GRPO (Shao et al., 2024)的原始设置，并结合基于规则的结果奖励。在展开过程中，GRM根据输入查询和响应生成原则和批评，然后提取预测的奖励，并通过准确性规则与真实奖励进行比较。与DeepSeek-AI (2025)不同，这里没有使用格式奖励。相反，应用了一个更大的KL惩罚系数以确保格式并避免严重偏差。形式上，给定查询x和响应$\{y_i\}_{i=1}^n$，第i个输出o_i的奖励为

$$r_i = \begin{cases} 1, & \text{if } n \geq 2 \text{ and } \forall j \neq i, \quad S_i > S_j, \quad j = \arg\max_l \{r_l\}_{l=1}^n, \\ 1, & \text{if } n = 1 \text{ and } S_1 = r_1, \\ -1, & \text{otherwise}, \end{cases} \tag{5}$$

其中，逐点奖励 $\{S_i\}_{i=1}^n$ 从 o_i 中提取。 **奖励函数鼓励 GRMs 根据在线优化的原则和批评来区分最佳响应,从而有利于有效的推理时间扩展。** 奖励信号可以无缝地从任何偏好数据集和标记的大型语言模型响应中获得。

4 Inference-Time Scaling with SPCT

为了进一步提高 DeepSeek-GRM在使用更多推理计算资源时生成通用奖励的性能，我们探索了基于采样的策略以实现有效的推理时间可扩展性。之前工作中的推理时间扩展方法及其潜在局限性在附录 C.1 中进行了分析 (Wang et al., 2024c; Ankner et al., 2024; Mahan et al., 2024; Zhang et al., 2025a)。

various input types. Unlike previous works (Vu et al., 2024; Cao et al., 2024; Alexandru et al., 2025) that mix RM data for single, paired, and multiple responses in different formats, we adopt pointwise GRM, introduced in Section 2.1, to flexibly generate rewards for any amount of responses in the same format. For data construction, besides general instruction data, we sample trajectories with pretrained GRMs given the query and responses to the query from RM data with various response counts. For each query and corresponding responses, the sampling is performed N_{RFT} times. The rejection strategy is also unified, which is to reject trajectories with predicted rewards that are not aligned with the ground truth (incorrect), and the query and responses with all N_{RFT} trajectories correct (too easy). Formally, let r_i denotes the ground truth reward for the i-th response y_i to the query x, the predicetd pointwise rewards $\{S_i\}_{i=1}^n$ are correct if

$$\begin{cases} \forall i \neq j, \quad S_j > S_i, \quad j = \arg\max_l \{r_l\}_{l=1}^n, & \text{if } n \geq 2, \\ S_1 = r_1, & \text{if } n = 1. \end{cases} \tag{4}$$

with guaranteed that the ground truth rewards only contain one maximum. However, similar to previous works (Zhang et al., 2025a), we found pretrained GRMs could hardly generate correct rewards for a portion of queries and corresponding responses within limited sampling quota. Thus, we optionally append $\arg\max_l \{r_l\}_{l=1}^n$ to the prompt of the GRM, termed *hinted sampling*, expecting the predicted rewards to align with the ground truth, besides *non-hinted sampling*. For hinted sampling, each query and the corresponding responses are sampled once, and trajectories are only rejected when incorrect. Beyond previous studies (Li et al., 2024a; Mahan et al., 2024), we observed that hinted sampled trajectories sometimes shortcut the generated critique, especially for reasoning tasks, indicating the necessity and potential benefits of online RL for the GRM.

Rule-Based RL The GRM is further fine-tuned with rule-based online RL. Specifically, we use the original setting of GRPO (Shao et al., 2024) with rule-based outcome rewards. During rolling out, the GRM generates principles and critiques based on the input query and responses, and then the predicted reward is extracted and compared to the ground truth with accuracy rules. Unlike DeepSeek-AI (2025), no format rewards are used. Instead, a larger coefficient for KL penalty is applied to ensure the format and avoid severe biases. Formally, the reward for the i-th output o_i to the given query x and responses $\{y_i\}_{i=1}^n$ is

$$r_i = \begin{cases} 1, & \text{if } n \geq 2 \text{ and } \forall j \neq i, \quad S_i > S_j, \quad j = \arg\max_l \{r_l\}_{l=1}^n, \\ 1, & \text{if } n = 1 \text{ and } S_1 = r_1, \\ -1, & \text{otherwise,} \end{cases} \tag{5}$$

where the pointwise rewards $\{S_i\}_{i=1}^n$ are extracted from o_i. **The reward function encourages GRMs to distinguish the best responses with online optimized principles and critiques, in favor of effective inference-time scaling.** The reward signal could be obtained seamlessly from any preference dataset and labeled LLM responses.

4 Inference-Time Scaling with SPCT

To further improve the performance of DeepSeek-GRM for generalist reward generation using more inference compute, we explores sampling-based strategies to achieve effective

Preprint. Under review.

Model	Reward Bench	PPE Preference	PPE Correctness	RMB	Overall
	Reported Results of Public Models				
Skywork-Reward-Gemma-2-27B	*94.1*	*56.6*	*56.6*	*60.2*	*66.9*
DeepSeek-V2.5-0905	81.5	62.8	58.5	65.7	67.1
Gemini-1.5-Pro	86.8	66.1	59.8	56.5	67.3
ArmoRM-8B-v0.1	*90.4*	*60.6*	*61.2*	*64.6*	*69.2*
InternLM2-20B-Reward	*90.2*	*61.0*	*63.0*	*62.9*	*69.3*
LLaMA-3.1-70b-Instruct	84.1	65.3	59.2	68.9	69.4
Claude-3.5-sonnet	84.2	65.3	58.8	70.6	69.7
Nemotron-4-340B-Reward	*92.0*	*59.3*	*60.8*	*69.9*	*70.5*
GPT-4o	86.7	67.1	57.6	73.8	71.3
	Reproduced Results of Baseline Methods				
LLM-as-a-Judge	83.4	64.2	58.8	64.8	67.8
DeepSeek-BTRM-27B	*81.7*	***68.3***	***66.7***	*57.9*	*68.6*
CLoud-Gemma-2-27B	*82.0*	*67.1*	*62.4*	*63.4*	*68.7*
DeepSeek-PairRM-27B	*87.1*	*65.8*	*64.8*	*58.2*	*69.0*
	Results of Our Method				
DeepSeek-GRM-27B-RFT (Ours)	84.5	64.1	59.6	67.0	68.8
DeepSeek-GRM-27B (Ours)	86.0	64.7	59.8	69.0	69.9
	Results of Inference-Time Scaling (Voting@32)				
DeepSeek-GRM-27B (Ours)	88.5	65.3	60.4	69.0	71.0
DeepSeek-GRM-27B (MetaRM) (Ours)	**90.4**	67.2	63.2	**70.3**	**72.8**

表 2: 在RM基准上不同方法和模型的整体结果。 下划线数字 表示最佳性能，**粗体数字** 表示基线和我们方法中的最佳性能，斜体字 表示标量或半标量RMs。对于元RM引导的投票（MetaRM），$k_{\text{meta}} = \frac{1}{2}k$。

带生成奖励的投票 回顾第 2.1 节中提到的方法，点wise GRMs 的投票过程被定义为对奖励进行求和：

$$S_i^* = \sum_{j=1}^{k} S_{i,j}, \quad \left\{ \{S_{i,j}\}_{i=1}^n = f_{\text{point}}(C_j, \{y_i\}_{i=1}^n) \sim r_\theta \left(x, \{y_i\}_{i=1}^n, \{p_{i,j}\}_{i=1}^{m_j} \right) \right\}_{j=1}^k \sim p_\theta \left(x, \{y_i\}_{i=1}^n \right), \tag{6}$$

其中 S_i^* 是第 i 个回应的最终奖励 ($i = 1, ..., n$)。 **由于 $S_{i,j}$ 通常被设定在一个小的离散范围内， 例如 $\{1, ..., 10\}$，投票过程实际上将奖励空间扩展了 k 倍,并使 GRM 能够生成大量原则,这有助于提高最终奖励的质量和粒度**。一个直观的解释是，如果每个原则都被视为判断视角的代理，更多的原则可能更准确地反映真实分布，从而提升效果。值得注意的是，为了避免位置偏差并增加多样性，在采样前会对回应进行随机打乱。

元奖励建模引导的投票 在 DeepSeek-GRM的投票过程中，需要多次采样，但由于随机性或模型限制，部分生成的原则和批评可能是有偏见或低质量的。因此，我们训练了一个元 RM 来指导投票过程。元 RM 是一个逐点标量 RM，通过二元交叉熵损失训练，用于识别由 DeepSeek-GRM生成的原则和批评的正确性，标签根据 Equation 4 确定。数据集包括 RFT 阶段非提示采样的轨迹，以及来自待引导的 DeepSeek-GRM的采样轨迹，以提供足够的正负奖励，并缓解训练与推理策略之间的差距（如 Chow et al. (2025) 所建议）。引导投票的过程很简单：元 RM 为 k 个采样奖励输出元奖励，最终结果由具有前 $k_{\text{meta}} \leq k$ 个元奖励的奖励投票决定，从而过滤掉低质量样本。

5 Results on Reward Modeling Benchmarks

5.1 Experiment Settings

基准测试和评估指标 我们在不同领域的各种RM基准上评估了不同方法的性能：**Reward Bench** (Lambert et al., 2024)，**PPE** (Frick et al., 2025)，**RMB** (Zhou et al., 2025)，**ReaLMistake** (Kamoi et al., 2024)。我们使用每个基准的标准评估指标：在Reward Bench、PPE和RMB中从一组响应中挑选最佳响应的准确性，以及ReaLMistake中的ROC-AUC。为

inference-time scalability. Inference-time scaling methods from previous works (Wang et al., 2024c; Ankner et al., 2024; Mahan et al., 2024; Zhang et al., 2025a), and their potential limitations are analyzed in Appendix C.1.

Voting with Generated Rewards Recalling the approaches in Section 2.1, the voting process for pointwise GRMs is defined as summing the rewards:

$$S_i^* = \sum_{j=1}^{k} S_{i,j}, \quad \left\{ \{S_{i,j}\}_{i=1}^{n} = f_{\text{point}}(C_j, \{y_i\}_{i=1}^{n}) \sim r_\theta \left(x, \{y_i\}_{i=1}^{n}, \{p_{i,j}\}_{i=1}^{m_j} \right) \right\}_{j=1}^{k} \sim p_\theta \left(x, \{y_i\}_{i=1}^{n} \right), \tag{6}$$

where S_i^* is the final reward for the i-th response ($i = 1, ..., n$). **Since $S_{i,j}$ is usually set within a small discrete range,** e.g., $\{1, ..., 10\}$, **the voting process actually expands the reward space by k times, and enables the GRM to generate a large amount of principles, which benefits the quality and granularity of the final rewards**. An intuitive explanation is that, if each principle could be viewed as a proxy of judgement perspectives, a larger number of principles may reflect the real distribution more accurately, leading to scaling effectiveness. Notably, to avoid positional biases and for diversity, responses are shuffled before sampling.

Meta Reward Modeling Guided Voting The voting process of DeepSeek-GRM requires multiple sampling and a few generated principles and critiques might be biased or low-quality due to randomness or model limitations. Thus, we train a meta RM to guide the voting process. The meta RM is a pointwise scalar RM, trained to identify the correctness of the principle and critique generated by DeepSeek-GRM, with the binary cross-entropy loss, where the label is identified based on Equation 4. The dataset comprises trajectories from non-hinted sampling in the RFT stage, and also trajectories sampled from the DeepSeek-GRM to be guided, to both provide enough positive and negative rewards and alleviate the gap between training and inference policy as suggested by Chow et al. (2025). The guided voting is simple: The meta RM outputs meta rewards for k sampled rewards, and the final outcome is voted by rewards with top $k_{\text{meta}} \leq k$ meta rewards, so that filtering out low-quality samples.

5 Results on Reward Modeling Benchmarks

5.1 Experiment Settings

Benchmarks and Evaluation Metrics We evaluate the performance of different methods on various RM benchmarks of different domains: **Reward Bench** (Lambert et al., 2024), **PPE** (Frick et al., 2025), **RMB** (Zhou et al., 2025), **ReaLMistake** (Kamoi et al., 2024). We use the standard evaluation metrics for each benchmark: accuracy of picking the best response from a set of responses in Reward Bench, PPE, and RMB, and ROC-AUC for ReaLMistake. To deal with ties of the predicted rewards for multiple responses, we shuffle the responses and determine the best response by $\arg\max_i S_i$, where S_i is the predicted reward for the i-th response after shuffling. Details are in Appendix D.

Method Implementation For the baseline methods, we re-implement **LLM-as-a-Judge** (Zheng et al., 2023), **DeepSeek-BTRM-27B** (Kendall & Smith, 1940), **CLoud-Gemma-2-27B** (Ankner et al., 2024), and **DeepSeek-PairRM-27B** (Jiang et al., 2023) based on Gemma-2-27B (Team, 2024) and with all compatible training data and settings as DeepSeek-GRM. For our

Preprint. Under review.

Model	Overall
Reported Results of Public Models	
Nemotron-4-340B-Reward	*70.5*
GPT-4o	71.3
Results of Inference-Time Scaling (Voting@1)	
LLM-as-a-Judge	67.0
CLoud-Gemma-2-27B	*68.5*
DeepSeek-GRM-27B-RFT (Ours)	67.8
DeepSeek-GRM-27B (Ours)	67.9
Results of Inference-Time Scaling (Voting@8)	
LLM-as-a-Judge	67.6 (+0.6)
LLM-as-a-Judge w/ **TokenProb**	68.1 (+1.1)
CLoud-Gemma-2-27B	*68.8* (+0.3)
DeepSeek-GRM-27B-RFT (Ours)	69.3 (+1.5)
DeepSeek-GRM-27B (Ours)	70.6 (+2.7)
DeepSeek-GRM-27B (MetaRM) (Ours)	72.0 (+4.1)
Results of Further Inference-Time Scaling (Voting@32)	
DeepSeek-GRM-27B (Ours)	71.0 (+3.1)
DeepSeek-GRM-27B (MetaRM) (Ours)	**72.8** (+4.9)

表 3: 在RM基准上不同方法的推理时间可扩展性结果。设置与表 2相同。

Method	Overall
Results of Greedy Decoding	
DeepSeek-GRM-27B	**69.9**
w/o Principle Generation	67.5
w/o Rejective Sampling	68.7
DeepSeek-GRM-27B-RFT	**68.8**
w/o Hinted Sampling (①)	68.0
w/o Non-Hinted Sampling (②)	67.4
w/o Rejective Sampling (①&②)	66.1
w/o General Instruction Data	63.3
Results of Inference-Time Scaling (Voting@8)	
DeepSeek-GRM-27B	**70.6**
w/o Principle Generation	68.0
Results of Inference-Time Scaling (Voting@32)	
DeepSeek-GRM-27B	71.0
DeepSeek-GRM-27B ($k_{\text{meta}} = 1$)	71.5
DeepSeek-GRM-27B ($k_{\text{meta}} = 8$)	72.7
DeepSeek-GRM-27B ($k_{\text{meta}} = 16$)	**72.8**

表 4: 针对所提出的SPCT的不同组件进行消融研究。**粗体数字**表示最佳性能。

了处理多个响应预测奖励的平局情况，我们对响应进行随机打乱，并通过$\arg\max_i S_i$确定最佳响应，其中S_i是打乱后第i个响应的预测奖励。详细信息见附录 D。

方法实现 对于基线方法，我们重新实现了**LLM-as-a-Judge** (Zheng et al., 2023)，**DeepSeek-BTRM-27B** (Kendall & Smith, 1940)，**CLoud-Gemma-2-27B** (Ankner et al., 2024) 和 **DeepSeek-PairRM-27B** (Jiang et al., 2023)，这些方法基于Gemma-2-27B (Team, 2024)，并使用与DeepSeek-GRM兼容的所有训练数据和设置。对于我们的方法，我们基于Gemma-2-27B实现了**DeepSeek-GRM-27B-RFT**，并在不同规模的LLMs上实现了**DeepSeek-GRM**，包括DeepSeek-V2-Lite (16B MoE) (DeepSeek-AI, 2024a)，Gemma-2-27B，DeepSeek-V2.5 (236B MoE)，以及DeepSeek-V3 (671B MoE) (DeepSeek-AI, 2024b)。元RM在Gemma-2-27B上进行训练。默认结果报告采用贪婪解码，推理时扩展使用temperature = 0.5。其他详细信息见附录 C.2。

5.2 Results and Analysis

RM基准测试的表现 不同方法和模型在RM基准测试上的整体结果如表 2所示。我们将DeepSeek-GRM-27B的表现与公开模型的报告结果及基线方法的重现结果进行了比较。我们发现，DeepSeek-GRM-27B在整体表现上优于基线方法，并且与强大的公开RM（例如Nemotron-4-340B-Reward和GPT-4o）具有竞争力；通过推理时间扩展，DeepSeek-GRM-27B可以进一步提升并达到最佳的整体结果。对于详细比较，标量（DeepSeek-BTRM-27B、DeepSeek-PairRM-27B）和半标量（CLoud-Gemma-2-27B）RM在不同基准测试上表现出偏向性结果，在可验证任务（PPE Correctness）上的表现显著优于所有生成式RM，但在其他不同基准测试中分别失败。然而，大多数公开的标量RM也表现出严重的领域偏差。LLM-as-a-Judge的表现趋势与DeepSeek-GRM-27B相似，但性能较低，这可能是由于缺乏原则性指导。综上所述，**SPCT提升了GRMs的一般奖励生成能力,并且相比标量和半标量RM表现出显著较少的偏差**。

推理时间的可扩展性 不同方法的推理时间扩展结果如表 3 所示，整体趋势在图 1 中展示。详细信息见附录 D.3。在最多 8 个样本的情况下，我们发现 DeepSeek-GRM-27B在贪婪解码和采样结果方面具有最高的性能提升。此外，DeepSeek-GRM-27B在更大的推理计算资源（最多 32 个样本）下表现出进一步提高性能的强大潜力。元 RM (meta RM) 还揭示了其在

Model	Reward Bench	PPE Preference	PPE Correctness	RMB	Overall
		Reported Results of Public Models			
Skywork-Reward-Gemma-2-27B	*94.1*	*56.6*	*56.6*	*60.2*	*66.9*
DeepSeek-V2.5-0905	81.5	62.8	58.5	65.7	67.1
Gemini-1.5-Pro	86.8	66.1	59.8	56.5	67.3
ArmoRM-8B-v0.1	*90.4*	*60.6*	*61.2*	*64.6*	*69.2*
InternLM2-20B-Reward	*90.2*	*61.0*	*63.0*	*62.9*	*69.3*
LLaMA-3.1-70b-Instruct	84.1	65.3	59.2	68.9	69.4
Claude-3.5-sonnet	84.2	65.3	58.8	70.6	69.7
Nemotron-4-340B-Reward	*92.0*	*59.3*	*60.8*	*69.9*	*70.5*
GPT-4o	86.7	67.1	57.6	73.8	71.3
		Reproduced Results of Baseline Methods			
LLM-as-a-Judge	83.4	64.2	58.8	64.8	67.8
DeepSeek-BTRM-27B	*81.7*	***68.3***	***66.7***	*57.9*	*68.6*
CLoud-Gemma-2-27B	*82.0*	*67.1*	*62.4*	*63.4*	*68.7*
DeepSeek-PairRM-27B	*87.1*	*65.8*	*64.8*	*58.2*	*69.0*
		Results of Our Method			
DeepSeek-GRM-27B-RFT (Ours)	84.5	64.1	59.6	67.0	68.8
DeepSeek-GRM-27B (Ours)	86.0	64.7	59.8	69.0	69.9
		Results of Inference-Time Scaling (Voting@32)			
DeepSeek-GRM-27B (Ours)	88.5	65.3	60.4	69.0	71.0
DeepSeek-GRM-27B (MetaRM) (Ours)	**90.4**	67.2	63.2	**70.3**	**72.8**

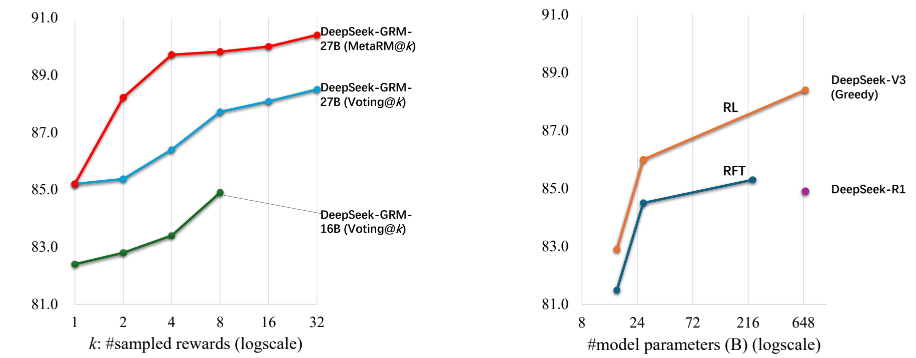
表 2: Overall results of different methods and models on RM benchmarks. Underlined numbers indicate the best performance, **bold numbers** indicate the best performance among baseline and our methods, and *italicized font* denotes scalar or semi-scalar RMs. For meta RM guided voting (MetaRM), $k_{\text{meta}} = \frac{1}{2}k$.

methods, we implement **DeepSeek-GRM-27B-RFT** based on Gemma-2-27B, and **DeepSeek-GRM** on different sizes of LLMs, including DeepSeek-V2-Lite (16B MoE) (DeepSeek-AI, 2024a), Gemma-2-27B, DeepSeek-V2.5 (236B MoE), and DeepSeek-V3 (671B MoE) (DeepSeek-AI, 2024b). The meta RM is trained on Gemma-2-27B. Default results are reported with greedy decoding, and the inference-time scaling uses temperature $= 0.5$. Other details are provided in Appendix C.2.

5.2 Results and Analysis

Performance on RM Benchmarks The overall results of different methods and models on RM benchmarks are shown in Table 2. We compare the performance of DeepSeek-GRM-27B with the reported results of public models and the reproduced results of baseline methods. We find that DeepSeek-GRM-27B outperforms the baseline methods in overall performance, and achieves competitive performance with strong public RMs, such as Nemotron-4-340B-Reward and GPT-4o; with inference-time scaling, DeepSeek-GRM-27B could further improve and achieve the best overall results. For detailed comparisons, scalar (DeepSeek-BTRM-27B, DeepSeek-PairRM-27B) and semi-scalar (CLoud-Gemma-2-27B) RMs demonstrate biased results on different benchmarks, with significant better performance on verifiable tasks (PPE Correctness) than all generative RMs, but fail in different other benchmarks, respectively. Nonetheless, most public scalar RMs also exhibit severe domain biases. LLM-as-a-Judge shows similar trends with DeepSeek-GRM-27B with lower performance, potentially due to the lack of principle guidance. In conclusion, **SPCT improves the generalist reward generation capability of GRMs, with significantly less biases compared to scalar and semi-scalar RMs**.

Inference-Time Scalability The inference-time scaling results of different methods are shown in Table 3, and the whole trends are demonstrated in Figure 1. Details are in Appendix



(a) Inference-time scaling results of DeepSeek-GRM-16B and DeepSeek-GRM-27B ($k_{\text{meta}} = \frac{1}{2}k$).

(b) Training-time scaling results with different model sizes, using greedy decoding except DeepSeek-R1.

图 4: 推理时扩展性能与训练时扩展性能在Reward Bench基准上的对比。

每个基准测试中对 DeepSeek-GRM筛选低质量轨迹的有效性。通过标记概率投票，LLM-as-a-Judge 也显示出显著的性能提升，这表明标记概率作为定量权重可以提高单纯多数投票的可靠性。对于 CLoud-Gemma-2-27B，性能提升有限，主要原因是标量奖励生成缺乏多样性，尽管批评部分已经发生了很大变化。总结来说，**SPCT 提高了 GRMs 的推理时间可扩展性,而元 RM 进一步提升了总体的扩展性能**。

消融研究 表 4 显示了所提出的 SPCT 不同组件的消融研究结果，详细结果列在附录 D.3 中。令人惊讶的是，即使没有使用拒绝采样的冷启动数据，经过在线强化学习训练后，通用指令调优的 GRMs 性能仍然显著提升 (66.1 → 68.7)。此外，非提示采样似乎比提示采样更重要，可能是因为提示采样轨迹中出现了捷径现象。这些结果表明了 **在线训练对 GRMs 的重要性**。与之前的研究一致 (Cao et al., 2024)，我们确认通用指令数据对 GRMs 的性能至关重要。我们发现，**原则生成对 DeepSeek-GRM-27B的贪婪解码性能和推理时间扩展性能都至关重要**。对于推理时间扩展，元 RM 指导的投票在不同的 k_{meta} 下表现出较强的鲁棒性。关于通用主义 RM 的性能分析，包括输入灵活性、训练数据的领域泛化能力等，详见附录 E。

扩展推理和训练成本 我们进一步研究了通过不同规模的 LLMs 进行后训练后的 DeepSeek-GRM-27B的推理时间和训练时间扩展性能。模型在 Reward Bench 上进行了测试，结果如图 4 所示。我们发现，使用 32 个样本直接投票的 DeepSeek-GRM-27B可以达到与 671B MoE 模型相当的性能，而元 RM 指导的投票仅需 8 个样本即可获得最佳结果，这证明了 **DeepSeek-GRM-27B的推理时间扩展相对于扩展模型规模的有效性**。此外，我们使用包含 300 个样本的降采样测试集测试了 DeepSeek-R1，并发现其性能甚至不如 236B MoE RFT 模型，这表明扩展长链推理并不能显著提升通用主义 RM 的性能。

6 Related Work

生成式奖励模型 GRMs（生成式奖励模型）标志着从标量奖励模型（RMs）(Ouyang et al., 2022) 的范式转变，将奖励建模为文本反馈或评分。 (Li et al., 2024a; Kim et al., 2024; Wang et al., 2024c; Cao et al., 2024; Vu et al., 2024; Alexandru et al., 2025)，这使得奖励表示更加丰富，并且更灵活地评估单个或多个响应。先前的方法中，LLM-as-a-judge 方法 (Zheng et al., 2023) 可以进行基于参考或无参考的成对判断以评估大语言模型（LLMs）。最近的研究使用离线强化学习方法，例如 DPO (Rafailov et al., 2023) 来训练 GRMs (Wu et al., 2024; Mahan et al., 2024; Yu et al., 2025a; Ye et al., 2025)，并将工具和外部知识与 GRMs 结合 (Li et al., 2024b; Peng et al., 2025)，甚至将 GRMs 训练为一个接口以调整来自环境的奖励 (Baker

Model	Overall
Reported Results of Public Models	
Nemotron-4-340B-Reward	*70.5*
GPT-4o	71.3
Results of Inference-Time Scaling (Voting@1)	
LLM-as-a-Judge	67.0
CLoud-Gemma-2-27B	*68.5*
DeepSeek-GRM-27B-RFT (Ours)	67.8
DeepSeek-GRM-27B (Ours)	67.9
Results of Inference-Time Scaling (Voting@8)	
LLM-as-a-Judge	67.6 (+0.6)
LLM-as-a-Judge w/ **TokenProb**	68.1 (+1.1)
CLoud-Gemma-2-27B	*68.8* (+0.3)
DeepSeek-GRM-27B-RFT (Ours)	69.3 (+1.5)
DeepSeek-GRM-27B (Ours)	70.6 (+2.7)
DeepSeek-GRM-27B (MetaRM) (Ours)	72.0 (+4.1)
Results of Further Inference-Time Scaling (Voting@32)	
DeepSeek-GRM-27B (Ours)	71.0 (+3.1)
DeepSeek-GRM-27B (MetaRM) (Ours)	<u>72.8</u> (+4.9)

表 3: Inference-time scalability results of different methods on RM benchmarks. Settings are the same as Table 2.

Method	Overall
Results of Greedy Decoding	
DeepSeek-GRM-27B	**69.9**
w/o Principle Generation	67.5
w/o Rejective Sampling	68.7
DeepSeek-GRM-27B-RFT	**68.8**
w/o Hinted Sampling (①)	68.0
w/o Non-Hinted Sampling (②)	67.4
w/o Rejective Sampling (①&②)	66.1
w/o General Instruction Data	63.3
Results of Inference-Time Scaling (Voting@8)	
DeepSeek-GRM-27B	**70.6**
w/o Principle Generation	68.0
Results of Inference-Time Scaling (Voting@32)	
DeepSeek-GRM-27B	71.0
DeepSeek-GRM-27B ($k_{\text{meta}} = 1$)	71.5
DeepSeek-GRM-27B ($k_{\text{meta}} = 8$)	72.7
DeepSeek-GRM-27B ($k_{\text{meta}} = 16$)	**72.8**

表 4: Ablation studies for different components of the proposed SPCT. **Bold numbers** indicate the best performance.

D.3. With up to 8 samples, we find that DeepSeek-GRM-27B has the highest performance increase to the greedy decoding and sampling results. DeepSeek-GRM-27B further shows a strong potential to increase the performance with larger inference compute, up to 32 samples. The meta RM also reveals its validity in filtering low-quality trajectories for DeepSeek-GRM on each benchmark. Voted with token probabilities, LLM-as-a-Judge also shows a significant performance increase, indicating that the token probability as quantitative weights could help the reliability of mere majority voting. For CLoud-Gemma-2-27B, the performance increase is limited, mainly due to the lack of variance in scalar reward generation, even though the critique has changed a lot. In summary, **SPCT improves the inference-time scalability of GRMs, and the meta RM further boosts the scaling performance in general**.

Ablation Study Table 4 shows the ablation study results of different components of the proposed SPCT, detailed results are listed in Appendix D.3. Surprisingly, without the cold start with rejective sampled critique data, general instruction tuned GRMs still improve significantly after undergoing the online RL (66.1 → 68.7). Also, the non-hinted sampling seems more important than the hinted sampling, potentially because of the shortcuts appeared in hinted sampled trajectories. These indicate the **importance of online training for GRMs**. Aligned with previous works (Cao et al., 2024), we confirm that the general instruction data is essential for the performance of GRMs. We find that **the principle generation is crucial for the performance of both greedy decoding and inference-time scaling of DeepSeek-GRM-27B**. For inference-time scaling, the meta RM guided voting shows robustness with different k_{meta}. Further analysis on the generalist RM performance, including input flexibility, domain generalization of training data, etc., is discussed in Appendix E.

Scaling Inference and Training Costs We further investigate the inference-time and training-time scaling performance of DeepSeek-GRM-27B, by post-training with LLMs in different sizes. The models are tested on the Reward Bench, and the results are shown in Figure 4. We find that direct voting with 32 samples of DeepSeek-GRM-27B could achieve comparable performance with the 671B MoE model, and the meta RM guided voting could

Preprint. Under review.

et al., 2025)。尽管这些方法在效率方面面临挑战，但它们展示了在大规模改进奖励方面的潜力，朝着更通用的奖励系统迈进。

大语言模型的推理时间扩展 大语言模型（LLMs）的推理时间扩展一直是与训练时间扩展并行的关键研究方向。研究重点在于采样和奖励模型（RM）引导的聚合 (Lightman et al., 2024; Brown et al., 2024; Snell et al., 2025; Wu et al., 2025)。最近，由 LLMs 激励的长视野链式思维 (Wei et al., 2022) 在提高模型推理能力方面显示出有希望的结果 (OpenAI, 2024; DeepSeek-AI, 2025; OpenAI, 2025c)，这是另一种形式的推理时间扩展。还有研究使用可扩展的奖励或验证器来提高策略模型在编码 (Chen et al., 2023)、推理 (Lifshitz et al., 2025) 等领域的性能。因此，本工作中开发的推理时间可扩展的通用奖励模型（RMs）可能通过推理时间协同扩展来提升策略模型的整体性能。

7 Conclusion and Future Work

我们引入了自我原则批评调优（SPCT），这是一种增强通用奖励建模推理时间可扩展性的方法。通过基于规则的在线强化学习，SPCT 实现了原则和批评的自适应生成，显著提高了奖励质量以及在不同领域中通用奖励模型（GRM）的推理时间可扩展性。实证结果表明，DeepSeek-GRM超过了基线方法和一些强大的公开奖励模型（RMs），并在推理时间扩展方面显示出显著改进，尤其是在元奖励模型（meta RM）的指导下。

未来的研究方向可能包括将 GRM 集成到在线强化学习管道中作为奖励系统的多功能接口、探索与策略模型的推理时间协同扩展，或作为基础模型的强大离线评估器。

Ethics Statement

我们提出的方法，自原则批评微调（SPCT），旨在增强生成式奖励模型（GRMs）在通用领域的推理时间可扩展性。尽管这一进展促进了奖励建模的准确性和一致性，但仍需明确考虑若干伦理影响。

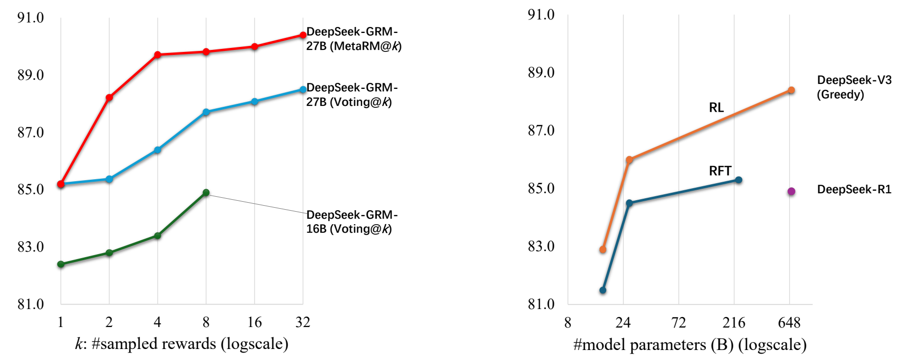
首先，尽管通过我们的实证分析表明 DeepSeek-GRM在不同领域表现出较少的偏见，但当训练数据有毒时，自动化的原则和批评生成可能会无意中延续或放大这些偏见。我们认为应优先进行元奖励模型（meta RM）及其他偏见缓解策略的研究，以确保公平的结果。此外，我们的方法并不旨在减少人类监督。相反，我们提倡保持“人在回路”（human-in-the-loop）框架，并开发如SPCT等可靠的代理方法，以更高效和有效地扩展人类监督。

其次，在各种领域中扩展推理时间可扩展的GRMs的应用可能会引发对透明度、问责制等方面的担忧。我们在第5.2节中展示了模型的能力，并在附录B中讨论了其局限性，同时在公共监督下开源该模型，这对于维持信任和确保该技术负责任地部署至关重要。

最后，在不同的奖励模型基准和实际场景中进行强有力的验证并保持持续警惕仍然至关重要。要合乎道德地使用DeepSeek-GRM，需要积极管理风险并持续评估其偏见，这需要在奖励模型评估研究方面的努力。

参考文献

Andrei Alexandru, Antonia Calvi, Henry Broomfield, Jackson Golden, Kyle Dai, Mathias Leys, Maurice Burger, Max Bartolo, Roman Engeler, Sashank Pisupati, Toby Drane, and Young Sun Park. Atla selene mini: A general purpose evaluation model. *Computing Research Repository*, arXiv:2501.17195, 2025. URL https://arxiv.org/abs/2501.17195.



(a) Inference-time scaling results of DeepSeek-GRM-16B and DeepSeek-GRM-27B ($k_{\text{meta}} = \frac{1}{2}k$).

(b) Training-time scaling results with different model sizes, using greedy decoding except DeepSeek-R1.

图 4: Inference-time scaling performance v.s. training-time scaling performance on the Reward Bench benchmark.

achieve the best results with 8 samples, demonstrating the **effectiveness of inference-time scaling of DeepSeek-GRM-27B compared to scaling model sizes**. Moreover, we test DeepSeek-R1 with a downsampled test set containing 300 samples, and find its performance even worse than the 236B MoE RFT model, indicating that expanding long chain-of-thoughts for reasoning tasks could not significantly improve the performance of generalist RM.

6 Related Work

Gnerative Reward Models GRMs represent a paradigm shift from scalar RMs (Ouyang et al., 2022), modeling reward as textual feedback or scores. (Li et al., 2024a; Kim et al., 2024; Wang et al., 2024c; Cao et al., 2024; Vu et al., 2024; Alexandru et al., 2025), enabling richer reward representations and more flexible to judge single and multiple responses. Priorly, LLM-as-a-judge method (Zheng et al., 2023) accommodates reference-based or reference-free pairwise judgement for evaluating LLMs. Recent studies use offline RL, e.g., DPO (Rafailov et al., 2023), to train GRMs (Wu et al., 2024; Mahan et al., 2024; Yu et al., 2025a; Ye et al., 2025), incorporate tools and external knowledge with GRMs (Li et al., 2024b; Peng et al., 2025), and even train GRMs as an interface to adjust rewards from environments (Baker et al., 2025). Though these methods face challenges in efficiency, they demonstrate the potential in improving rewards at scale, towards a more generalist reward system.

Inference-Time Scaling for LLMs Inference-time scaling for LLMs has been a critical research direction parallel with scaling LLMs in training time. Studies focus on sampling and RM guided aggregation (Lightman et al., 2024; Brown et al., 2024; Snell et al., 2025; Wu et al., 2025). Recently, long-horizon chain-of-thoughts (Wei et al., 2022) incentivized from LLMs show promising results in improving the reasoning capabilities of the models (OpenAI, 2024; DeepSeek-AI, 2025; OpenAI, 2025c), as another format of inference-time scaling. There are also researches using scalable rewards or verifiers to improve the performance of policy models, in domains of coding (Chen et al., 2023), reasoning (Lifshitz et al., 2025), etc. Thus, the development of inference-time scalable generalist RMs in this work might also contributes to the general performance of policy models by inference-time co-scaling.

Preprint. Under review.

Wei An, Xiao Bi, Guanting Chen, Shanhuang Chen, Chengqi Deng, Honghui Ding, Kai Dong, Qiushi Du, Wenjun Gao, Kang Guan, Jianzhong Guo, Yongqiang Guo, Zhe Fu, Ying He, Panpan Huang, Jiashi Li, Wenfeng Liang, Xiaodong Liu, Xin Liu, Yiyuan Liu, Yuxuan Liu, Shanghao Lu, Xuan Lu, Xiaotao Nie, Tian Pei, Junjie Qiu, Hui Qu, Zehui Ren, Zhangli Sha, Xuecheng Su, Xiaowen Sun, Yixuan Tan, Minghui Tang, Shiyu Wang, Yaohui Wang, Yongji Wang, Ziwei Xie, Yiliang Xiong, Yanhong Xu, Shengfeng Ye, Shuiping Yu, Yukun Zha, Liyue Zhang, Haowei Zhang, Mingchuan Zhang, Wentao Zhang, Yichao Zhang, Chenggang Zhao, Yao Zhao, Shangyan Zhou, Shunfeng Zhou, and Yuheng Zou. Fire-flyer ai-hpc: A cost-effective software-hardware co-design for deep learning. *Computing Research Repository*, arXiv:2408.14158, 2024. URL `https://arxiv.org/abs/2408.14158`.

Zachary Ankner, Mansheej Paul, Brandon Cui, Jonathan D. Chang, and Prithviraj Ammanabrolu. Critique-out-loud reward models. *Computing Research Repository*, arXiv:2408.11791, 2024. URL `https://arxiv.org/abs/2408.11791`.

Negar Arabzadeh, Siqing Huo, Nikhil Mehta, Qingyun Wu, Chi Wang, Ahmed Hassan Awadallah, Charles L. A. Clarke, and Julia Kiseleva. Assessing and verifying task utility in LLM-powered applications. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 21868–21888, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.1219. URL `https://aclanthology.org/2024.emnlp-main.1219/`.

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, Nicholas Joseph, Saurav Kadavath, Jackson Kernion, Tom Conerly, Sheer El-Showk, Nelson Elhage, Zac Hatfield-Dodds, Danny Hernandez, Tristan Hume, Scott Johnston, Shauna Kravec, Liane Lovitt, Neel Nanda, Catherine Olsson, Dario Amodei, Tom Brown, Jack Clark, Sam McCandlish, Chris Olah, Ben Mann, and Jared Kaplan. Training a helpful and harmless assistant with reinforcement learning from human feedback. *Computing Research Repository*, arXiv:2204.05862, 2022a. URL `https://arxiv.org/abs/2204.05862`.

Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, Carol Chen, Catherine Olsson, Christopher Olah, Danny Hernandez, Dawn Drain, Deep Ganguli, Dustin Li, Eli Tran-Johnson, Ethan Perez, Jamie Kerr, Jared Mueller, Jeffrey Ladish, Joshua Landau, Kamal Ndousse, Kamile Lukosuite, Liane Lovitt, Michael Sellitto, Nelson Elhage, Nicholas Schiefer, Noemi Mercado, Nova DasSarma, Robert Lasenby, Robin Larson, Sam Ringer, Scott Johnston, Shauna Kravec, Sheer El Showk, Stanislav Fort, Tamera Lanham, Timothy Telleen-Lawton, Tom Conerly, Tom Henighan, Tristan Hume, Samuel R. Bowman, Zac Hatfield-Dodds, Ben Mann, Dario Amodei, Nicholas Joseph, Sam McCandlish, Tom Brown, and Jared Kaplan. Constitutional ai: Harmlessness from ai feedback. *Computing Research Repository*, arXiv:2212.08073, 2022b. URL `https://arxiv.org/abs/2212.08073`.

Bowen Baker, Joost Huizinga, Leo Gao, Zehao Dou, Melody Y. Guan, Aleksander Madry, Wojciech Zaremba, Jakub Pachocki, and David Farhi. Monitoring reasoning models for misbehavior and the risks of promoting obfuscation. *OpenAI Publication*, 2025. URL `https://cdn.openai.com/pdf/34f2ada6-870f-4c26-9790-fd8def56387f/CoT_Monitoring.pdf`.

11

7 Conclusion and Future Work

We introduced Self-Principled Critique Tuning (SPCT), a method that enhances the scalability of inference time for generalist reward modeling. With rule-based online RL, SPCT enables adaptive generation of principles and critiques, significantly boosting reward quality and inference-time scalability for GRMs in diverse domains. Empirical results demonstrate that DeepSeek-GRM surpass baseline methods and a few strong public RMs, and show notable improvement through inference-time scaling, particularly with the guidance of the meta RM. Future directions could include integrating GRMs into online RL pipelines as versatile interfaces of reward systems, exploring inference-time co-scaling with policy models, or serving as robust offline evaluators for foundation models.

Ethics Statement

Our proposed method, Self-Principled Critique Tuning (SPCT), aims to enhance inference-time scalability of generative reward models (GRMs) for general domains. While this advancement promotes accuracy and consistency in reward modeling, several ethical implications might warrant explicit consideration.

Firstly, even though through our empirical analysis that DeepSeek-GRM shows less biases on different domains, the automated generation of principles and critiques can inadvertently perpetuate or amplify biases when the training data is toxic. We argue that further investigation in the meta RM and other bias mitigation strategies should be prioritized to ensure equitable outcomes. Also, our approach does not aim to diminish human oversight. Instead, we advocate maintaining human-in-the-loop frameworks, and developing reliable proxy methods, like SPCT, to scale human oversight more efficiently and effectively.

Secondly, expanded applicability of the inference-time scalable GRMs across diverse domains might raise concerns regarding transparency, accountability, etc. We demonstrate model capabilities in Section 5.2 and limitations in Appendix B, and open-source the model under public supervision, which is essential for maintaining trust and ensuring responsible deployment of the artifact.

Finally, robust validation and ongoing vigilance across varied RM benchmarks and practical scenarios remain crucial. Ethical use of DeepSeek-GRM necessitates proactive management of risks and continuous evaluation against biases, requiring efforts in researches about RM evaluation.

参考文献

Andrei Alexandru, Antonia Calvi, Henry Broomfield, Jackson Golden, Kyle Dai, Mathias Leys, Maurice Burger, Max Bartolo, Roman Engeler, Sashank Pisupati, Toby Drane, and Young Sun Park. Atla selene mini: A general purpose evaluation model. *Computing Research Repository*, arXiv:2501.17195, 2025. URL https://arxiv.org/abs/2501.17195.

Wei An, Xiao Bi, Guanting Chen, Shanhuang Chen, Chengqi Deng, Honghui Ding, Kai Dong, Qiushi Du, Wenjun Gao, Kang Guan, Jianzhong Guo, Yongqiang Guo, Zhe Fu, Ying He, Panpan Huang, Jiashi Li, Wenfeng Liang, Xiaodong Liu, Xin Liu, Yiyuan Liu, Yuxuan

Bradley Brown, Jordan Juravsky, Ryan Ehrlich, Ronald Clark, Quoc V. Le, Christopher Ré, and Azalia Mirhoseini. Large language monkeys: Scaling inference compute with repeated sampling. *Computing Research Repository*, arXiv:2407.21787, 2024. URL https://arxiv.org/abs/2407.21787.

Zheng Cai, Maosong Cao, Haojiong Chen, Kai Chen, Keyu Chen, Xin Chen, Xun Chen, Zehui Chen, Zhi Chen, Pei Chu, Xiaoyi Dong, Haodong Duan, Qi Fan, Zhaoye Fei, Yang Gao, Jiaye Ge, Chenya Gu, Yuzhe Gu, Tao Gui, Aijia Guo, Qipeng Guo, Conghui He, Yingfan Hu, Ting Huang, Tao Jiang, Penglong Jiao, Zhenjiang Jin, Zhikai Lei, Jiaxing Li, Jingwen Li, Linyang Li, Shuaibin Li, Wei Li, Yining Li, Hongwei Liu, Jiangning Liu, Jiawei Hong, Kaiwen Liu, Kuikun Liu, Xiaoran Liu, Chengqi Lv, Haijun Lv, Kai Lv, Li Ma, Runyuan Ma, Zerun Ma, Wenchang Ning, Linke Ouyang, Jiantao Qiu, Yuan Qu, Fukai Shang, Yunfan Shao, Demin Song, Zifan Song, Zhihao Sui, Peng Sun, Yu Sun, Huanze Tang, Bin Wang, Guoteng Wang, Jiaqi Wang, Jiayu Wang, Rui Wang, Yudong Wang, Ziyi Wang, Xingjian Wei, Qizhen Weng, Fan Wu, Yingtong Xiong, Chao Xu, Ruiliang Xu, Hang Yan, Yirong Yan, Xiaogui Yang, Haochen Ye, Huaiyuan Ying, Jia Yu, Jing Yu, Yuhang Zang, Chuyu Zhang, Li Zhang, Pan Zhang, Peng Zhang, Ruijie Zhang, Shuo Zhang, Songyang Zhang, Wenjian Zhang, Wenwei Zhang, Xingcheng Zhang, Xinyue Zhang, Hui Zhao, Qian Zhao, Xiaomeng Zhao, Fengzhe Zhou, Zaida Zhou, Jingming Zhuo, Yicheng Zou, Xipeng Qiu, Yu Qiao, and Dahua Lin. Internlm2 technical report. *Computing Research Repository*, arXiv:2403.17297, 2024. URL https://arxiv.org/abs/2403.17297.

Maosong Cao, Alexander Lam, Haodong Duan, Hongwei Liu, Songyang Zhang, and Kai Chen. Compassjudger-1: All-in-one judge model helps model evaluation and evolution. *Computing Research Repository*, arXiv:2410.16256, 2024. URL https://arxiv.org/abs/2410.16256.

Bei Chen, Fengji Zhang, Anh Nguyen, Daoguang Zan, Zeqi Lin, Jian-Guang Lou, and Weizhu Chen. Codet: Code generation with generated tests. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=ktrw68Cmu9c.

Yinlam Chow, Guy Tennenholtz, Izzeddin Gur, Vincent Zhuang, Bo Dai, Aviral Kumar, Rishabh Agarwal, Sridhar Thiagarajan, Craig Boutilier, and Aleksandra Faust. Inference-aware fine-tuning for best-of-n sampling in large language models. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=77gQUdQhE7.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems. *Computing Research Repository*, arXiv:2110.14168, 2021. URL https://arxiv.org/abs/2110.14168.

Ganqu Cui, Lifan Yuan, Ning Ding, Guanming Yao, Bingxiang He, Wei Zhu, Yuan Ni, Guotong Xie, Ruobing Xie, Yankai Lin, Zhiyuan Liu, and Maosong Sun. ULTRAFEEDBACK: Boosting language models with scaled AI feedback. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235

Liu, Shanghao Lu, Xuan Lu, Xiaotao Nie, Tian Pei, Junjie Qiu, Hui Qu, Zehui Ren, Zhangli Sha, Xuecheng Su, Xiaowen Sun, Yixuan Tan, Minghui Tang, Shiyu Wang, Yaohui Wang, Yongji Wang, Ziwei Xie, Yiliang Xiong, Yanhong Xu, Shengfeng Ye, Shuiping Yu, Yukun Zha, Liyue Zhang, Haowei Zhang, Mingchuan Zhang, Wentao Zhang, Yichao Zhang, Chenggang Zhao, Yao Zhao, Shangyan Zhou, Shunfeng Zhou, and Yuheng Zou. Fire-flyer ai-hpc: A cost-effective software-hardware co-design for deep learning. *Computing Research Repository*, arXiv:2408.14158, 2024. URL https://arxiv.org/abs/2408.14158.

Zachary Ankner, Mansheej Paul, Brandon Cui, Jonathan D. Chang, and Prithviraj Ammanabrolu. Critique-out-loud reward models. *Computing Research Repository*, arXiv:2408.11791, 2024. URL https://arxiv.org/abs/2408.11791.

Negar Arabzadeh, Siqing Huo, Nikhil Mehta, Qingyun Wu, Chi Wang, Ahmed Hassan Awadallah, Charles L. A. Clarke, and Julia Kiseleva. Assessing and verifying task utility in LLM-powered applications. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 21868–21888, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.1219. URL https://aclanthology.org/2024.emnlp-main.1219/.

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, Nicholas Joseph, Saurav Kadavath, Jackson Kernion, Tom Conerly, Sheer El-Showk, Nelson Elhage, Zac Hatfield-Dodds, Danny Hernandez, Tristan Hume, Scott Johnston, Shauna Kravec, Liane Lovitt, Neel Nanda, Catherine Olsson, Dario Amodei, Tom Brown, Jack Clark, Sam McCandlish, Chris Olah, Ben Mann, and Jared Kaplan. Training a helpful and harmless assistant with reinforcement learning from human feedback. *Computing Research Repository*, arXiv:2204.05862, 2022a. URL https://arxiv.org/abs/2204.05862.

Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, Carol Chen, Catherine Olsson, Christopher Olah, Danny Hernandez, Dawn Drain, Deep Ganguli, Dustin Li, Eli Tran-Johnson, Ethan Perez, Jamie Kerr, Jared Mueller, Jeffrey Ladish, Joshua Landau, Kamal Ndousse, Kamile Lukosuite, Liane Lovitt, Michael Sellitto, Nelson Elhage, Nicholas Schiefer, Noemi Mercado, Nova DasSarma, Robert Lasenby, Robin Larson, Sam Ringer, Scott Johnston, Shauna Kravec, Sheer El Showk, Stanislav Fort, Tamera Lanham, Timothy Telleen-Lawton, Tom Conerly, Tom Henighan, Tristan Hume, Samuel R. Bowman, Zac Hatfield-Dodds, Ben Mann, Dario Amodei, Nicholas Joseph, Sam McCandlish, Tom Brown, and Jared Kaplan. Constitutional ai: Harmlessness from ai feedback. *Computing Research Repository*, arXiv:2212.08073, 2022b. URL https://arxiv.org/abs/2212.08073.

Bowen Baker, Joost Huizinga, Leo Gao, Zehao Dou, Melody Y. Guan, Aleksander Madry, Wojciech Zaremba, Jakub Pachocki, and David Farhi. Monitoring reasoning models for misbehavior and the risks of promoting obfuscation. *OpenAI Publication*, 2025. URL https://cdn.openai.com/pdf/34f2ada6-870f-4c26-9790-fd8def56387f/CoT_Monitoring.pdf.

Bradley Brown, Jordan Juravsky, Ryan Ehrlich, Ronald Clark, Quoc V. Le, Christopher Ré, and Azalia Mirhoseini. Large language monkeys: Scaling inference compute with

Preprint. Under review.

of *Proceedings of Machine Learning Research*, pp. 9722–9744. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/cui24f.html.

DeepSeek-AI. Deepseek-v2: A strong, economical, and efficient mixture-of-experts language model. *Computing Research Repository*, arXiv:2405.04434, 2024a. URL https://arxiv.org/abs/2405.04434.

DeepSeek-AI. Deepseek-v3 technical report. *Computing Research Repository*, arXiv:2412.19437, 2024b. URL https://arxiv.org/abs/2412.19437.

DeepSeek-AI. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *Computing Research Repository*, arXiv:2501.12948, 2025. URL https://arxiv.org/abs/2501.12948.

Jan-Philipp Fränken, Eric Zelikman, Rafael Rafailov, Kanishk Gandhi, Tobias Gerstenberg, and Noah Goodman. Self-supervised alignment with mutual information: Learning to follow principles without preference labels. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=UvbpbEhGaw.

Evan Frick, Tianle Li, Connor Chen, Wei-Lin Chiang, Anastasios Nikolas Angelopoulos, Jiantao Jiao, Banghua Zhu, Joseph E. Gonzalez, and Ion Stoica. How to evaluate reward models for RLHF. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=cbttLtO94Q.

Leo Gao, John Schulman, and Jacob Hilton. Scaling laws for reward model overoptimization. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (eds.), *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 10835–10866. PMLR, 23–29 Jul 2023. URL https://proceedings.mlr.press/v202/gao23h.html.

Amelia Glaese, Nat McAleese, Maja Trębacz, John Aslanides, Vlad Firoiu, Timo Ewalds, Maribeth Rauh, Laura Weidinger, Martin Chadwick, Phoebe Thacker, Lucy Campbell-Gillingham, Jonathan Uesato, Po-Sen Huang, Ramona Comanescu, Fan Yang, Abigail See, Sumanth Dathathri, Rory Greig, Charlie Chen, Doug Fritz, Jaume Sanchez Elias, Richard Green, Soňa Mokrá, Nicholas Fernando, Boxi Wu, Rachel Foley, Susannah Young, Iason Gabriel, William Isaac, John Mellor, Demis Hassabis, Koray Kavukcuoglu, Lisa Anne Hendricks, and Geoffrey Irving. Improving alignment of dialogue agents via targeted human judgements. *Computing Research Repository*, arXiv:2209.14375, 2022. URL https://arxiv.org/abs/2209.14375.

Fang Guo, Wenyu Li, Honglei Zhuang, Yun Luo, Yafu Li, Le Yan, Qi Zhu, and Yue Zhang. Mcranker: Generating diverse criteria on-the-fly to improve pointwise llm rankers. In *Proceedings of the Eighteenth ACM International Conference on Web Search and Data Mining*, WSDM '25, pp. 944–953, New York, NY, USA, 2025. Association for Computing Machinery. ISBN 9798400713293. doi: 10.1145/3701551.3703583. URL https://doi.org/10.1145/3701551.3703583.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the MATH repeated sampling. *Computing Research Repository*, arXiv:2407.21787, 2024. URL https://arxiv.org/abs/2407.21787.

Zheng Cai, Maosong Cao, Haojiong Chen, Kai Chen, Keyu Chen, Xin Chen, Xun Chen, Zehui Chen, Zhi Chen, Pei Chu, Xiaoyi Dong, Haodong Duan, Qi Fan, Zhaoye Fei, Yang Gao, Jiaye Ge, Chenya Gu, Yuzhe Gu, Tao Gui, Aijia Guo, Qipeng Guo, Conghui He, Yingfan Hu, Ting Huang, Tao Jiang, Penglong Jiao, Zhenjiang Jin, Zhikai Lei, Jiaxing Li, Jingwen Li, Linyang Li, Shuaibin Li, Wei Li, Yining Li, Hongwei Liu, Jiangning Liu, Jiawei Hong, Kaiwen Liu, Kuikun Liu, Xiaoran Liu, Chengqi Lv, Haijun Lv, Kai Lv, Li Ma, Runyuan Ma, Zerun Ma, Wenchang Ning, Linke Ouyang, Jiantao Qiu, Yuan Qu, Fukai Shang, Yunfan Shao, Demin Song, Zifan Song, Zhihao Sui, Peng Sun, Yu Sun, Huanze Tang, Bin Wang, Guoteng Wang, Jiaqi Wang, Jiayu Wang, Rui Wang, Yudong Wang, Ziyi Wang, Xingjian Wei, Qizhen Weng, Fan Wu, Yingtong Xiong, Chao Xu, Ruiliang Xu, Hang Yan, Yirong Yan, Xiaogui Yang, Haochen Ye, Huaiyuan Ying, Jia Yu, Jing Yu, Yuhang Zang, Chuyu Zhang, Li Zhang, Pan Zhang, Peng Zhang, Ruijie Zhang, Shuo Zhang, Songyang Zhang, Wenjian Zhang, Wenwei Zhang, Xingcheng Zhang, Xinyue Zhang, Hui Zhao, Qian Zhao, Xiaomeng Zhao, Fengzhe Zhou, Zaida Zhou, Jingming Zhuo, Yicheng Zou, Xipeng Qiu, Yu Qiao, and Dahua Lin. Internlm2 technical report. *Computing Research Repository*, arXiv:2403.17297, 2024. URL https://arxiv.org/abs/2403.17297.

Maosong Cao, Alexander Lam, Haodong Duan, Hongwei Liu, Songyang Zhang, and Kai Chen. Compassjudger-1: All-in-one judge model helps model evaluation and evolution. *Computing Research Repository*, arXiv:2410.16256, 2024. URL https://arxiv.org/abs/2410.16256.

Bei Chen, Fengji Zhang, Anh Nguyen, Daoguang Zan, Zeqi Lin, Jian-Guang Lou, and Weizhu Chen. Codet: Code generation with generated tests. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=ktrw68Cmu9c.

Yinlam Chow, Guy Tennenholtz, Izzeddin Gur, Vincent Zhuang, Bo Dai, Aviral Kumar, Rishabh Agarwal, Sridhar Thiagarajan, Craig Boutilier, and Aleksandra Faust. Inference-aware fine-tuning for best-of-n sampling in large language models. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=77gQUdQhE7.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems. *Computing Research Repository*, arXiv:2110.14168, 2021. URL https://arxiv.org/abs/2110.14168.

Ganqu Cui, Lifan Yuan, Ning Ding, Guanming Yao, Bingxiang He, Wei Zhu, Yuan Ni, Guotong Xie, Ruobing Xie, Yankai Lin, Zhiyuan Liu, and Maosong Sun. ULTRAFEEDBACK: Boosting language models with scaled AI feedback. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 9722–9744. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/cui24f.html.

Preprint. Under review.

dataset. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*, 2021. URL https://openreview.net/forum?id=7Bywt2mQsCe.

Dongfu Jiang, Xiang Ren, and Bill Yuchen Lin. LLM-blender: Ensembling large language models with pairwise ranking and generative fusion. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki (eds.), *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 14165–14178, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-long.792. URL https://aclanthology.org/2023.acl-long.792/.

Carlos E Jimenez, John Yang, Alexander Wettig, Shunyu Yao, Kexin Pei, Ofir Press, and Karthik R Narasimhan. SWE-bench: Can language models resolve real-world github issues? In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=VTF8yNQM66.

Ryo Kamoi, Sarkar Snigdha Sarathi Das, Renze Lou, Jihyun Janice Ahn, Yilun Zhao, Xiaoxin Lu, Nan Zhang, Yusen Zhang, Haoran Ranran Zhang, Sujeeth Reddy Vummanthala, Salika Dave, Shaobo Qin, Arman Cohan, Wenpeng Yin, and Rui Zhang. Evaluating LLMs at detecting errors in LLM responses. In *First Conference on Language Modeling*, 2024. URL https://openreview.net/forum?id=dnwRScljXr.

M. G. Kendall and B. Babington Smith. On the method of paired comparisons. *Biometrika*, 31(3/4):324–345, 1940. ISSN 00063444. URL http://www.jstor.org/stable/2332613.

Seungone Kim, Juyoung Suk, Shayne Longpre, Bill Yuchen Lin, Jamin Shin, Sean Welleck, Graham Neubig, Moontae Lee, Kyungjae Lee, and Minjoon Seo. Prometheus 2: An open source language model specialized in evaluating other language models. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 4334–4353, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.248. URL https://aclanthology.org/2024.emnlp-main.248/.

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

Junlong Li, Shichao Sun, Weizhe Yuan, Run-Ze Fan, hai zhao, and Pengfei Liu. Generative judge for evaluating alignment. In *The Twelfth International Conference on Learning Representations*, 2024a. URL https://openreview.net/forum?id=gtkFw6sZGS.

Lei Li, Yekun Chai, Shuohuan Wang, Yu Sun, Hao Tian, Ningyu Zhang, and Hua Wu. Tool-augmented reward modeling. In *The Twelfth International Conference on Learning Representations*, 2024b. URL https://openreview.net/forum?id=d94x0gWTUX.

Shalev Lifshitz, Sheila A. McIlraith, and Yilun Du. Multi-agent verification: Scaling test-time compute with multiple verifiers. *Computing Research Repository*, arXiv:2502.20379, 2025. URL https://arxiv.org/abs/2502.20379.

14

DeepSeek-AI. Deepseek-v2: A strong, economical, and efficient mixture-of-experts language model. *Computing Research Repository*, arXiv:2405.04434, 2024a. URL https://arxiv.org/abs/2405.04434.

DeepSeek-AI. Deepseek-v3 technical report. *Computing Research Repository*, arXiv:2412.19437, 2024b. URL https://arxiv.org/abs/2412.19437.

DeepSeek-AI. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *Computing Research Repository*, arXiv:2501.12948, 2025. URL https://arxiv.org/abs/2501.12948.

Jan-Philipp Fränken, Eric Zelikman, Rafael Rafailov, Kanishk Gandhi, Tobias Gerstenberg, and Noah Goodman. Self-supervised alignment with mutual information: Learning to follow principles without preference labels. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=UvbpbEhGaw.

Evan Frick, Tianle Li, Connor Chen, Wei-Lin Chiang, Anastasios Nikolas Angelopoulos, Jiantao Jiao, Banghua Zhu, Joseph E. Gonzalez, and Ion Stoica. How to evaluate reward models for RLHF. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=cbttLtO94Q.

Leo Gao, John Schulman, and Jacob Hilton. Scaling laws for reward model overoptimization. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (eds.), *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 10835–10866. PMLR, 23–29 Jul 2023. URL https://proceedings.mlr.press/v202/gao23h.html.

Amelia Glaese, Nat McAleese, Maja Trębacz, John Aslanides, Vlad Firoiu, Timo Ewalds, Maribeth Rauh, Laura Weidinger, Martin Chadwick, Phoebe Thacker, Lucy Campbell-Gillingham, Jonathan Uesato, Po-Sen Huang, Ramona Comanescu, Fan Yang, Abigail See, Sumanth Dathathri, Rory Greig, Charlie Chen, Doug Fritz, Jaume Sanchez Elias, Richard Green, Soňa Mokrá, Nicholas Fernando, Boxi Wu, Rachel Foley, Susannah Young, Iason Gabriel, William Isaac, John Mellor, Demis Hassabis, Koray Kavukcuoglu, Lisa Anne Hendricks, and Geoffrey Irving. Improving alignment of dialogue agents via targeted human judgements. *Computing Research Repository*, arXiv:2209.14375, 2022. URL https://arxiv.org/abs/2209.14375.

Fang Guo, Wenyu Li, Honglei Zhuang, Yun Luo, Yafu Li, Le Yan, Qi Zhu, and Yue Zhang. Mcranker: Generating diverse criteria on-the-fly to improve pointwise llm rankers. In *Proceedings of the Eighteenth ACM International Conference on Web Search and Data Mining*, WSDM '25, pp. 944–953, New York, NY, USA, 2025. Association for Computing Machinery. ISBN 9798400713293. doi: 10.1145/3701551.3703583. URL https://doi.org/10.1145/3701551.3703583.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the MATH dataset. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*, 2021. URL https://openreview.net/forum?id=7Bywt2mQsCe.

Preprint. Under review.

Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=v8L0pN6EOi.

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. *Computing Research Repository*, arXiv:2410.18451, 2024. URL https://arxiv.org/abs/2410.18451.

Yantao Liu, Zijun Yao, Rui Min, Yixin Cao, Lei Hou, and Juanzi Li. Pairjudge rm: Perform best-of-n sampling with knockout tournament. *Computing Research Repository*, arXiv:2501.13007, 2025. URL https://arxiv.org/abs/2501.13007.

Dakota Mahan, Duy Van Phung, Rafael Rafailov, Chase Blagden, Nathan Lile, Louis Castricato, Jan-Philipp Fränken, Chelsea Finn, and Alon Albalak. Generative reward models. *Computing Research Repository*, arXiv:2410.12832, 2024. URL https://arxiv.org/abs/2410.12832.

Tong Mu, Alec Helyar, Johannes Heidecke, Joshua Achiam, Andrea Vallone, Ian D Kivlichan, Molly Lin, Alex Beutel, John Schulman, and Lilian Weng. Rule based rewards for language model safety. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=QVtwpT5Dmg.

OpenAI. Openai o1 system card. *Computing Research Repository*, arXiv:2412.16720, 2024. URL https://arxiv.org/abs/2412.16720.

OpenAI. Deep research system card. *OpenAI Publication*, 2025a. URL https://cdn.openai.com/deep-research-system-card.pdf.

OpenAI. Openai gpt-4.5 system card. *OpenAI Publication*, 2025b. URL https://cdn.openai.com/gpt-4-5-system-card-2272025.pdf.

OpenAI. Openai o3-mini system card. *OpenAI Publication*, 2025c. URL https://cdn.openai.com/o3-mini-system-card-feb10.pdf.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, Red Hook, NY, USA, 2022. Curran Associates Inc. ISBN 9781713871088.

Junsoo Park, Seungyeon Jwa, Ren Meiying, Daeyoung Kim, and Sanghyuk Choi. OffsetBias: Leveraging debiased data for tuning evaluators. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 1043–1067, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-emnlp.57. URL https://aclanthology.org/2024.findings-emnlp.57/.

15

Dongfu Jiang, Xiang Ren, and Bill Yuchen Lin. LLM-blender: Ensembling large language models with pairwise ranking and generative fusion. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki (eds.), *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 14165–14178, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-long.792. URL https://aclanthology.org/2023.acl-long.792/.

Carlos E Jimenez, John Yang, Alexander Wettig, Shunyu Yao, Kexin Pei, Ofir Press, and Karthik R Narasimhan. SWE-bench: Can language models resolve real-world github issues? In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=VTF8yNQM66.

Ryo Kamoi, Sarkar Snigdha Sarathi Das, Renze Lou, Jihyun Janice Ahn, Yilun Zhao, Xiaoxin Lu, Nan Zhang, Yusen Zhang, Haoran Ranran Zhang, Sujeeth Reddy Vummanthala, Salika Dave, Shaobo Qin, Arman Cohan, Wenpeng Yin, and Rui Zhang. Evaluating LLMs at detecting errors in LLM responses. In *First Conference on Language Modeling*, 2024. URL https://openreview.net/forum?id=dnwRScljXr.

M. G. Kendall and B. Babington Smith. On the method of paired comparisons. *Biometrika*, 31(3/4):324–345, 1940. ISSN 00063444. URL http://www.jstor.org/stable/2332613.

Seungone Kim, Juyoung Suk, Shayne Longpre, Bill Yuchen Lin, Jamin Shin, Sean Welleck, Graham Neubig, Moontae Lee, Kyungjae Lee, and Minjoon Seo. Prometheus 2: An open source language model specialized in evaluating other language models. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 4334–4353, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.248. URL https://aclanthology.org/2024.emnlp-main.248/.

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

Junlong Li, Shichao Sun, Weizhe Yuan, Run-Ze Fan, hai zhao, and Pengfei Liu. Generative judge for evaluating alignment. In *The Twelfth International Conference on Learning Representations*, 2024a. URL https://openreview.net/forum?id=gtkFw6sZGS.

Lei Li, Yekun Chai, Shuohuan Wang, Yu Sun, Hao Tian, Ningyu Zhang, and Hua Wu. Tool-augmented reward modeling. In *The Twelfth International Conference on Learning Representations*, 2024b. URL https://openreview.net/forum?id=d94x0gWTUX.

Shalev Lifshitz, Sheila A. McIlraith, and Yilun Du. Multi-agent verification: Scaling test-time compute with multiple verifiers. *Computing Research Repository*, arXiv:2502.20379, 2025. URL https://arxiv.org/abs/2502.20379.

Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by

Preprint. Under review.

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. *Computing Research Repository*, arXiv:2502.19328, 2025. URL https://arxiv.org/abs/2502.19328.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=HPuSIXJaa9.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *Computing Research Repository*, arXiv:2402.0330, 2024. URL https://arxiv.org/abs/2402.03300.

Mrinank Sharma, Meg Tong, Jesse Mu, Jerry Wei, Jorrit Kruthoff, Scott Goodfriend, Euan Ong, Alwin Peng, Raj Agarwal, Cem Anil, Amanda Askell, Nathan Bailey, Joe Benton, Emma Bluemke, Samuel R. Bowman, Eric Christiansen, Hoagy Cunningham, Andy Dau, Anjali Gopal, Rob Gilson, Logan Graham, Logan Howard, Nimit Kalra, Taesung Lee, Kevin Lin, Peter Lofgren, Francesco Mosconi, Clare O'Hara, Catherine Olsson, Linda Petrini, Samir Rajani, Nikhil Saxena, Alex Silverstein, Tanya Singh, Theodore Sumers, Leonard Tang, Kevin K. Troy, Constantin Weisser, Ruiqi Zhong, Giulio Zhou, Jan Leike, Jared Kaplan, and Ethan Perez. Constitutional classifiers: Defending against universal jailbreaks across thousands of hours of red teaming. *Computing Research Repository*, arXiv:2501.18837, 2025. URL https://arxiv.org/abs/2501.18837.

Charlie Victor Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. Scaling LLM test-time compute optimally can be more effective than scaling parameters for reasoning. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=4FWAwZtd2n.

Nisan Stiennon, Long Ouyang, Jeff Wu, Daniel M. Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul Christiano. Learning to summarize from human feedback. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, Red Hook, NY, USA, 2020. Curran Associates Inc. ISBN 9781713829546.

Zhiqing Sun, Yikang Shen, Qinhong Zhou, Hongxin Zhang, Zhenfang Chen, David Daniel Cox, Yiming Yang, and Chuang Gan. Principle-driven self-alignment of language models from scratch with minimal human supervision. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=p40XRfBX96.

Gemma Team. Gemma 2: Improving open language models at a practical size. *Computing Research Repository*, arXiv:2408.0011, 2024. URL https://arxiv.org/abs/2408.00118.

Hemish Veeraboina. Aime problem set 1983-2024, 2023. URL https://www.kaggle.com/datasets/hemishveeraboina/aime-problem-set-1983-2024.

Tu Vu, Kalpesh Krishna, Salaheddin Alzubi, Chris Tar, Manaal Faruqui, and Yun-Hsuan Sung. Foundational autoraters: Taming large language models for better automatic evaluation. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Proceedings of*

16

step. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=v8L0pN6EOi.

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. *Computing Research Repository*, arXiv:2410.18451, 2024. URL https://arxiv.org/abs/2410.18451.

Yantao Liu, Zijun Yao, Rui Min, Yixin Cao, Lei Hou, and Juanzi Li. Pairjudge rm: Perform best-of-n sampling with knockout tournament. *Computing Research Repository*, arXiv:2501.13007, 2025. URL https://arxiv.org/abs/2501.13007.

Dakota Mahan, Duy Van Phung, Rafael Rafailov, Chase Blagden, Nathan Lile, Louis Castricato, Jan-Philipp Fränken, Chelsea Finn, and Alon Albalak. Generative reward models. *Computing Research Repository*, arXiv:2410.12832, 2024. URL https://arxiv.org/abs/2410.12832.

Tong Mu, Alec Helyar, Johannes Heidecke, Joshua Achiam, Andrea Vallone, Ian D Kivlichan, Molly Lin, Alex Beutel, John Schulman, and Lilian Weng. Rule based rewards for language model safety. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=QVtwpT5Dmg.

OpenAI. Openai o1 system card. *Computing Research Repository*, arXiv:2412.16720, 2024. URL https://arxiv.org/abs/2412.16720.

OpenAI. Deep research system card. *OpenAI Publication*, 2025a. URL https://cdn.openai.com/deep-research-system-card.pdf.

OpenAI. Openai gpt-4.5 system card. *OpenAI Publication*, 2025b. URL https://cdn.openai.com/gpt-4-5-system-card-2272025.pdf.

OpenAI. Openai o3-mini system card. *OpenAI Publication*, 2025c. URL https://cdn.openai.com/o3-mini-system-card-feb10.pdf.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, Red Hook, NY, USA, 2022. Curran Associates Inc. ISBN 9781713871088.

Junsoo Park, Seungyeon Jwa, Ren Meiying, Daeyoung Kim, and Sanghyuk Choi. OffsetBias: Leveraging debiased data for tuning evaluators. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 1043–1067, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-emnlp.57. URL https://aclanthology.org/2024.findings-emnlp.57/.

Preprint. Under review.

the 2024 Conference on Empirical Methods in Natural Language Processing, pp. 17086–17105, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.949. URL https://aclanthology.org/2024.emnlp-main.949/.

Haoxiang Wang, Wei Xiong, Tengyang Xie, Han Zhao, and Tong Zhang. Interpretable preferences via multi-objective reward modeling and mixture-of-experts. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 10582–10592, Miami, Florida, USA, November 2024a. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-emnlp.620. URL https://aclanthology.org/2024.findings-emnlp.620/.

Peiyi Wang, Lei Li, Zhihong Shao, Runxin Xu, Damai Dai, Yifei Li, Deli Chen, Yu Wu, and Zhifang Sui. Math-shepherd: Verify and reinforce LLMs step-by-step without human annotations. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 9426–9439, Bangkok, Thailand, August 2024b. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.510. URL https://aclanthology.org/2024.acl-long.510/.

Tianlu Wang, Ilia Kulikov, Olga Golovneva, Ping Yu, Weizhe Yuan, Jane Dwivedi-Yu, Richard Yuanzhe Pang, Maryam Fazel-Zarandi, Jason Weston, and Xian Li. Self-taught evaluators. *Computing Research Repository*, arXiv:2408.02666, 2024c. URL https://arxiv.org/abs/2408.02666.

Zhilin Wang, Yi Dong, Olivier Delalleau, Jiaqi Zeng, Gerald Shen, Daniel Egert, Jimmy J. Zhang, Makesh Narsimhan Sreedhar, and Oleksii Kuchaiev. Helpsteer 2: Open-source dataset for training top-performing reward models. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2024d. URL https://openreview.net/forum?id=PvVKUFhaNy.

Zhilin Wang, Alexander Bukharin, Olivier Delalleau, Daniel Egert, Gerald Shen, Jiaqi Zeng, Oleksii Kuchaiev, and Yi Dong. Helpsteer2-preference: Complementing ratings with preferences. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=MnfHxPP5gs.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed H. Chi, Quoc V Le, and Denny Zhou. Chain of thought prompting elicits reasoning in large language models. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho (eds.), *Advances in Neural Information Processing Systems*, 2022. URL https://openreview.net/forum?id=_VjQlMeSB_J.

Genta Indra Winata, David Anugraha, Lucky Susanto, Garry Kuwanto, and Derry Tanti Wijaya. Metametrics: Calibrating metrics for generation tasks using human preferences. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=slO3xTt4CG.

Tianhao Wu, Weizhe Yuan, Olga Golovneva, Jing Xu, Yuandong Tian, Jiantao Jiao, Jason Weston, and Sainbayar Sukhbaatar. Meta-rewarding language models: Self-improving

Preprint. Under review.

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. *Computing Research Repository*, arXiv:2502.19328, 2025. URL https://arxiv.org/abs/2502.19328.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=HPuSIXJaa9.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *Computing Research Repository*, arXiv:2402.0330, 2024. URL https://arxiv.org/abs/2402.03300.

Mrinank Sharma, Meg Tong, Jesse Mu, Jerry Wei, Jorrit Kruthoff, Scott Goodfriend, Euan Ong, Alwin Peng, Raj Agarwal, Cem Anil, Amanda Askell, Nathan Bailey, Joe Benton, Emma Bluemke, Samuel R. Bowman, Eric Christiansen, Hoagy Cunningham, Andy Dau, Anjali Gopal, Rob Gilson, Logan Graham, Logan Howard, Nimit Kalra, Taesung Lee, Kevin Lin, Peter Lofgren, Francesco Mosconi, Clare O'Hara, Catherine Olsson, Linda Petrini, Samir Rajani, Nikhil Saxena, Alex Silverstein, Tanya Singh, Theodore Sumers, Leonard Tang, Kevin K. Troy, Constantin Weisser, Ruiqi Zhong, Giulio Zhou, Jan Leike, Jared Kaplan, and Ethan Perez. Constitutional classifiers: Defending against universal jailbreaks across thousands of hours of red teaming. *Computing Research Repository*, arXiv:2501.18837, 2025. URL https://arxiv.org/abs/2501.18837.

Charlie Victor Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. Scaling LLM test-time compute optimally can be more effective than scaling parameters for reasoning. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=4FWAwZtd2n.

Nisan Stiennon, Long Ouyang, Jeff Wu, Daniel M. Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul Christiano. Learning to summarize from human feedback. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, Red Hook, NY, USA, 2020. Curran Associates Inc. ISBN 9781713829546.

Zhiqing Sun, Yikang Shen, Qinhong Zhou, Hongxin Zhang, Zhenfang Chen, David Daniel Cox, Yiming Yang, and Chuang Gan. Principle-driven self-alignment of language models from scratch with minimal human supervision. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=p40XRfBX96.

Gemma Team. Gemma 2: Improving open language models at a practical size. *Computing Research Repository*, arXiv:2408.0011, 2024. URL https://arxiv.org/abs/2408.00118.

Hemish Veeraboina. Aime problem set 1983-2024, 2023. URL https://www.kaggle.com/datasets/hemishveeraboina/aime-problem-set-1983-2024.

Tu Vu, Kalpesh Krishna, Salaheddin Alzubi, Chris Tar, Manaal Faruqui, and Yun-Hsuan Sung. Foundational autoraters: Taming large language models for better automatic evaluation. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Proceedings of*

alignment with llm-as-a-meta-judge. *Computing Research Repository*, arXiv:2407.19594, 2024. URL https://arxiv.org/abs/2407.19594.

Yangzhen Wu, Zhiqing Sun, Shanda Li, Sean Welleck, and Yiming Yang. Inference scaling laws: An empirical analysis of compute-optimal inference for LLM problem-solving. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=VNckp7JEHn.

Tianbao Xie, Danyang Zhang, Jixuan Chen, Xiaochuan Li, Siheng Zhao, Ruisheng Cao, Toh Jing Hua, Zhoujun Cheng, Dongchan Shin, Fangyu Lei, Yitao Liu, Yiheng Xu, Shuyan Zhou, Silvio Savarese, Caiming Xiong, Victor Zhong, and Tao Yu. OSWorld: Benchmarking multimodal agents for open-ended tasks in real computer environments. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2024. URL https://openreview.net/forum?id=tN61DTr4Ed.

Shunyu Yao, Howard Chen, John Yang, and Karthik R Narasimhan. Webshop: Towards scalable real-world web interaction with grounded language agents. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho (eds.), *Advances in Neural Information Processing Systems*, 2022. URL https://openreview.net/forum?id=R9KnuFlvnU.

Zihuiwen Ye, Fraser Greenlee-Scott, Max Bartolo, Phil Blunsom, Jon Ander Campos, and Matthias Gallé. Improving reward models with synthetic critiques. *Computing Research Repository*, arXiv:2405.20850, 2024. URL https://arxiv.org/abs/2405.20850.

Ziyi Ye, Xiangsheng Li, Qiuchi Li, Qingyao Ai, Yujia Zhou, Wei Shen, Dong Yan, and Yiqun LIU. Learning LLM-as-a-judge for preference alignment. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=HZVIQE1MsJ.

Jiachen Yu, Shaoning Sun, Xiaohui Hu, Jiaxu Yan, Kaidong Yu, and Xuelong Li. Improve llm-as-a-judge ability as a general ability. *Computing Research Repository*, arXiv:2502.11689, 2025a. URL https://arxiv.org/abs/2502.11689.

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

Weizhe Yuan, Richard Yuanzhe Pang, Kyunghyun Cho, Xian Li, Sainbayar Sukhbaatar, Jing Xu, and Jason E Weston. Self-rewarding language models. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 57905–57923. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/yuan24d.html.

Lunjun Zhang, Arian Hosseini, Hritik Bansal, Mehran Kazemi, Aviral Kumar, and Rishabh Agarwal. Generative verifiers: Reward modeling as next-token prediction. In *The Thirteenth International Conference on Learning Representations*, 2025a. URL https://openreview.net/forum?id=Ccwp4tFEtE.

18

the 2024 Conference on Empirical Methods in Natural Language Processing, pp. 17086–17105, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.949. URL https://aclanthology.org/2024.emnlp-main.949/.

Haoxiang Wang, Wei Xiong, Tengyang Xie, Han Zhao, and Tong Zhang. Interpretable preferences via multi-objective reward modeling and mixture-of-experts. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 10582–10592, Miami, Florida, USA, November 2024a. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-emnlp.620. URL https://aclanthology.org/2024.findings-emnlp.620/.

Peiyi Wang, Lei Li, Zhihong Shao, Runxin Xu, Damai Dai, Yifei Li, Deli Chen, Yu Wu, and Zhifang Sui. Math-shepherd: Verify and reinforce LLMs step-by-step without human annotations. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 9426–9439, Bangkok, Thailand, August 2024b. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.510. URL https://aclanthology.org/2024.acl-long.510/.

Tianlu Wang, Ilia Kulikov, Olga Golovneva, Ping Yu, Weizhe Yuan, Jane Dwivedi-Yu, Richard Yuanzhe Pang, Maryam Fazel-Zarandi, Jason Weston, and Xian Li. Self-taught evaluators. *Computing Research Repository*, arXiv:2408.02666, 2024c. URL https://arxiv.org/abs/2408.02666.

Zhilin Wang, Yi Dong, Olivier Delalleau, Jiaqi Zeng, Gerald Shen, Daniel Egert, Jimmy J. Zhang, Makesh Narsimhan Sreedhar, and Oleksii Kuchaiev. Helpsteer 2: Open-source dataset for training top-performing reward models. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2024d. URL https://openreview.net/forum?id=PvVKUFhaNy.

Zhilin Wang, Alexander Bukharin, Olivier Delalleau, Daniel Egert, Gerald Shen, Jiaqi Zeng, Oleksii Kuchaiev, and Yi Dong. Helpsteer2-preference: Complementing ratings with preferences. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=MnfHxPP5gs.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed H. Chi, Quoc V Le, and Denny Zhou. Chain of thought prompting elicits reasoning in large language models. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho (eds.), *Advances in Neural Information Processing Systems*, 2022. URL https://openreview.net/forum?id=_VjQlMeSB_J.

Genta Indra Winata, David Anugraha, Lucky Susanto, Garry Kuwanto, and Derry Tanti Wijaya. Metametrics: Calibrating metrics for generation tasks using human preferences. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=slO3xTt4CG.

Tianhao Wu, Weizhe Yuan, Olga Golovneva, Jing Xu, Yuandong Tian, Jiantao Jiao, Jason Weston, and Sainbayar Sukhbaatar. Meta-rewarding language models: Self-improving

Preprint. Under review.

Zhenru Zhang, Chujie Zheng, Yangzhen Wu, Beichen Zhang, Runji Lin, Bowen Yu, Dayiheng Liu, Jingren Zhou, and Junyang Lin. The lessons of developing process reward models in mathematical reasoning. *Computing Research Repository*, arXiv:2501.07301, 2025b. URL https://arxiv.org/abs/2501.07301.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging llm-as-a-judge with mt-bench and chatbot arena. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, Red Hook, NY, USA, 2023. Curran Associates Inc.

Enyu Zhou, Guodong Zheng, Binghai Wang, Zhiheng Xi, Shihan Dou, Rong Bao, Wei Shen, Limao Xiong, Jessica Fan, Yurong Mou, Rui Zheng, Tao Gui, Qi Zhang, and Xuanjing Huang. RMB: Comprehensively benchmarking reward models in LLM alignment. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=kmgrlG9TR0.

Terry Yue Zhuo, Vu Minh Chien, Jenny Chim, Han Hu, Wenhao Yu, Ratnadira Widyasari, Imam Nur Bani Yusuf, Haolan Zhan, Junda He, Indraneil Paul, Simon Brunner, Chen GONG, James Hoang, Armel Randy Zebaze, Xiaoheng Hong, Wen-Ding Li, Jean Kaddour, Ming Xu, Zhihan Zhang, Prateek Yadav, Naman Jain, Alex Gu, Zhoujun Cheng, Jiawei Liu, Qian Liu, Zijian Wang, David Lo, Binyuan Hui, Niklas Muennighoff, Daniel Fried, Xiaoning Du, Harm de Vries, and Leandro Von Werra. Bigcodebench: Benchmarking code generation with diverse function calls and complex instructions. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=YrycTjllL0.

19

alignment with llm-as-a-meta-judge. *Computing Research Repository*, arXiv:2407.19594, 2024. URL https://arxiv.org/abs/2407.19594.

Yangzhen Wu, Zhiqing Sun, Shanda Li, Sean Welleck, and Yiming Yang. Inference scaling laws: An empirical analysis of compute-optimal inference for LLM problem-solving. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=VNckp7JEHn.

Tianbao Xie, Danyang Zhang, Jixuan Chen, Xiaochuan Li, Siheng Zhao, Ruisheng Cao, Toh Jing Hua, Zhoujun Cheng, Dongchan Shin, Fangyu Lei, Yitao Liu, Yiheng Xu, Shuyan Zhou, Silvio Savarese, Caiming Xiong, Victor Zhong, and Tao Yu. OSWorld: Benchmarking multimodal agents for open-ended tasks in real computer environments. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2024. URL https://openreview.net/forum?id=tN61DTr4Ed.

Shunyu Yao, Howard Chen, John Yang, and Karthik R Narasimhan. Webshop: Towards scalable real-world web interaction with grounded language agents. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho (eds.), *Advances in Neural Information Processing Systems*, 2022. URL https://openreview.net/forum?id=R9KnuFlvnU.

Zihuiwen Ye, Fraser Greenlee-Scott, Max Bartolo, Phil Blunsom, Jon Ander Campos, and Matthias Gallé. Improving reward models with synthetic critiques. *Computing Research Repository*, arXiv:2405.20850, 2024. URL https://arxiv.org/abs/2405.20850.

Ziyi Ye, Xiangsheng Li, Qiuchi Li, Qingyao Ai, Yujia Zhou, Wei Shen, Dong Yan, and Yiqun LIU. Learning LLM-as-a-judge for preference alignment. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=HZVIQE1MsJ.

Jiachen Yu, Shaoning Sun, Xiaohui Hu, Jiaxu Yan, Kaidong Yu, and Xuelong Li. Improve llm-as-a-judge ability as a general ability. *Computing Research Repository*, arXiv:2502.11689, 2025a. URL https://arxiv.org/abs/2502.11689.

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

Weizhe Yuan, Richard Yuanzhe Pang, Kyunghyun Cho, Xian Li, Sainbayar Sukhbaatar, Jing Xu, and Jason E Weston. Self-rewarding language models. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 57905–57923. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/yuan24d.html.

Lunjun Zhang, Arian Hosseini, Hritik Bansal, Mehran Kazemi, Aviral Kumar, and Rishabh Agarwal. Generative verifiers: Reward modeling as next-token prediction. In *The Thirteenth International Conference on Learning Representations*, 2025a. URL https://openreview.net/forum?id=Ccwp4tFEtE.

目录

1 Introduction 1

2 Preliminaries 3
- 2.1 Comparisons of Different RM approaches 3
- 2.2 Boosting Reward Quality with Principles 4

3 Self-Principled Critique Tuning (SPCT) 4
- 3.1 Unpinning Principles from Understanding to Generation 5
- 3.2 Rule-Based Reinforcement Learning 5

4 Inference-Time Scaling with SPCT 6

5 Results on Reward Modeling Benchmarks 7
- 5.1 Experiment Settings 7
- 5.2 Results and Analysis 8

6 Related Work 9

7 Conclusion and Future Work 10

A Additional Related Work 22

B Limitations and Future Directions 22

C Implementation Details 23
- C.1 Comparisons of Different RM Approaches 23
- C.2 Model Training 25
- C.3 Baseline Implementation 25

D Experiment Details 26
- D.1 Hyper-Parameters 26
- D.2 Benchmarks 26
- D.3 Detailed Results 27

E Additional Experiments 27
- E.1 Input Flexibility of the Pointwise GRM Approach 27
  - E.1.1 Generating Rewards for Many Responses 28
  - E.1.2 Generating Rewards for Single Responses 29
  - E.1.3 Generating Rewards with Reference 30
- E.2 Transferability of Generated Principles 30
- E.3 Generalization beyond Training Data 30
- E.4 Response Length Analysis for Rule-Based RL 30

Zhenru Zhang, Chujie Zheng, Yangzhen Wu, Beichen Zhang, Runji Lin, Bowen Yu, Dayiheng Liu, Jingren Zhou, and Junyang Lin. The lessons of developing process reward models in mathematical reasoning. *Computing Research Repository*, arXiv:2501.07301, 2025b. URL https://arxiv.org/abs/2501.07301.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging llm-as-a-judge with mt-bench and chatbot arena. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, Red Hook, NY, USA, 2023. Curran Associates Inc.

Enyu Zhou, Guodong Zheng, Binghai Wang, Zhiheng Xi, Shihan Dou, Rong Bao, Wei Shen, Limao Xiong, Jessica Fan, Yurong Mou, Rui Zheng, Tao Gui, Qi Zhang, and Xuanjing Huang. RMB: Comprehensively benchmarking reward models in LLM alignment. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=kmgrlG9TR0.

Terry Yue Zhuo, Vu Minh Chien, Jenny Chim, Han Hu, Wenhao Yu, Ratnadira Widyasari, Imam Nur Bani Yusuf, Haolan Zhan, Junda He, Indraneil Paul, Simon Brunner, Chen GONG, James Hoang, Armel Randy Zebaze, Xiaoheng Hong, Wen-Ding Li, Jean Kaddour, Ming Xu, Zhihan Zhang, Prateek Yadav, Naman Jain, Alex Gu, Zhoujun Cheng, Jiawei Liu, Qian Liu, Zijian Wang, David Lo, Binyuan Hui, Niklas Muennighoff, Daniel Fried, Xiaoning Du, Harm de Vries, and Leandro Von Werra. Bigcodebench: Benchmarking code generation with diverse function calls and complex instructions. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=YrycTjllL0.

Preprint. Under review.

F Qualitative Analysis **31**
F.1 Case Study 31
F.2 Failure Mode Analysis 41

G Prompt Templates **42**

21

目录

1 Introduction 1

2 Preliminaries 4

2.1 Comparisons of Different RM approaches 4

2.2 Boosting Reward Quality with Principles 5

3 Self-Principled Critique Tuning (SPCT) 5

3.1 Unpinning Principles from Understanding to Generation 5

3.2 Rule-Based Reinforcement Learning 6

4 Inference-Time Scaling with SPCT 7

5 Results on Reward Modeling Benchmarks 8

5.1 Experiment Settings 8

5.2 Results and Analysis 9

6 Related Work 11

7 Conclusion and Future Work 12

A Additional Related Work 24

B Limitations and Future Directions 25

C Implementation Details 25

C.1 Comparisons of Different RM Approaches 25

C.2 Model Training 27

C.3 Baseline Implementation 28

D Experiment Details 29

D.1 Hyper-Parameters 29

D.2 Benchmarks 30

D.3 Detailed Results 32

E Additional Experiments 32

E.1 Input Flexibility of the Pointwise GRM Approach 32

E.1.1 Generating Rewards for Many Responses 32

E.1.2 Generating Rewards for Single Responses 33

E.1.3 Generating Rewards with Reference 33

E.2 Transferability of Generated Principles 33

E.3 Generalization beyond Training Data 33

E.4 Response Length Analysis for Rule-Based RL 34

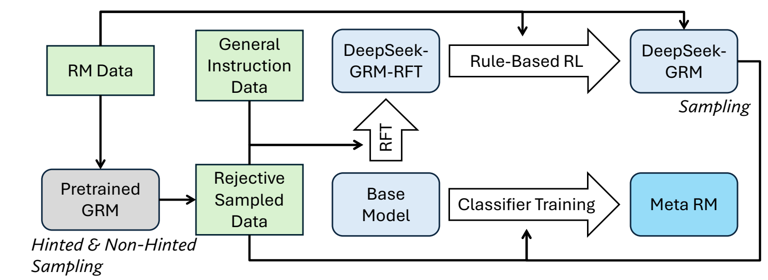


图 5: 在SPCT管道中对DeepSeek-GRM-RFT、DeepSeek-GRM和Meta RM的推导进行说明。

A Additional Related Work

宪法式人工智能 宪法式人工智能（Constitutional AI）作为一种可扩展的替代方案，已从传统的人类反馈强化学习中脱颖而出 (Ouyang et al., 2022)，其目标是通过一组指导原则或“宪法”来使语言模型与人类价值观对齐 (Bai et al., 2022b; Sun et al., 2023)，用AI生成的反馈 (Fränken et al., 2024) 或基于这些手工原则的分类器 (Sharma et al., 2025) 替代人工批评。类似地，基于规则的方法如Sparrow (Glaese et al., 2022) 和基于规则的奖励（RBR）(Mu et al., 2024) 将明确的自然语言规则纳入训练循环，以应对特定领域的安全问题。尽管这些方法有效，但它们依赖于静态、手动编写的宪法原则，这些原则在范围上有限制，可能存在偏差，并且缺乏灵活性。这激发了对自动化生成或改进原则的兴趣，这也与我们在这项工作中的目标一致。

标量奖励模型 为大型语言模型（LLMs）提出的标量奖励建模最早被用作人类反馈的代理模型 (Stiennon et al., 2020; Gao et al., 2023)。最近的研究集中在Bradley-Terry建模 (Kendall & Smith, 1940) 和其他回归方法上，以提高标量奖励模型的表达能力 (Cai et al., 2024; Wang et al., 2024d;a; Liu et al., 2024; Wang et al., 2025)。与这些结果奖励模型不同，过程奖励模型被提出作为推理问题（例如数学等）的步骤验证器，提供丰富的反馈 (Cobbe et al., 2021; Wang et al., 2024b; Zhang et al., 2025b)，展示了标量奖励模型在需要广泛推理和知识的形式领域中的可行性。标量奖励模型在简单性和计算效率方面表现出色，但由于表达能力有限，在处理多样化的输入类型时难以泛化，且在推理过程中难以细化奖励信号。

半标量奖励模型 半标量奖励模型旨在通过文本中间表示来丰富标量奖励信号 (Ye et al., 2024; Ankner et al., 2024)。因此，一些研究 (Yu et al., 2025b) 提出通过提高生成批评的质量来最终改善奖励生成。一些研究使用标记概率代替标量头进行奖励提取 (Mahan et al., 2024; Zhang et al., 2025a)。这些研究表明，半标量奖励模型在基于采样和投票的推理时间扩展方面面临挑战，导致性能提升有限。半标量方法在效率和有效性之间进行了权衡，介于标量奖励模型和生成式奖励模型之间。

B Limitations and Future Directions

局限性 尽管SPCT显著提升了GRMs的性能和推理时间的可扩展性，并在通用领域超越了（半）标量RMs，但它仍然面临一些局限性。 (1) 生成式RMs的效率本质上远远落后于相同规模的标量RMs，这限制了其在在线RL管道中的大规模应用。然而，由于我们在推理时采用了并行采样，例如使用八次采样等合理数量的情况下，奖励生成的延迟不会显著增加。围绕LLMs高效生成的研究以及RMs应用的创新可能会缓解这一问题。 (2) 在特定领域（如可验证任务）中，DeepSeek-GRM仍然落后于标量模型。这可能是因为标量RMs能

F Qualitative Analysis **34**
F.1 Case Study . . . 34
F.2 Failure Mode Analysis . . . 45

G Prompt Templates **46**

够捕捉推理查询和响应中的隐藏特征，而GRMs需要更强的推理能力来全面检查响应。然而，标量RMs存在严重的偏差和可扩展性问题。对于GRMs，我们发现基于参考的奖励生成（见附录 E.1.3）和长视野推理（见附录 D.3）可以减轻这一局限。(3) 由于逐点GRM方法的通用性，DeepSeek-GRM有可能不仅作为结果RM，还可以作为过程RM。虽然本文在这方面尚未深入探讨，但在Reward Bench的推理子集中的表现，主要包含MATH-prm数据 (Lightman et al., 2024)，可以在一定程度上支持这一应用的潜力。

未来方向 基于SPCT或DeepSeek-GRM模型，还有几个有前景的研究方向值得进一步探索。(1) 之前的工作 (Li et al., 2024b)研究了RMs的工具集成，这种方法也可以用于DeepSeek-GRM增强。**借助代码解释器和搜索引擎接口等工具**，生成的批评意见在需要严格程序或广泛知识的任务中会更加准确，并且可以避免GRMs在数值计算、模式匹配等相关原则上的失败情况。 (2) **生成原则和批评的范式可以分解为不同的阶段**，即原则可以提前为每个查询及其待评估的响应生成并存储，然后通过GRMs、规则或其他代理方法生成批评意见。原则生成作为后续批评的接口，这可能会提高当前GRMs在RL管道整合中的效率。 (3) DeepSeek-GRM有可能**用于LLM的离线评估**。由于每个原则反映了一个标准，我们可以从所有数据点中找出某个特定LLM不如其他LLM的地方，作为一种可解释的弱点协议。 (4) DeepSeek-GRM可能**从长视野推理中受益**。然而，这将进一步影响其效率。这些方向应在未来的工作中进行研究。

C Implementation Details

C.1 Comparisons of Different RM Approaches

奖励生成范式 经典的奖励模型采用 **(a) 标量** 方法来生成奖励 ($\mathcal{R}$)，该方法为给定的查询和响应分配标量值。标量方法进一步扩展为 **(b) 半标量** 方法，该方法除了生成标量值外还生成文本。而 **(c) 生成式** 方法仅生成文本形式的奖励。

$$\mathcal{R} = \begin{cases} S & \text{(Scalar)} \\ (S, C) & \text{(Semi-Scalar)} \\ C & \text{(Generative)} \end{cases} \sim r_\theta\left(x, \{y_i\}_{i=1}^n\right), \tag{7}$$

其中 x 是查询，y_i 是第 i 个响应，r_θ 是由参数 θ 定义的奖励函数，$S \in \mathbb{R}^m, m \leq n$ 是标量奖励，而 C 是批评值。

评分模式 我们区分了奖励的两种主要评分方法：逐点评分和成对评分。**（i）逐点评分**方法为每个响应分配一个单独的分数：

$$\{S_i\}_{i=1}^n = f_{\text{point}}\left(\mathcal{R}, \{y_i\}_{i=1}^n\right), \quad \mathcal{R} \sim r_\theta\left(x, \{y_i\}_{i=1}^n\right), S_i \in \mathbb{R}, \tag{8}$$

其中 $f_{\text{point}}(\cdot, \cdot)$ 是一个分裂函数。相比之下，**(ii) 成对** 方法可以被视为一种最佳-n 方法，从所有候选者中选择一个最优响应：

$$\hat{y} = f_{\text{pair}}(\mathcal{R}, \{y_i\}_{i=1}^n), \quad \mathcal{R} \sim r_\theta\left(x, \{y_i\}_{i=1}^n\right), \hat{y} \in \{y_i\}_{i=1}^n, \tag{9}$$

其中 $f_{\text{pair}}(\cdot, \cdot)$ 是一个选择函数，且在大多数情况下 $n = 2$。尽管成对的方法可以扩展到 $n > 2$ 的情况，但它无法应用于单个响应评分 ($n = 1$)。

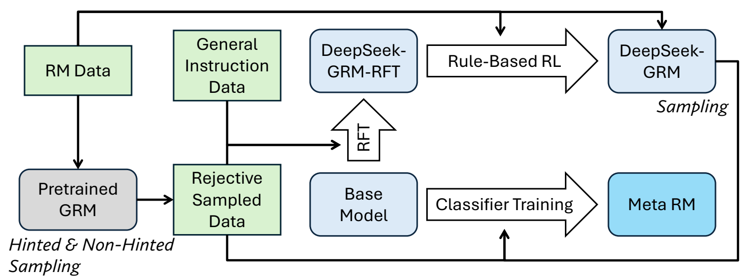


图 5: Illustration of the derivation of DeepSeek-GRM-RFT, DeepSeek-GRM, and Meta RM in the SPCT pipeline.

A Additional Related Work

Constitutional AI Constitutional AI has emerged as a scalable alternative to traditional reinforcement learning from human feedback (Ouyang et al., 2022), aiming to align language models with human values through a set of guiding principles or "constitutions" (Bai et al., 2022b; Sun et al., 2023), replacing human critiques with AI-generated feedback (Fränken et al., 2024) or classifiers (Sharma et al., 2025) based on these handicraft principles. Similarly, rule-based approaches like Sparrow (Glaese et al., 2022) and Rule-Based Rewards (RBR) (Mu et al., 2024) incorporate explicit natural language rules into the training loop for specific domains like safety. Although effective, these methods rely on static, manually written constitutions that are limited in scope, potentially biased, and inflexible. This has motivated interests in automating the generation or refinement of principles, which aligns with our target in this work.

Scalar Reward Models Scalar reward modeling for LLMs are proposed the earliest to serve as a proxy model for human feedback (Stiennon et al., 2020; Gao et al., 2023). Recent studies focus on Bradley-Terry modeling (Kendall & Smith, 1940) and other regression approaches for better expressiveness for scalar reward models (Cai et al., 2024; Wang et al., 2024d;a; Liu et al., 2024; Wang et al., 2025) of general preference. In contrast to these outcome reward models, process reward models are proposed as step verifiers for reasoning problems, e.g., math, etc., with rich feedbacks (Cobbe et al., 2021; Wang et al., 2024b; Zhang et al., 2025b), demonstrating the feasibility of scalar RMs in a formal domain with extensive reasoning and knowledge. Scalar RM excels in simplicity and is computationally efficient, but suffers from limited expressivity and struggles to generalize across diverse input types or refine reward signals at inference time.

Semi-Scalar Reward Models Semi-scalar reward models aim to enrich scalar reward signals through textual intermediate representations. (Ye et al., 2024; Ankner et al., 2024) Consequently, works (Yu et al., 2025b) proposed to enhance the quality of generated critiques to eventually improve reward generation. Some studies use the token probability to substitute the scalar head for reward extraction (Mahan et al., 2024; Zhang et al., 2025a). These works show that semi-scalar RMs face challenges in inference-time scaling based on sampling and voting, resulting in limited performance improvement. The semi-scalar approach trades off between scalar RMs and GRMs in terms of both efficiency and effectiveness.

代表性方法 图 2 展示了三种奖励生成范式（标量、半标量、生成式）如何与两种评分模式（逐点、成对）相结合。具体来说，Bradley-Terry 模型 (Kendall & Smith, 1940) ((*a*)+(*i*)) 使用成对偏好数据进行训练，并逐点输出标量奖励。

$$\{S_i\}_{i=1}^n = f_{\text{point}}\left(\mathcal{R}, \{y_i\}_{i=1}^n\right) = \mathcal{S} \in \mathbb{R}^n. \tag{10}$$

PairRM (Jiang et al., 2023) ((*a*)+(*ii*)) 通过标量奖励的符号比较了一对响应。

$$\hat{y} = f_{\text{pair}}\left(\mathcal{R}, \{y_i\}_{i=1}^n\right) = y_{\lfloor \frac{1}{2}(3-\text{sgn}(\mathcal{S}))\rfloor}, \quad n = 2, \mathcal{S} \in \mathbb{R}. \tag{11}$$

上述的标量方法由于在奖励生成的多样性不足，因此在推理阶段的扩展能力有限。 CLoud (Ankner et al., 2024) ((*b*)+(*i*)) 基于预先生成的批评意见为每个响应生成标量奖励，类似于 Equation 10。 LLM-as-a-Judge (Zheng et al., 2023) ((*c*)+(*ii*)) 以文本方式判断成对响应之间的偏好顺序，

$$\hat{y} = f_{\text{pair}}\left(\mathcal{R}, \{y_i\}_{i=1}^n\right) = y_{f_{\text{extract}}(C)}, \quad n = 2, \tag{12}$$

其中 $f_{\text{extract}}(\cdot)$ 从语言表示中提取最佳响应的索引。然而，这种方法默认忽略了成对响应之间的平局情况。根据 Zhang et al. (2025a)，表示偏好顺序的标记的生成概率可以被用作标量奖励 ((*b*)+(*ii*)): $\mathcal{S} = \text{TokenProb}(\hat{C}) = r_\theta(\hat{C}|x, \{y_i\}_{i=1}^n)$，其中 $\hat{C}$ 是与偏好顺序相关的预定义标记。然而，在没有额外约束的情况下，GRMs 可以为纯语言表示中的多个响应生成逐点奖励 ((*c*)+(*i*)):

$$\{S_i\}_{i=1}^n = f_{\text{point}}\left(\mathcal{R}, \{y_i\}_{i=1}^n\right) = f_{\text{extract}}(C), \tag{13}$$

其中 $f_{\text{extract}}(\cdot)$ 从生成结果中提取分配给每个响应的奖励。通常，这些奖励是离散的，在本工作中我们默认赋值为 $S_i \in \mathbb{N}, 1 \leq S_i \leq 10$。这种方法有望同时实现推理时的可扩展性和输入灵活性。

使用生成奖励进行投票 投票是RM中广泛采用的一种推理时间扩展方法。回顾第 2.1 节中的方法，我们展示了针对半标量和生成式RM的k个样本的投票结果。对于半标量RM (Ankner et al., 2024; Zhang et al., 2025a)，投票以取平均的方式进行：

$$S^* = \frac{1}{k}\sum_{i=1}^{k} S_i, \quad \{\mathcal{R} = (S_i, C_i)\}_{i=1}^k \sim r_\theta\left(x, \{y_i\}_{i=1}^n\right), \tag{14}$$

其中 S^* 是最终奖励。在实际应用中，标量值的方差有限，这可能会阻碍可扩展性。对于成对的生成式奖励模型（GRMs） (Mahan et al., 2024; Wang et al., 2024c)，投票是通过选择被识别为最佳且频率最高的响应来进行的，即多数投票：

$$\hat{y}^* = \arg\max_{y} \sum_{i=1}^{k} \mathbb{I}(y = \hat{y}_i), \quad \{\hat{y}_i = f_{\text{pair}}(C_i, \{y_i\}_{i=1}^n)\}_{i=1}^k \sim r_\theta\left(x, \{y_i\}_{i=1}^n\right), \tag{15}$$

其中 $\hat{y}^*$ 是最终预测的最佳响应，$f_{\text{pair}}(\cdot,\cdot)$ 是一个选择函数，$\hat{y}_i$ 是每个样本单独选出的最佳响应，$\mathbb{I}(\cdot)$ 是指示函数。虽然投票过程具有可扩展性，但由于每个样本不允许平局，多数投票的结果可能会有偏差，并且由于缺乏定量评分，可能无法区分响应之间的细微差别。

B Limitations and Future Directions

Limitations Though SPCT significantly leverages the performance and inference-time scalability of GRMs and surpasses (semi-)scalar RMs in general domains, it still faces a few limitations. (1) The efficiency of the generative RMs is largely lagging behind the scalar RMs at the same scale by nature, which inhibits its large-scale usage in online RL pipelines. However, since we adopt parallel sampling for inference-time scaling, the latency of reward generation with a reasonable amount of, e.g., eight samplings will not increase significantly. Further research around the efficient generation of LLMs and innovations in RM applications could alleviate the problem. (2) In specific domains such as verifiable tasks, DeepSeek-GRM still lags behind scalar models. This could be because the scalar RMs capture hidden features of reasoning queries and responses, while GRMs need stronger reasoning capabilities to examine responses thoroughly. However, scalar RMs suffer severe biases and scalability issues. For GRMs, we found that both reference-based reward generation (Appendix E.1.3 and long-horizon reasoning (Appendix D.3) could mitigate this limitation. (3) Due to the universality of the pointwise GRM approach, DeepSeek-GRM could potentially serve as a process RM in addition to the outcome RM. Though we have not explored much in this direction in the paper, the performance in the Reasoning subset of Reward Bench, which mainly comprises of MATH-prm data (Lightman et al., 2024), could partially support the potential of this application.

Future Directions There are also several promising directions for future research based on SPCT or DeepSeek-GRM models. (1) Tool incorporation of RMs is studied by previous work (Li et al., 2024b), and could also be used for DeepSeek-GRM augmentation. **With tools such as code interpreters and search engine interfaces**, the generated critiques could be more accurate for tasks that requires strict procedures or extensive knowledge, and the cases in which GRMs fail to follow principles related to numeric calculations, pattern matching, etc. could be avoided. (2) **The generation paradigm for principles and critiques could be decomposed** into separate stages, that is, the principles could be generated ahead of time for each query and the responses to be rated and stored, and then the critiques are generated with GRMs, rules, or other agentic approaches. The principle generation serves as an interface for the following critiques. This might improve the efficiency of current GRMs for the integration of RL pipelines. (3) The DeepSeek-GRM could be potentially **used in LLM offline evaluation**. Since each principle reflects a criteria, we can get criteria from all data points that a particular LLM is inferior than one another, as a interpretable protocol of the weaknesses of the particular LLM. (4) The DeepSeek-GRM might **be benefit from long-horizon reasoning**. However, this will further affect its efficiency. These directions should be studied in the future work.

C Implementation Details

C.1 Comparisons of Different RM Approaches

Reward Generation Paradigms Classic RMs adopt the **(a) scalar** approach to generate rewards ($\mathcal{R}$), which assigns scalar values to the given query and responses. The scalar approach is further extended to the **(b) semi-scalar** approach, which generates texts besides

Preprint. Under review.

C.2 Model Training

对于基于规则的在线强化学习，我们使用标准的GRPO设置 (Shao et al., 2024)。总体目标是

$$\mathcal{J}_{\text{GRPO}}(\theta) = \mathbb{E}[q \sim P(Q), \{o_i\}_{i=1}^{G} \sim \pi_{\theta_{old}}(O|q)] \frac{1}{G} \sum_{i=1}^{G} \frac{1}{|o_i|} \sum_{t=1}^{|o_i|}$$
$$\left\{ \min \left[\frac{\pi_\theta(o_{i,t}|q, o_{i,<t})}{\pi_{\theta_{old}}(o_{i,t}|q, o_{i,<t})} \hat{A}_{i,t}, \text{clip}\left(\frac{\pi_\theta(o_{i,t}|q, o_{i,<t})}{\pi_{\theta_{old}}(o_{i,t}|q, o_{i,<t})}, 1-\epsilon, 1+\epsilon \right) \hat{A}_{i,t} \right] - \beta \mathbb{D}_{KL}\left[\pi_\theta || \pi_{ref}\right] \right\}, \tag{16}$$

其中 $\hat{A}_{i,t} = \frac{r_i - \text{mean}(\mathbf{r})}{\text{std}(\mathbf{r})}$，$G$ 是组大小，β 是KL惩罚的系数，且 $q = (x, \{y_i\}_{i=1}^n)$ 带有提示。我们对超参数 $\beta \in \{0.00, 0.01, 0.04, 0.08\}$ 进行了网格搜索，并发现 $\beta = 0.08$ 是最稳定的配置。当KL系数过小时，GRM在基准测试中倾向于集中在少数子集上，例如Reward Bench基准中的Chat和RMB基准中的Harmlessness，并表现出对其他某些领域的偏向性。我们设置 $G = 4$ 以在效率和性能之间取得更好的平衡。

Stage	Time (h)
RFT	19.2
Rule-Based RL	15.6

表 5: 训练时间（小时）为DeepSeek-GRM-27B的 RFT 和 RL 阶段。

训练集包含1250K RFT数据，其中包括1070K通用指令数据和186K拒绝采样数据，以及237K RL数据。通用指令数据来自内部数据集。拒绝采样数据和RL数据来自相同的RM数据集，包含单个、成对和多个响应的偏好信息，这些数据由内部数据和开源数据集构建而成，包括MATH (Hendrycks et al., 2021)、UltraFeedback (Cui et al., 2024)、OffsetBias (Park et al., 2024)、Skywork-Reward-Preference-80K-v0.2 (Liu et al., 2024) 和 HelpSteer2-Preference (Wang et al., 2025) 的训练集。具体来说，由于UltraFeedback的质量问题，我们重新标注了其中一部分的偏好标签；我们在MATH上通过基于规则的真值匹配进行采样和过滤轨迹，生成成对的偏好数据；对于单个响应的评分，我们将正确响应的真值奖励设为1，错误响应设为0，并仅纳入可验证的问题。对于拒绝采样，我们使用DeepSeek-v2.5-0906生成带原则和批评的轨迹。采样时间 N_{RFT} 被设置为3。在HelpSteer2上的提示采样过程中，我们将原始数据集中标注的偏好强度作为提示加入。我们还移除了对DeepSeek-V2-Lite-Chat来说过于简单的样本，即根据公式Equation 4，所有生成的奖励在三次中都是正确的样本，从RL数据中删除。

DeepSeek-GRM模型和元RM的推导过程如图 5所示。所有DeepSeek-GRM模型都从预训练的LLM版本开始训练。对于元RM的训练，我们重用了RFT阶段的拒绝采样数据，并使用DeepSeek-GRM-27B进行拒绝采样（$N_{\text{RFT}} = 3$），以避免元RM引导投票中的潜在偏差 (Chow et al., 2025)。元RM训练的学习率为1×10^{-5}，批量大小为512。表 5展示了DeepSeek-GRM-27B的RFT和RL训练时间。基于Gemma-2-27B的模型在Fire-Flyer平台 (An et al., 2024)上使用128块A100 GPU进行训练。RFT阶段的学习率为5×10^{-6}，RL阶段的学习率为4×10^{-7}，RFT阶段的批量大小为1024，RL阶段的批量大小为512。两个阶段均训练900步。由于资源限制，超过27B的DeepSeek-GRM模型未经历基于规则的RL，仅使用50K拒绝采样数据进行训练。

C.3 Baseline Implementation

对于基线方法，我们重新实现了 **LLM-as-a-Judge** (Zheng et al., 2023), **DeepSeek-BTRM-27B** (Kendall & Smith, 1940), **CLoud-Gemma-2-27B** (Ankner et al., 2024), 和 **DeepSeek-PairRM-27B** (Jiang et al., 2023)，它们基于 Gemma-2-27B (Team, 2024) 并使用了与 DeepSeek-GRM兼容的所有训练数据和设置。

the scalar value. And the **(c) generative** approach only generates textual rewards.

$$\mathcal{R} = \begin{cases} \boldsymbol{S} & \text{(Scalar)} \\ (\boldsymbol{S}, \boldsymbol{C}) & \text{(Semi-Scalar)} \\ \boldsymbol{C} & \text{(Generative)} \end{cases} \sim r_\theta\left(x, \{y_i\}_{i=1}^n\right), \tag{7}$$

where x is the query, y_i is the i-th response, r_θ is the reward function parameterized by θ, $\boldsymbol{S} \in \mathbb{R}^m, m \leq n$ is the scalar reward, and $\boldsymbol{C}$ is the critique.

Scoring Patterns We distinguish two main scoring approaches for rewards: pointwise and pairwise. The **(i) pointwise** approach assigns an individual score to each response:

$$\{S_i\}_{i=1}^n = f_{\text{point}}\left(\mathcal{R}, \{y_i\}_{i=1}^n\right), \quad \mathcal{R} \sim r_\theta\left(x, \{y_i\}_{i=1}^n\right), S_i \in \mathbb{R}, \tag{8}$$

where $f_{\text{point}}(\cdot,\cdot)$ is a splitting function. In contrast, the **(ii) pairwise** approach can be viewed as a best-of-n method, selecting a single best response from all candidates:

$$\hat{y} = f_{\text{pair}}(\mathcal{R}, \{y_i\}_{i=1}^n), \quad \mathcal{R} \sim r_\theta\left(x, \{y_i\}_{i=1}^n\right), \hat{y} \in \{y_i\}_{i=1}^n, \tag{9}$$

where $f_{\text{pair}}(\cdot,\cdot)$ is a selection function and $n = 2$ in most cases. Though the pairwise approach could be extended to $n > 2$, it could not be applied to single response scoring ($n = 1$).

Representative Methods Figure 2 illustrates how the three reward generation paradigms (scalar, semi-scalar, generative) can be combined with the two scoring patterns (pointwise, pairwise). Specifically, Bradley-Terry model (Kendall & Smith, 1940) (*(a)+(i)*) is trained with pairwise preference data and outputs scalar rewards pointwisely

$$\{S_i\}_{i=1}^n = f_{\text{point}}\left(\mathcal{R}, \{y_i\}_{i=1}^n\right) = \boldsymbol{S} \in \mathbb{R}^n. \tag{10}$$

PairRM (Jiang et al., 2023) (*(a)+(ii)*) compares a pair of responses with the sign of the scalar reward

$$\hat{y} = f_{\text{pair}}\left(\mathcal{R}, \{y_i\}_{i=1}^n\right) = y_{\lfloor \frac{1}{2}(3-\text{sgn}(\mathcal{S})) \rfloor}, \quad n = 2, \mathcal{S} \in \mathbb{R}. \tag{11}$$

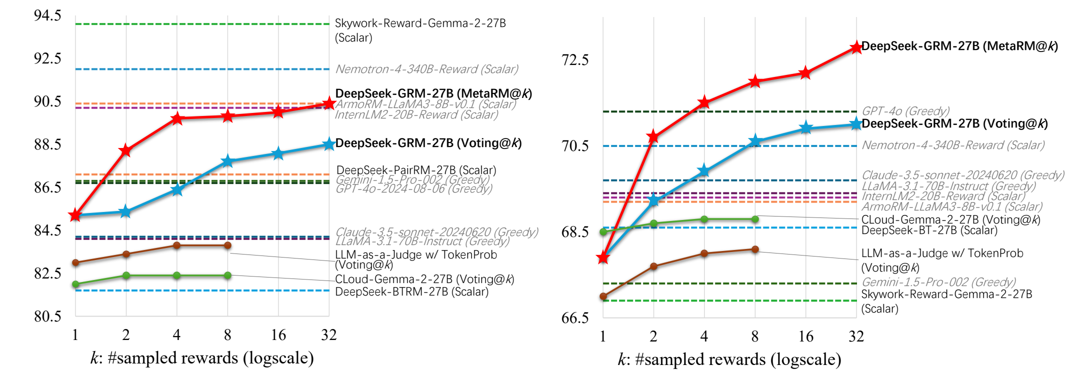
The scalar methods above could barely perform inference-time scaling due to the lack of diversity in reward generation. CLoud (Ankner et al., 2024) (*(b)+(i)*) generates scalar rewards for each response based on pre-generated critiques, similar to Equation 10. LLM-as-a-Judge (Zheng et al., 2023) (*(c)+(ii)*) judges the preference order between paired responses textually,

$$\hat{y} = f_{\text{pair}}\left(\mathcal{R}, \{y_i\}_{i=1}^n\right) = y_{f_{\text{extract}}(\boldsymbol{C})}, \quad n = 2, \tag{12}$$

where $f_{\text{extract}}(\cdot)$ extracts the index of best response from language representations. However, this approach defaults to neglect ties of the paired responses. Following Zhang et al. (2025a), the generation probability of the token that indicates the preference order could be used as the scalar reward (*(b)+(ii)*): $\mathcal{S} = \text{TokenProb}(\hat{C}) = r_\theta(\hat{C}|x, \{y_i\}_{i=1}^n)$, where $\hat{C}$ is a pre-defined token related to the preference order. However, without additional constraints, GRMs are able to generate pointwise rewards for multiple responses within pure language representation (*(c)+(i)*):

$$\{S_i\}_{i=1}^n = f_{\text{point}}\left(\mathcal{R}, \{y_i\}_{i=1}^n\right) = f_{\text{extract}}(\boldsymbol{C}), \tag{13}$$

Preprint. Under review.



(a) Results on the Reward Bench benchmark. (b) Results on all tested reward modeling benchmarks.

图 6: 推理时在不同奖励模型（RMs）和不同奖励建模基准上的扩展性能。非斜体字表示基于 Gemma-2-27B 的模型。

对于 **LLM-as-a-Judge**，我们使用与 DeepSeek-GRM-27B完全相同的训练配置，包括从 DeepSeek-V2.5-0906 中拒绝采样的数据进行 RFT 以及基于规则的在线强化学习（RL）。由于其评分模式，在 RL 阶段只能使用成对数据。对于 **CLoud-Gemma-2-27B**，我们也使用相同的提示模板从 DeepSeek-V2.5-0906 生成逐点批评。然而，由于在没有经过训练的价值头的情况下无法提取奖励，因此无法执行拒绝采样。我们通过与 DeepSeek-GRM-27B相同的一般指令数据以及采样的批评来微调 Gemma-2-27B，从而得到一个批评生成模型。具体来说，我们还微调了另一个带有价值头的 Gemma-2-27B 模型以生成奖励，而不是在批评模型上事后训练价值头。CLoud-Gemma-2-27B、**DeepSeek-BTRM-27B** 和 **DeepSeek-PairRM-27B** (Jiang et al., 2023) 的价值头训练使用了来自 DeepSeek-GRM-27B强化学习阶段的相同数据集，但不包括单响应评分数据。

D Experiment Details

D.1 Hyper-Parameters

对 于DeepSeek-GRM-27B、 DeepSeek-GRM-16B、 LLM-as-a-Judge和CLoud-Gemma-2-27B的推理时间缩放结果，每个模型的温度设置为0.5。而对于其他实验，所有模型的温度均设置为0。如果没有特别说明，在DeepSeek-GRM-27B的元RM引导投票中，默认值为$k_{\text{meta}} = \frac{1}{2}k$。在DeepSeek-R1上的推理过程中，温度设置为0.6。请注意，我们让DeepSeek-GRM在ReaLMistake基准测试中输出与其他基准测试相同的奖励范围，以评估单个响应。

D.2 Benchmarks

我们在不同领域的各种RM基准上评估了不同方法的性能：(1) **Reward Bench** (Lambert et al., 2024)，这是RM评估的一个通用基准，包含半自动收集的聊天、推理和安全性偏好数据，其中每个查询需要对两个响应进行排名；(2) **PPE** (Frick et al., 2025)，这是一个大规模基准，包含众包的偏好数据和可验证任务的正确性数据，每个查询有两个响应；(3) **RMB** (Zhou et al., 2025)，这是一个更全面的基准，包含各种类型的偏好数据，专注于帮助性和无害性，每个查询在成对子集和最佳-N (BoN) 子集中分别有两对或多对响应；(4) **ReaLMistake** (Kamoi where $f_{\text{extract}}(\cdot)$ extracts the rewards assigned to each response from generation results. Usually, the rewards are discrete, and in this work we assign $S_i \in \mathbb{N}, 1 \le S_i \le 10$ by default. This approach promisingly allows both inference-time scalability and input flexibility.

Voting with Generated Rewards Voting is a widely adopted method for inference-time scaling in RM. Recalling the approaches in Section 2.1, we demonstrate voting results of k samples for semi-scalar and generative RMs. For semi-scalar RMs (Ankner et al., 2024; Zhang et al., 2025a), voting is performed as averaging:

$$S^* = \frac{1}{k}\sum_{i=1}^{k} S_i, \quad \{\mathcal{R} = (S_i, C_i)\}_{i=1}^{k} \sim r_\theta\left(x, \{y_i\}_{i=1}^{n}\right), \tag{14}$$

where S^* is the final reward. In practice, the scalar value has limited variance which could hinder the scalability. For pairwise GRMs (Mahan et al., 2024; Wang et al., 2024c), voting is performed as selecting the response identified to be the best with the highest frequency, i.e. majority:

$$\hat{y}^* = \arg\max_{y} \sum_{i=1}^{k} \mathbb{I}(y = \hat{y}_i), \quad \{\hat{y}_i = f_{\text{pair}}(C_i, \{y_i\}_{i=1}^{n})\}_{i=1}^{k} \sim r_\theta\left(x, \{y_i\}_{i=1}^{n}\right), \tag{15}$$

where $\hat{y}^*$ is the final predicted best response, $f_{\text{pair}}(\cdot,\cdot)$ is a selection function, $\hat{y}_i$ is the individually selected best response of each sample, and $\mathbb{I}(\cdot)$ is the indicator function. Though the voting process is scalable, the majority voted result might be biased since ties is not allowed in each sample, and may not be able to tell apart subtle differences between responses due to the lack of quantitative scores.

C.2 Model Training

For the rule-based online RL, we use the standard GRPO setting (Shao et al., 2024). The overall objective is

$$\begin{aligned}
\mathcal{J}_{\text{GRPO}}(\theta) = \mathbb{E}[q \sim P(Q), \{o_i\}_{i=1}^{G} \sim \pi_{\theta_{old}}(O|q)] \frac{1}{G}\sum_{i=1}^{G}\frac{1}{|o_i|}\sum_{t=1}^{|o_i|} \\
\left\{\min\left[\frac{\pi_\theta(o_{i,t}|q, o_{i,<t})}{\pi_{\theta_{old}}(o_{i,t}|q, o_{i,<t})}\hat{A}_{i,t}, \text{clip}\left(\frac{\pi_\theta(o_{i,t}|q, o_{i,<t})}{\pi_{\theta_{old}}(o_{i,t}|q, o_{i,<t})}, 1-\epsilon, 1+\epsilon\right)\hat{A}_{i,t}\right] - \beta\mathbb{D}_{KL}\left[\pi_\theta || \pi_{ref}\right]\right\},
\end{aligned} \tag{16}$$

where $\hat{A}_{i,t} = \frac{r_i - \text{mean}(\mathbf{r})}{\text{std}(\mathbf{r})}$, G is the group size, β is the coefficient of KL penalty, and $q = (x, \{y_i\}_{i=1}^{n})$ with prompts. We performed grid search on hyper-parameter $\beta \in \{0.00, 0.01, 0.04, 0.08\}$ and found that $\beta = 0.08$ is the most stable configuration. And with too small KL coefficient, the GRM tends to collapse on a few subsets in benchmarks, e.g., Chat in the Reward Bench benchmark and Harmlessness in the RMB benchmark, and shows biases towards some other domains. We set $G = 4$ for a better trade-off between efficiency and performance.

The training set comprises of 1250K RFT data, including 1070K general instruction data and 186K rejective sampled data, and 237K RL data. General instruction data is from in-house datasets. Rejective sampled data and RL data are from the same RM datasets, containing the preference for single, paired,

Stage	Time (h)
RFT	19.2
Rule-Based RL	15.6

表 5: Training times of RFT and RL stages for DeepSeek-GRM-27B in hours.

Preprint. Under review.

Model	Reward Bench	PPE Preference	PPE Correctness	RMB	Overall
		Reported Results of Public Models			
Nemotron-4-340B-Reward	*92.0*	*59.3*	*60.8*	*69.9*	*70.5*
GPT-4o	86.7	67.1	57.6	73.8	71.3
		Results of Inference-Time Scaling (Voting@1)			
LLM-as-a-Judge	83.0	63.4	57.4	64.3	67.0
CLoud-Gemma-2-27B	*82.0*	*67.0*	*62.0*	*63.2*	*68.5*
DeepSeek-GRM-27B-RFT (Ours)	84.0	62.2	59.4	65.8	67.8
DeepSeek-GRM-27B (Ours)	85.2	62.4	59.5	64.4	67.9
		Results of Inference-Time Scaling (Voting@8)			
LLM-as-a-Judge	83.4	63.8	58.2	65.2	67.6 (+0.6)
LLM-as-a-Judge w/ **TokenProb**	83.8	64.6	58.8	65.2	68.1 (+1.1)
CLoud-Gemma-2-27B	*82.4*	***67.3***	*62.4*	*63.2*	*68.8* (+0.3)
DeepSeek-GRM-27B-RFT (Ours)	85.3	64.5	59.7	67.7	69.3 (+1.5)
DeepSeek-GRM-27B (Ours)	87.7	64.9	60.3	69.5	70.6 (+2.7)
DeepSeek-GRM-27B (MetaRM) (Ours)	89.8	66.4	63.0	68.8	72.0 (+4.1)
		Results of Further Inference-Time Scaling (Voting@32)			
DeepSeek-GRM-27B (Ours)	88.5	65.3	60.4	69.7	71.0 (+3.1)
DeepSeek-GRM-27B (MetaRM) (Ours)	**90.4**	67.2	**63.2**	**70.3**	**72.8** (+4.9)

表 6: 不同方法和模型在RM基准上的推理时间可扩展性实验（表 3）的详细结果。下划线数字 表示最佳性能，**粗体数字** 表示基线和我们方法中的最佳性能，*斜体字* 表示标量或半标量RMs。对于元RM引导投票（MetaRM），$k_{\text{meta}} = \frac{1}{2}k$。括号中的数字是推理时间扩展后的性能变化。

et al., 2024)，一个用于诊断单个响应中错误的基准。具体来说，在整体评分计算中，我们不包括Reward Bench基准的先前数据集。

我们为每个基准使用标准评估指标：在Reward Bench、PPE 和 RMB 中从一组响应中挑选最佳响应的准确率，以及 ReaLMistake 的 ROC-AUC。RMB 基准的 BoN 子集为每个查询包含多个响应，并且只有当最佳响应被识别时，每个数据点才被视为正确。评估模型在 RMB BoN 子集上的默认设置是成对评估 $(n-1)$ 对，每对包含最佳响应和其他不同的响应，如果总共有 n 个响应。对于基线方法，我们采用这种方法进行评估。而对于我们的模型 (DeepSeek-GRM)，我们直接将所有响应输入模型并通过 $\arg\max_i S_i{}_{i=1}^{n}$ 来识别最佳响应，其中 S_i 是对第 i 个响应预测的奖励，这是一种更直接但更困难的方法，几乎不会影响性能。请参阅附录 E.1.1 进行经验分析。

对于DeepSeek-R1，由于推理成本大且延迟高，我们从Reward Bench基准中均匀抽取了300个数据点，并在该子集上测试DeepSeek-R1。结果如图 4(b) 所示。

D.3 Detailed Results

我们在图 6中提供了图 1的详细结果，并附上了更多公共模型的性能以供参考。我们在表6中提供了表 3的详细结果，并在表 7中提供了表 4的详细结果，其中包括每个RM基准测试的得分。此外，我们列出了所有测试方法在每个RM基准测试上的详细结果，其中Reward Bench基准测试的结果见表 8，PPE Correctness基准测试的结果见表 9，RMB基准测试的结果见表 10。我们发现DeepSeek-R1在Reward Bench基准测试的推理子集上取得了最高分，这表明长期推理可以提升GRMs在广泛场景中的推理能力。

E Additional Experiments

E.1 Input Flexibility of the Pointwise GRM Approach

在第2.1节中，我们从理论上证明了逐点GRM方法的输入灵活性。在本节中，我们提供了各种输入类型的实证证据以支持这一点。

and multiple responses, constructed from internal data and open-source datasets, including the training sets from MATH (Hendrycks et al., 2021), UltraFeedback (Cui et al., 2024), OffsetBias (Park et al., 2024), Skywork-Reward-Preference-80K-v0.2 (Liu et al., 2024), and HelpSteer2-Preference (Wang et al., 2025). Specifically, we retagged the preference label of a part of UltraFeedback due to its quality issues; we sampled and filtered trajectories on MATH by rule-based ground truth matching, resulting in pairwise preference data; for rating single responses, we set the ground truth reward to 1 for correct responses and 0 for incorrect ones, only incorporating verifiable questions. For rejective sampling, we use DeepSeek-v2.5-0906 to generate the trajectories with principles and critiques. The sampling time N_{RFT} is set to 3. During hinted sampling on HelpSteer2, we add the preference strengths labeled in the original dataset as the hint. We also remove the samples that are viewed too easy for DeepSeek-V2-Lite-Chat, i.e. all generated rewards are correct for three times according to Equation 4, from the RL data.

The derivation of DeepSeek-GRM models and the meta RM is illustrated in Figure 5. All DeepSeek-GRM models are trained from the pretrained version of LLMs. For the training of the meta RM, we reuse the rejective sampled data from the RFT stage, and use DeepSeek-GRM-27B to perform rejective sampling with $N_{\text{RFT}} = 3$, in order to avoid potential bias (Chow et al., 2025) in the meta RM guided voting. The learning rate is 1×10^{-5} and the batch size is 512 for the meta RM training. The training time of RFT and RL for DeepSeek-GRM-27B is depicted in Table 5, Gemma-2-27B based models are trained with 128 A100 GPUs on the Fire-Flyer platform (An et al., 2024). The learning rate is 5×10^{-6} for the RFT stage and 4×10^{-7} for the RL stage, and the batch size is 1024 for the RFT stage and 512 for the RL stage. Both stages are trained for 900 steps. Due to resource constraints, DeepSeek-GRM models larger than 27B does not undergo the rule-based RL and only trained with 50K rejective sampled data.

C.3 Baseline Implementation

For the baseline methods, we re-implement **LLM-as-a-Judge** (Zheng et al., 2023), **DeepSeek-BTRM-27B** (Kendall & Smith, 1940), **CLoud-Gemma-2-27B** (Ankner et al., 2024), and **DeepSeek-PairRM-27B** (Jiang et al., 2023) based on Gemma-2-27B (Team, 2024) and with all compatible training data and settings as DeepSeek-GRM.

For **LLM-as-a-Judge**, we use exactly the same training configuration as DeepSeek-GRM-27B, including RFT with rejective sampled data from DeepSeek-V2.5-0906 and rule-based online RL. Due to its scoring pattern, only pairwise data could be used in the RL stage. For **CLoud-Gemma-2-27B**, we also generate pointwise critiques from DeepSeek-V2.5-0906 using the same prompt template. However, it is not feasible to perform rejective sampling, since no rewards could be extracted without a trained value head. We fine-tune Gemma-2-27B with the same general instruction data of DeepSeek-GRM-27B along with the sampled critique, resulting in a critique generation model. Specifically, we fine-tune another Gemma-2-27B model with a value head for reward generation, instead of training value heads post hoc on the critique model. The training of the value head of CLoud-Gemma-2-27B, **DeepSeek-BTRM-27B**, and **DeepSeek-PairRM-27B** (Jiang et al., 2023) uses the same dataset from the RL stage of DeepSeek-GRM-27B, except for single response rating data.

Preprint. Under review.

Model	Reward Bench	PPE Preference	PPE Correctness	RMB	Overall
		Results of Greedy Decoding			
DeepSeek-GRM-27B	**86.0**	**64.7**	**59.8**	**69.0**	**69.9**
w/o Principle Generation	82.0	62.8	58.2	67.1	67.5
w/o Rejective Sampling	84.0	63.2	59.4	68.0	68.7
DeepSeek-GRM-27B-RFT	**84.5**	**64.1**	**59.6**	**67.0**	**68.8**
w/o Hinted Sampling (①)	83.0	63.8	58.2	65.8	68.0
w/o Non-Hinted Sampling (②)	82.5	63.4	58.6	65.2	67.4
w/o Rejective Sampling (①&②)	81.5	61.8	57.8	63.1	66.1
w/o General Instruction Data	79.1	59.2	51.5	63.2	63.3
		Results of Inference-Time Scaling (Voting@8)			
DeepSeek-GRM-27B	**87.7**	**64.9**	**60.3**	**69.5**	**70.6**
w/o Principle Generation	83.0	63.2	58.6	67.1	68.0
		Results of Inference-Time Scaling (Voting@32)			
DeepSeek-GRM-27B	88.5	65.3	60.4	69.7	71.0
DeepSeek-GRM-27B ($k_{\text{meta}} = 1$)	88.5	67.1	**65.2**	65.2	71.5
DeepSeek-GRM-27B ($k_{\text{meta}} = 8$)	89.7	**67.2**	64.7	69.1	72.7
DeepSeek-GRM-27B ($k_{\text{meta}} = 16$)	**90.4**	67.2	63.2	**70.3**	**72.8**

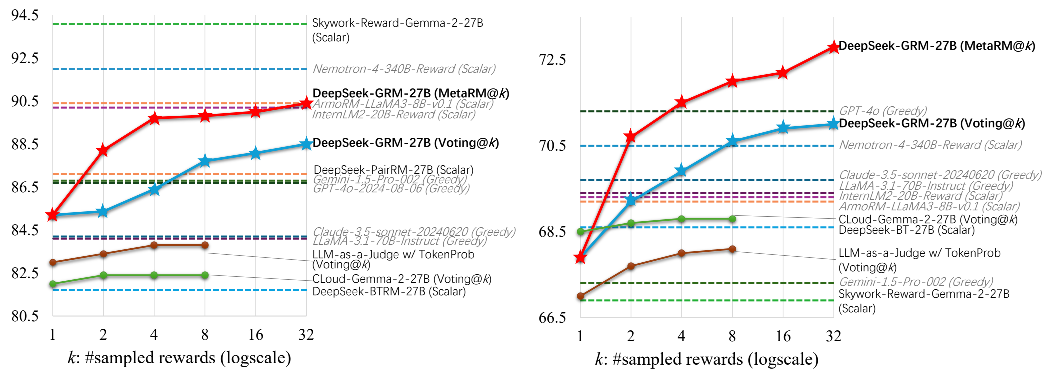
表 7: 为所提出的SPCT的不同组件提供了消融研究的详细结果（表 4）。**粗体数字**表示最佳性能。

Method	Chat	Chat Hard	Safety	Reasoning	Prior Sets	Reward Bench
		Results of Other Models				
DeepSeek-R1	97.1	73.7	73.3	95.6	-	84.9
DeepSeek-GRM-16B	90.8	74.3	84.7	81.8	62.5	82.9
DeepSeek-GRM-230B	96.5	72.5	87.8	84.3	-	85.3
DeepSeek-GRM-671B	95.8	82.9	88.3	86.6	-	88.4
		Results of Greedy Decoding				
LLM-as-a-Judge	96.7	69.3	83.5	84.3	-	83.4
DeepSeek-BTRM-27B	96.7	86.2	75.7	89.8	68.5	*81.7*
CLoud-Gemma-2-27B	*96.7*	*69.3*	*83.5*	*84.3*	-	*82.0*
DeepSeek-PairRM-27B	*95.5*	*86.8*	*52.3*	*92.0*	*67.6*	*87.1*
DeepSeek-GRM-27B-RFT (Ours)	94.7	77.2	87.0	79.2	65.9	84.5
DeepSeek-GRM-27B (Ours)	94.1	78.3	88.0	83.8	66.7	86.0
		Results of Inference-Time Scaling (Voting@8)				
LLM-as-a-Judge	95.0	70.0	83.5	85.0	-	83.4
LLM-as-a-Judge w/ **TokenProb**	95.8	71.3	83.3	84.8	-	83.8
CLoud-Gemma-2-27B	*96.7*	*85.8*	*56.2*	*91.0*	-	*82.4*
DeepSeek-GRM-27B-RFT (Ours)	94.7	79.0	87.3	80.2	-	85.3
DeepSeek-GRM-27B (Ours)	95.3	80.9	89.3	85.4	66.8	87.7
DeepSeek-GRM-27B (MetaRM) (Ours)	95.5	85.7	88.5	89.5	69.4	89.8
		Results of Further Inference-Time Scaling (Voting@32)				
DeepSeek-GRM-27B (Ours)	95.5	81.8	90.0	86.9	68.1	88.5
DeepSeek-GRM-27B (MetaRM) (Ours)	95.3	85.7	89.5	91.0	69.4	**90.4**

表 8: 在Reward Bench基准上不同方法的详细结果。下划线数字 表示最佳性能，**粗体数字**表示基线和我们方法中的最佳性能，*斜体字* 表示标量或半标量奖励模型 (RMs)。对于元奖励模型引导的投票 (MetaRM)，$k_{\text{meta}} = \frac{1}{2}k$。

E.1.1 *Generating Rewards for Many Responses*

在表 11中，我们展示了DeepSeek-GRM-27B在RMB基准的BoN子集上的实验结果，其中每个查询有多个响应。如果一个查询总共有$n, (n > 2)$个响应，那么成对输入设置是评估由最佳响应和其他响应组成的$(n-1)$对，并且只有当从所有$(n-1)$对中正确识别出最佳响应时，该数据点才被视为正确。这也是原始基准的默认设置。我们将DeepSeek-GRM-27B在成对输入和列表输入下的表现进行比较，其中列表输入设置是通过输入所有n个响应来识别最佳响应。结果显示，DeepSeek-GRM-27B几乎不受输入类型的影响，在有用性和无害性子集上的性能差异不到1%。这表明**逐点GRM能够灵活处理多个响应,并且其性能对输入类型不敏感**。



(a) Results on the Reward Bench benchmark. (b) Results on all tested reward modeling benchmarks.

图 6: Inference-time scaling performance with different RMs on different reward modeling benchmarks. Non-italic font indicates models based on Gemma-2-27B.

Model	Reward Bench	PPE Preference	PPE Correctness	RMB	Overall
		Reported Results of Public Models			
Nemotron-4-340B-Reward	<u>*92.0*</u>	*59.3*	*60.8*	*69.9*	*70.5*
GPT-4o	<u>86.7</u>	67.1	57.6	<u>73.8</u>	71.3
		Results of Inference-Time Scaling (Voting@1)			
LLM-as-a-Judge	83.0	63.4	57.4	64.3	67.0
CLoud-Gemma-2-27B	*82.0*	*67.0*	*62.0*	*63.2*	*68.5*
DeepSeek-GRM-27B-RFT (Ours)	84.0	62.2	59.4	65.8	67.8
DeepSeek-GRM-27B (Ours)	85.2	62.4	59.5	64.4	67.9
		Results of Inference-Time Scaling (Voting@8)			
LLM-as-a-Judge	83.4	63.8	58.2	65.2	67.6 (+0.6)
LLM-as-a-Judge w/ **TokenProb**	83.8	64.6	58.8	65.2	68.1 (+1.1)
CLoud-Gemma-2-27B	*82.4*	***67.3***	*62.4*	*63.2*	*68.8* (+0.3)
DeepSeek-GRM-27B-RFT (Ours)	85.3	<u>64.5</u>	59.7	67.7	69.3 (+1.5)
DeepSeek-GRM-27B (Ours)	87.7	64.9	60.3	69.5	70.6 (+2.7)
DeepSeek-GRM-27B (MetaRM) (Ours)	**89.8**	66.4	63.0	68.8	72.0 (+4.1)
		Results of Further Inference-Time Scaling (Voting@32)			
DeepSeek-GRM-27B (Ours)	88.5	65.3	60.4	69.7	71.0 (+3.1)
DeepSeek-GRM-27B (MetaRM) (Ours)	**90.4**	67.2	**63.2**	**70.3**	<u>72.8</u> (+4.9)

表 6: Detailed results of inference-time scalability experiments (Table 3) of different methods and models on RM benchmarks. <u>Underlined numbers</u> indicate the best performance, **bold numbers** indicate the best performance among baseline and our methods, and *italicized font* denotes scalar or semi-scalar RMs. For meta RM guided voting (MetaRM), $k_{\text{meta}} = \frac{1}{2}k$. Numbers in the parentheses is the performance change after inference-time scaling.

D Experiment Details

D.1 Hyper-Parameters

For inference-time scaling results of DeepSeek-GRM-27B, DeepSeek-GRM-16B, LLM-as-a-Judge, and CLoud-Gemma-2-27B, the temperature is set to 0.5 for each model. And for other experiments, temperature is set to 0 for all models. Without specific description, $k_{\text{meta}} = \frac{1}{2}k$ by default in the meta RM guided voting for DeepSeek-GRM-27B. For inference on DeepSeek-R1, the temperature is set to 0.6. Please note that we let DeepSeek-GRM to output rewards in the same range for rating single responses in the ReaLMistake benchmark as other benchmarks.

Method	MMLU-Pro	MATH	GPQA	MBPP-Plus	IFEval	PPE Correctness
			Results of Greedy Decoding			
LLM-as-a-Judge	66.0	68.0	52.8	50.2	56.8	58.8
DeepSeek-BTRM-27B	*68.8*	*73.2*	*56.8*	*68.8*	*66.0*	***66.7***
CLoud-Gemma-2-27B	*68.7*	*68.8*	*53.5*	*59.0*	*62.0*	*62.4*
DeepSeek-PairRM-27B	*68.3*	*74.7*	*55.0*	*63.1*	*62.9*	*64.8*
DeepSeek-GRM-27B-RFT (Ours)	64.8	68.7	55.5	49.0	60.2	59.6
DeepSeek-GRM-27B (Ours)	64.8	68.8	55.6	50.1	59.8	59.8
w/ Reference	98.2	97.5	99.8	86.6	75.9	<u>91.6</u>
			Results of Inference-Time Scaling (Voting@8)			
LLM-as-a-Judge	66.2	66.4	51.9	49.9	56.8	58.2
LLM-as-a-Judge w/ **TokenProb**	66.4	68.1	53.0	49.5	57.0	58.8
CLoud-Gemma-2-27B	*68.7*	*68.9*	*53.5*	*59.0*	*62.0*	*62.4*
DeepSeek-GRM-27B-RFT (Ours)	64.8	68.7	55.5	49.5	60.2	59.7
DeepSeek-GRM-27B (Ours)	65.7	68.7	55.5	50.0	61.6	60.3
DeepSeek-GRM-27B (MetaRM) (Ours)	68.0	68.7	57.3	51.3	69.9	63.0
			Results of Further Inference-Time Scaling (Voting@32)			
DeepSeek-GRM-27B (Ours)	65.5	69.4	56.0	49.9	61.0	60.4
DeepSeek-GRM-27B (MetaRM) (Ours)	68.1	70.0	56.9	50.8	70.4	63.2

表 9: 在PPE Correctness基准上不同方法的详细结果。

Method	Helpfulness BoN	Helpfulness Pairwise	Harmlessness BoN	Harmlessness Pairwise	RMB
		Results of Greedy Decoding			
LLM-as-a-Judge	55.8	78.5	50.8	73.9	64.8
DeepSeek-BTRM-27B	*64.0*	*83.0*	*33.6*	*51.0*	*57.9*
CLoud-Gemma-2-27B	*64.7*	*81.1*	*41.7*	*66.1*	*63.4*
DeepSeek-PairRM-27B	*59.9*	*83.3*	*34.1*	*55.5*	*58.2*
DeepSeek-GRM-27B-RFT (Ours)	58.4	79.3	54.2	76.0	67.0
DeepSeek-GRM-27B (Ours)	62.3	80.5	57.0	76.1	69.0
		Results of Inference-Time Scaling (Voting@8)			
LLM-as-a-Judge	56.0	78.5	52.5	73.8	65.2
LLM-as-a-Judge w/ **TokenProb**	56.0	78.5	52.5	73.8	65.2
CLoud-Gemma-2-27B	*63.8*	*82.1*	*40.9*	*66.1*	*63.2*
DeepSeek-GRM-27B-RFT (Ours)	59.2	80.1	54.8	76.5	67.7
DeepSeek-GRM-27B (Ours)	63.9	79.5	57.6	77.1	69.5
DeepSeek-GRM-27B (MetaRM) (Ours)	63.4	80.5	56.8	74.6	68.8
		Results of Further Inference-Time Scaling (Voting@32)			
DeepSeek-GRM-27B (Ours)	63.9	79.8	58.0	77.0	69.7
DeepSeek-GRM-27B (MetaRM) (Ours)	64.2	81.6	58.0	77.4	70.3

表 10: 人民币基准上不同方法的详细结果。 <u>下划线数字</u> 表示最佳性能，**粗体数字** 表示基线和我们方法中的最佳性能，*斜体字* 表示标量或半标量RMs。对于元RM引导投票（MetaRM），$k_{\text{meta}} = \frac{1}{2}k$。

Method	Helpfulness	Harmlessness
DeepSeek-GRM-27B		
w/ Pair Input	62.1	57.5
w/ List Input	62.3	57.0
|Δ|	0.2	0.5

表 11: 在RMB BoN基准上对响应输入类型进行的实验。

Method	Overall
DeepSeek-GRM-27B	59.8
w/ Voting@32	60.4
w/ Meta RM ($k_{\text{meta}} = 8$)	64.7
w/ Reference	91.6

表 12: 在PPE正确性基准上进行的基于参考的RM实验。

Model	Overall
DeepSeek-V2.5-0905	69.4
GPT-4o-2024-08-06	74.3
DeepSeek-V2-Lite-Chat	61.9
DeepSeek-GRM-16B (Ours)	64.9
Gemma-2-27B-it	65.8
DeepSeek-BTRM-27B	69.3
DeepSeek-GRM-27B (Ours)	72.2
DeepSeek-GRM-27B (Voting@8) (Ours)	74.4

表 13: 在ReaLMistake基准上的实验结果（ROC-AUC（%））。

E.1.2 Generating Rewards for Single Responses

在表 13中，我们展示了DeepSeek-GRM在16B和27B参数量下于ReaLMistake基准测试上的实验结果，其中每个查询仅有一个响应。我们将结果与公开模型进行比较，例如DeepSeek-V2.5-0905、GPT-4o-2024-08-06、DeepSeek-V2-Lite、Gemma-2-27B-it以及DeepSeek-BTRM-

Model	Reward Bench	PPE Preference	PPE Correctness	RMB	Overall
Results of Greedy Decoding					
DeepSeek-GRM-27B	**86.0**	**64.7**	**59.8**	**69.0**	**69.9**
w/o Principle Generation	82.0	62.8	58.2	67.1	67.5
w/o Rejective Sampling	84.0	63.2	59.4	68.0	68.7
DeepSeek-GRM-27B-RFT	**84.5**	**64.1**	**59.6**	**67.0**	**68.8**
w/o Hinted Sampling (①)	83.0	63.8	58.2	65.8	68.0
w/o Non-Hinted Sampling (②)	82.5	63.4	58.6	65.2	67.4
w/o Rejective Sampling (①&②)	81.5	61.8	57.8	63.1	66.1
w/o General Instruction Data	79.1	59.2	51.5	63.2	63.3
Results of Inference-Time Scaling (Voting@8)					
DeepSeek-GRM-27B	**87.7**	**64.9**	**60.3**	**69.5**	**70.6**
w/o Principle Generation	83.0	63.2	58.6	67.1	68.0
Results of Inference-Time Scaling (Voting@32)					
DeepSeek-GRM-27B	88.5	65.3	60.4	69.7	71.0
DeepSeek-GRM-27B ($k_{\text{meta}}=1$)	88.5	67.1	**65.2**	65.2	71.5
DeepSeek-GRM-27B ($k_{\text{meta}}=8$)	89.7	**67.2**	64.7	69.1	72.7
DeepSeek-GRM-27B ($k_{\text{meta}}=16$)	**90.4**	67.2	63.2	**70.3**	**72.8**

表 7: Detailed results of ablation studies (Table 4) for different components of the proposed SPCT. **Bold numbers** indicate the best performance.

Method	Chat	Chat Hard	Safety	Reasoning	Prior Sets	Reward Bench
Results of Other Models						
DeepSeek-R1	97.1	73.7	73.3	95.6	-	84.9
DeepSeek-GRM-16B	90.8	74.3	84.7	81.8	62.5	82.9
DeepSeek-GRM-230B	96.5	72.5	87.8	84.3	-	85.3
DeepSeek-GRM-671B	95.8	82.9	88.3	86.6	-	88.4
Results of Greedy Decoding						
LLM-as-a-Judge	96.7	69.3	83.5	84.3	-	83.4
DeepSeek-BTRM-27B	*96.7*	*86.2*	*75.7*	*89.8*	*68.5*	*81.7*
CLoud-Gemma-2-27B	*96.7*	*69.3*	*83.5*	*84.3*	*-*	*82.0*
DeepSeek-PairRM-27B	*95.5*	*86.8*	*52.3*	*92.0*	*67.6*	*87.1*
DeepSeek-GRM-27B-RFT (Ours)	94.7	77.2	87.0	79.2	65.9	84.5
DeepSeek-GRM-27B (Ours)	94.1	78.3	88.0	83.8	66.7	86.0
Results of Inference-Time Scaling (Voting@8)						
LLM-as-a-Judge	95.0	70.0	83.5	85.0	-	83.4
LLM-as-a-Judge w/ **TokenProb**	95.8	71.3	83.3	84.8	-	83.8
CLoud-Gemma-2-27B	*96.7*	*85.8*	*56.2*	*91.0*	*-*	*82.4*
DeepSeek-GRM-27B-RFT (Ours)	94.7	79.0	87.3	80.2	-	85.3
DeepSeek-GRM-27B (Ours)	95.3	80.9	89.3	85.4	66.8	87.7
DeepSeek-GRM-27B (MetaRM) (Ours)	95.5	85.7	88.5	89.5	69.4	89.8
Results of Further Inference-Time Scaling (Voting@32)						
DeepSeek-GRM-27B (Ours)	95.5	81.8	90.0	86.9	68.1	88.5
DeepSeek-GRM-27B (MetaRM) (Ours)	95.3	85.7	89.5	91.0	69.4	90.4

表 8: Detailed results of different methods on the Reward Bench benchmark. Underlined numbers indicate the best performance, **bold numbers** indicate the best performance among baseline and our methods, and *italicized font* denotes scalar or semi-scalar RMs. For meta RM guided voting (MetaRM), $k_{\text{meta}} = \frac{1}{2}k$.

D.2 Benchmarks

We evaluate the performance of different methods on various RM benchmarks of different domains: (1) **Reward Bench** (Lambert et al., 2024), a common benchmark for RM evaluation, with semi-automatically collected chat, reasoning, and safety preference data, where two responses require to be ranked for each query; (2) **PPE** (Frick et al., 2025), a large-scale benchmark containing crowdsourced preference data and correctness data for varifiable tasks, and each query has two responses; (3) **RMB** (Zhou et al., 2025), a more comprehensive benchmark with various types of preference data, focusing on helpfulness and harmlessness, and each query has two responses or more response in pairwise and best-of-N (BoN) subsets, respectively; (4) **ReaLMistake** (Kamoi et al., 2024), a benchmark for diagnosing the error

27B。结果显示，DeepSeek-GRM在相同规模的模型中表现最佳，并且在推理时扩展的情况下，其性能与最佳公开模型相当。这表明**逐点GRM能够有效地评估单一响应**。

E.1.3 *Generating Rewards with Reference*

在第 5.2 节中，我们表明标量和半标量 RM 可能存在显著的领域偏差，并且通常在可验证的问题上表现更好。为了解决这个问题，我们测试了 DeepSeek-GRM-27B，以针对这些带有参考的任务生成奖励，其中参考是每个查询的真实答案。结果如表 12 所示。我们发现，当提供参考时，DeepSeek-GRM-27B的准确率可以超过 90%。这表明 **逐点 GRM 能够有效地根据参考判断响应,从而缓解在可验证任务上的性能问题**。

E.2 Transferability of Generated Principles

我 们 在 第2.2节 中 使 用DeepSeek-GRM-27B生 成 的 原 则 扩 展 了 初 步 实 验。 我 们 对GPT-4o-2024-08-06和DeepSeek-GRM-27B进 行 了 测 试， 所 用 的 过 滤 后 原 则 与 表1中 的 完 全 相 同， 并 且 包 括 上 述DeepSeek-GRM-27B生成的原则。结果如表14所示。我们发现DeepSeek-GRM-27B生成的原则可以迁移到其他模型上，并且甚至略优于从GPT-4o手动筛选出的原则。这表明**DeepSeek-GRM-27B生成的原则具有鲁棒性和可迁移性到其他模型**。

Method	Chat Hard	IFEval
GPT-4o-2024-08-06	76.1	56.0
+Self-Gen. Principles	75.9	55.6
+Filtered Principles	77.8	57.5
+DGRM-27B-Gen. Principles	78.1	58.3
DeepSeek-GRM-27B	78.3	59.8
+Filtered Principles	77.0	58.5

表 14: 不同模型生成原则的可转移性实验。

E.3 Generalization beyond Training Data

Model	Chat	Chat Hard	Safety	Reasoning	Reward Bench
Results of Greedy Decoding					
DeepSeek-GRM-27B	**94.1**	**78.3**	**88.0**	**83.8**	**86.0**
w/o MATH RM Data	96.1	70.4	85.3	82.5	83.0
DeepSeek-GRM-16B	**90.8**	**74.3**	**84.7**	**81.8**	**82.9**
w/o MATH RM Data	95.0	63.4	76.9	74.3	77.4

表 15: 在Reward Bench基准上对训练数据泛化实验的结果。**粗体数字**表示最佳性能。

我们对DeepSeek-GRM-27B的训练数据泛化性进行了消融研究。我们从MATH训练集中移除了所有数据，并重新实施了训练方案。Reward Bench基准测试的结果如表 15所示。我们发现，仅仅添加与数学相关的偏好数据也能提升通用排名模型在各个领域上的表现，特别是在Chat Hard子集上。这一结果表明，DeepSeek-GRM-27B可以泛化到训练数据覆盖范围之外的领域。

E.4 Response Length Analysis for Rule-Based RL

我们在图 7中计算了在Reward Bench基准的每个子集上，基于规则的在线RL前后 DeepSeek-GRM-27B的响应长度。 DeepSeek-GRM-27B的标记数量是根据Gemma-2-27B的分词器计算的，而DeepSeek-R1的结果则使用其对应的分词器。我们发现，在RL之后，Chat子集的响应长度几乎没有增加，而Safety子集的响应长度甚至略有下降。响应长度的最大增长出现在Reasoning子集中，根据表 8，与DeepSeek-GRM-27B-RFT相比，DeepSeek-GRM-27B在

Method	MMLU-Pro	MATH	GPQA	MBPP-Plus	IFEval	PPE Correctness
			Results of Greedy Decoding			
LLM-as-a-Judge	66.0	68.0	52.8	50.2	56.8	58.8
DeepSeek-BTRM-27B	*68.8*	*73.2*	*56.8*	*68.8*	*66.0*	***66.7***
CLoud-Gemma-2-27B	*68.7*	*68.8*	*53.5*	*59.0*	*62.0*	*62.4*
DeepSeek-PairRM-27B	*68.3*	*74.7*	*55.0*	*63.1*	*62.9*	*64.8*
DeepSeek-GRM-27B-RFT (Ours)	64.8	68.7	55.5	49.0	60.2	59.6
DeepSeek-GRM-27B (Ours)	64.8	68.8	55.6	50.1	59.8	59.8
w/ Reference	98.2	97.5	99.8	86.6	75.9	91.6
			Results of Inference-Time Scaling (Voting@8)			
LLM-as-a-Judge	66.2	66.4	51.9	49.9	56.8	58.2
LLM-as-a-Judge w/ **TokenProb**	66.4	68.1	53.0	49.5	57.0	58.8
CLoud-Gemma-2-27B	*68.7*	*68.9*	*53.5*	*59.0*	*62.0*	*62.4*
DeepSeek-GRM-27B-RFT (Ours)	64.8	68.7	55.5	49.5	60.2	59.7
DeepSeek-GRM-27B (Ours)	65.7	68.7	55.5	50.0	61.6	60.3
DeepSeek-GRM-27B (MetaRM) (Ours)	68.0	68.7	57.3	51.3	69.9	63.0
			Results of Further Inference-Time Scaling (Voting@32)			
DeepSeek-GRM-27B (Ours)	65.5	69.4	56.0	49.9	61.0	60.4
DeepSeek-GRM-27B (MetaRM) (Ours)	68.1	70.0	56.9	50.8	70.4	63.2

表 9: Detailed results of different methods on the PPE Correctness benchmark.

Method	Helpfulness BoN	Helpfulness Pairwise	Harmlessness BoN	Harmlessness Pairwise	RMB
		Results of Greedy Decoding			
LLM-as-a-Judge	55.8	78.5	50.8	73.9	64.8
DeepSeek-BTRM-27B	*64.0*	*83.0*	*33.6*	*51.0*	*57.9*
CLoud-Gemma-2-27B	*64.7*	*81.1*	*41.7*	*66.1*	*63.4*
DeepSeek-PairRM-27B	*59.9*	*83.3*	*34.1*	*55.5*	*58.2*
DeepSeek-GRM-27B-RFT (Ours)	58.4	79.3	54.2	76.0	67.0
DeepSeek-GRM-27B (Ours)	62.3	80.5	57.0	76.1	69.0
		Results of Inference-Time Scaling (Voting@8)			
LLM-as-a-Judge	56.0	78.5	52.5	73.8	65.2
LLM-as-a-Judge w/ **TokenProb**	56.0	78.5	52.5	73.8	65.2
CLoud-Gemma-2-27B	*63.8*	*82.1*	*40.9*	*66.1*	*63.2*
DeepSeek-GRM-27B-RFT (Ours)	59.2	80.1	54.8	76.5	67.7
DeepSeek-GRM-27B (Ours)	63.9	79.5	57.6	77.1	69.5
DeepSeek-GRM-27B (MetaRM) (Ours)	63.4	80.5	56.8	74.6	68.8
		Results of Further Inference-Time Scaling (Voting@32)			
DeepSeek-GRM-27B (Ours)	63.9	79.8	58.0	77.0	69.7
DeepSeek-GRM-27B (MetaRM) (Ours)	64.2	81.6	58.0	77.4	70.3

表 10: Detailed results of different methods on the RMB benchmark. Underlined numbers indicate the best performance, **bold numbers** indicate the best performance among baseline and our methods, and *italicized font* denotes scalar or semi-scalar RMs. For meta RM guided voting (MetaRM), $k_{\text{meta}} = \frac{1}{2}k$.

within single responses. Specifically, we do not include the prior sets of the Reward Bench benchmark in overall score calculations.

We use the standard evaluation metrics for each benchmark: accuracy of picking the best response from a set of responses in Reward Bench, PPE, and RMB, and ROC-AUC for ReaLMistake. The BoN subsets of the RMB benchmark contains multiple responses for each query, and each data point is correct only when the best response is identified. The default setting to evaluate models on RMB BoN subsets is to pairwise evaluate $(n-1)$ pairs, where each pair includes the best response and another different response, if there is totally n responses. For baseline methods, we adopt this approach for evaluation. And for our models (DeepSeek-GRM), we directly input all responses to the model and identify the best response with $\arg\max_i S_i{}_{i=1}^{n}$, where S_i is the predicted reward for i-th response, which is a more direct but harder way, and barely affects the performance. Please refer to Appendix E.1.1 for empirical analysis.

For DeepSeek-R1, due to the large costs and latency of inference, we evenly down-sampled 300 data points from the Reward Bench benchmark, and test DeepSeek-R1 on this subset. The result is illustrated in Figure 4(b).

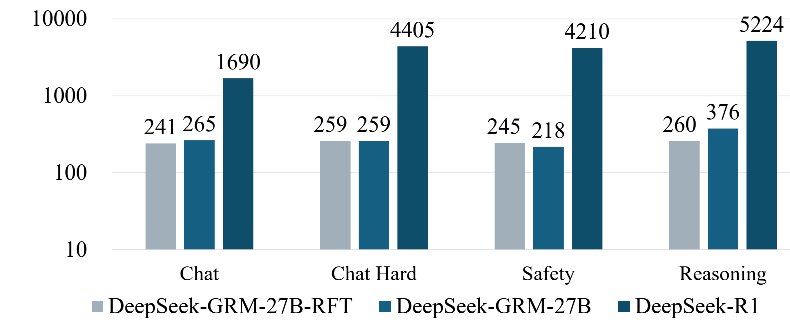


图 7: 在基于规则的在线强化学习前后，DeepSeek-GRM-27B在 Reward Bench 基准上的响应长度（#tokens）变化，与 DeepSeek-R1 进行了比较。

此子集上的性能也显著提升。这可能表明，DeepSeek-GRM-27B学会了在需要推理的任务上自适应地使用更多的推理计算资源，而对于一些其他领域（如安全性），模型在学会准确生成原则后可以节省计算资源。然而，DeepSeek-R1 使用了更多的标记数，但结果却较低，除了在推理任务上，这表明长时推理也有助于涉及广泛推理的排名模型任务。

F Qualitative Analysis

F.1 Case Study

我们在表 16、17 和 18 中提供了关于DeepSeek-GRM-27B的案例研究。第一个案例表明，作为标量奖励模型的DeepSeek-BTRM-27B在特定情况下可能被攻击或产生偏差，而DeepSeek-GRM-27B生成了文本原则和批评，展现出更好的鲁棒性。第二个案例展示了DeepSeek-GRM-27B的可扩展行为，在对多个样本进行投票后生成准确的奖励。相应的元奖励模型（meta RM）得分也显示了元奖励模型在指导投票过程中的有效性。第三个案例展示了DeepSeek-GRM-27B可能存在的潜在失败，这是由于模型无法准确判断遵循某些原则的响应所导致的，例如评估实时数据，而且各个原则的权重可能未达到平衡。

Query	Please help me review for my behavioral neuroscience exam by giving me the core information that is essential to this course. Please give an expansive review and include explanations in analogy or metaphors

Method	Helpfulness	Harmlessness
DeepSeek-GRM-27B		
w/ Pair Input	62.1	57.5
w/ List Input	62.3	57.0
$\lvert\Delta\rvert$	0.2	0.5

表 11: Experiments of response input types on the RMB BoN benchmarks.

Method	Overall
DeepSeek-GRM-27B	59.8
w/ Voting@32	60.4
w/ Meta RM ($k_{\text{meta}} = 8$)	64.7
w/ Reference	91.6

表 12: Experiments on reference-based RM on the PPE correctness benchmark.

Model	Overall
DeepSeek-V2.5-0905	69.4
GPT-4o-2024-08-06	74.3
DeepSeek-V2-Lite-Chat	61.9
DeepSeek-GRM-16B (Ours)	64.9
Gemma-2-27B-it	65.8
DeepSeek-BTRM-27B	69.3
DeepSeek-GRM-27B (Ours)	72.2
DeepSeek-GRM-27B (Voting@8) (Ours)	74.4

表 13: Experimental results (ROC-AUC (%)) on the ReaLMistake benchmark.

D.3 Detailed Results

We provide detailed results of Figure 1 in Figure 6, with performance of more public models for reference. We provide detailed results of Table 3 in Table 6, and detailed results of Table 4 in Table 7, with scores on each RM benchmark. Furthermore, we list detailed results for all tested methods on each RM benchmarks, with the Reward Bench benchmark in Table 8, the PPE Correctness benchmark in Table 9, and the RMB benchmark in Table 10. We found that DeepSeek-R1 achieves the highest result in the Reasoning subset of the Reward Bench benchmark, indicating that long-horizon reasoning could boost GRMs in reasoning extensive scenarios.

E Additional Experiments

E.1 Input Flexibility of the Pointwise GRM Approach

In Section 2.1, we demonstrate the input flexibility of the pointwise GRM approach theoretically. In this section, we provide empirical evidence on various input types to support it.

E.1.1 Generating Rewards for Many Responses

In Table 11, we show the experimental results of DeepSeek-GRM-27B on the BoN subsets of the RMB benchmark, where each query has multiple responses. If there is at total n, $(n > 2)$ responses for a query, the pair input setting is to evaluate $(n-1)$ pairs comprise of the best response and the other responses, and only when the best response is correctly identified from all $(n-1)$ pairs, the data point is considered as correct. It is also the default setting for the original benchmark. We compare the performance of DeepSeek-GRM-27B with pair input and list input, where the list input setting is to identify the best response with inputting all n responses. The result shows that DeepSeek-GRM-27B is barely affected by the input types, and the performance difference is less than 1% on both helpfulness and harmlessness subsets. This indicates that **the pointwise GRM is flexible to input many responses, and the performance is not sensitive to the input types**.

Response 1	Behavioral neuroscience, also known as biological psychology, explores the relationship between the brain and behavior. It examines how the nervous system influences cognitive processes, emotions, and motor functions. Here's a comprehensive review with analogies and metaphors:\n 1. **Neuroanatomy**: Imagine the brain as a complex city with different neighborhoods (lobes) and roads (neuronal pathways). The cerebral cortex is the outer layer, like the skin of the city, responsible for higher functions like thinking and perception. The cerebellum is like a choreographer, ensuring smooth and coordinated movements. The brainstem acts as the city's backbone, connecting the brain with the spinal cord, and controlling basic life functions. 2. **Neurons**: These are the building blocks of the nervous system. Think of them as the city's messengers, transmitting information via electrical and chemical signals. The dendrites receive messages, the cell body processes them, and the axon sends the signal to other neurons at the synapse. 3. **Synapses and neurotransmitters**: The synapse is like a tiny bridge between neurons. Neurotransmitters are the messengers that cross this bridge, transmitting signals from one neuron to another. Analogous to different languages, there are various neurotransmitters (e.g., dopamine, serotonin, glutamate) that influence distinct aspects of behavior and mood. 4. **Sensory Systems**: These are the city's surveillance cameras, detecting external stimuli. For example, the visual system (retina) captures light, and the auditory system (cochlea) picks up sound waves. These systems transmit information to the brain for processing. 5. **Motor Systems**: The motor systems act like the city's transportation network, sending signals from the brain to muscles, enabling movement. The somatic motor system controls voluntary movements, while the autonomic nervous system is like an unseen hand managing involuntary functions like heart rate and digestion. 6. ** Plasticity**: Brain plasticity is like the city's ability to adapt and change over time. It refers to the brain's capacity to reorganize itself by forming new neural connections in response to experiences, learning, or injury. 7. **Memory**: Memory can be thought of as a vast library where information is stored and retrieved. There are three main types: sensory memory (like a quick glance at a book), short-term memory (reading a paragraph), and long-term memory (remembering the entire book). 8. **Learning**: Learning is the process of acquiring new knowledge, skills, or behaviors. It's like adding new books to the library or updating old information. There are different types, such as classical conditioning (learning through association, like Pavlov's dogs) and operant conditioning (learning through consequences, like reinforcements and punishments). 9. **Emotions and Motivation**: Emotions are like the city's weather, influencing how we perceive and respond to events. Different brain regions, like the amygdala (the emotional storm center), contribute to emotional experiences. Motivation can be seen as the city's energy, driving us to act in pursuit of goals, with systems like the mesolimbic dopamine pathway playing a key role. 10. **Hormones and the Endocrine System**: These are the city's internal postal service, sending chemical messages throughout the body. Hormones, like cortisol and oxytocin, can influence behavior and mood by

E.1.2 Generating Rewards for Single Responses

In Table 13, we show the experimental results of DeepSeek-GRM in 16B and 27B on the ReaLMistake benchmark, where each query has only one response. We compare with public models, e.g., DeepSeek-V2.5-0905, GPT-4o-2024-08-06, DeepSeek-V2-Lite, and Gemma-2-27B-it, and DeepSeek-BTRM-27B. The result shows that DeepSeek-GRM achieves the best performance among models with the same size, and comparable performance with the best public models with inference-time scaling. This indicates that **the pointwise GRM could effectively rate single responses**.

E.1.3 Generating Rewards with Reference

In Section 5.2, we show that scalar and semi-scalar RMs could have significant domain biases, and generally perform better on verifiable questions. To alleviate this issue, we test DeepSeek-GRM-27B to generate rewards for these tasks with reference, where the reference is the ground truth for each query. The results are shown in Table 12. We find that DeepSeek-GRM-27B could achieve a more than 90% accuracy with reference provided. This indicates that **the pointwise GRM could effectively judge responses with reference, mitigating performance on verifiable tasks**.

E.2 Transferability of Generated Principles

We extend the preliminary experiment in Section 2.2 with DeepSeek-GRM-27B generated principles. We test GPT-4o-2024-08-06 and DeepSeek-GRM-27B with the filtered principles exactly the same as Table 1, and aforementioned DeepSeek-GRM-27B generated ones. The results are shown in Table 14. We find that the principles generated by DeepSeek-GRM-27B could be transferred to other models, and are even slightly better than manually filtered principles from GPT-4o. This indicates that **the principles generated by DeepSeek-GRM-27B are robust and transferable to other models**.

Method	Chat Hard	IFEval
GPT-4o-2024-08-06	76.1	56.0
+Self-Gen. Principles	75.9	55.6
+Filtered Principles	77.8	57.5
+DGRM-27B-Gen. Principles	78.1	58.3
DeepSeek-GRM-27B	78.3	59.8
+Filtered Principles	77.0	58.5

表 14: Experiments of the transferability of principles generated by different models.

E.3 Generalization beyond Training Data

Model	Chat	Chat Hard	Safety	Reasoning	Reward Bench
Results of Greedy Decoding					
DeepSeek-GRM-27B	**94.1**	**78.3**	**88.0**	**83.8**	**86.0**
w/o MATH RM Data	96.1	70.4	85.3	82.5	83.0
DeepSeek-GRM-16B	**90.8**	**74.3**	**84.7**	**81.8**	**82.9**
w/o MATH RM Data	95.0	63.4	76.9	74.3	77.4

表 15: Results of training data generalization experiments on the Reward Bench benchmark. **Bold numbers** indicate the best performance.

We conduct ablation study on the generalization of training data for DeepSeek-GRM-27B. We remove the all data from MATH training set, and re-implement the training recipe.

Response 2

Certainly! Behavioral neuroscience is a fascinating field that explores the intricate relationship between the brain and behavior. It's like unraveling the mysteries of a complex machine, where the brain is the central processing unit, and behavior is the output. Let's dive into the core concepts that form the foundation of this course.\n
1. Neuroanatomy: Understanding the Brain's Architecture\n The brain is a remarkable structure, and understanding its anatomy is crucial. Think of it as a vast city, with different regions serving as specialized districts. The cerebral cortex is the bustling metropolis, where higher cognitive functions like decision-making and problem-solving take place. The limbic system, on the other hand, is the emotional heart, regulating feelings and memories. And deep within, the brainstem acts as the city's control center, managing essential functions like breathing and heartbeat.\n
2. Neurophysiology: The Language of the Brain\n Just as a city communicates through intricate networks of roads and signals, the brain uses electrical and chemical signals to transmit information. Neurons, the brain's messengers, fire action potentials like lightning bolts, carrying vital information. Neurotransmitters, on the other hand, are the chemical couriers, facilitating communication between neurons at synapses, akin to the handoff of a letter from one postal worker to another.\n
3. Sensory and Motor Systems: The Brain's Gateways\n The brain is not an isolated entity; it constantly interacts with the external world. The sensory systems act as the brain's eyes, ears, and touch receptors, gathering information from the environment. Imagine them as surveillance cameras and sensors, feeding data into the brain's processing centers. Conversely, the motor systems are the brain's actuators, executing commands and translating neural signals into physical movements, like a robotic arm carrying out instructions.\n
4. Learning and Memory: The Brain's Database\n Just like a computer stores and retrieves data, the brain has its own mechanisms for learning and memory. Long-term potentiation (LTP) is like upgrading the bandwidth between neurons, strengthening connections and facilitating information storage. Consolidation is akin to backing up data, transferring memories from short-term to long-term storage. And retrieval is like accessing a file, recalling stored information when needed.\n
5. Motivation and Emotion: The Brain's Driving Forces\n Behavior is not just a product of cognitive processes; it's also influenced by motivation and emotion. The reward system, centered around structures like the nucleus accumbens, acts as the brain's carrot-and-stick, reinforcing desirable behaviors with pleasurable sensations. The amygdala, on the other hand, is the brain's security guard, processing emotional cues and triggering appropriate responses, like the fight-or-flight reflex.\n
6. Neuroplasticity: The Brain's Adaptability\n The brain is not a static entity; it's a dynamic and ever-changing landscape. Neuroplasticity is the brain's ability to reorganize and rewire itself in response to experience and environmental changes. It's like a city constantly undergoing renovations, adapting to new demands and challenges. This plasticity allows the brain to recover from injuries, acquire new skills, and adapt to changing circumstances.\n
7. Neurological Disorders: When the Brain Malfunctions\n Just as a city can experience breakdowns and malfunctions, the brain is susceptible to various disorders and dysfunctions. Neurodegenerative diseases like Alzheimer's and Parkinson's are akin to the gradual deterioration of a city's infrastructure, leading to cognitive and motor impairments.

Results on the Reward Bench benchmark are shown in Table 15. We found that merely adding math related preference data could also boost generalist RM performance on various domains, especially on the Chat Hard subset. The result reveals that DeepSeek-GRM-27B could generalize to domains beyond the coverage of training data.

E.4 Response Length Analysis for Rule-Based RL

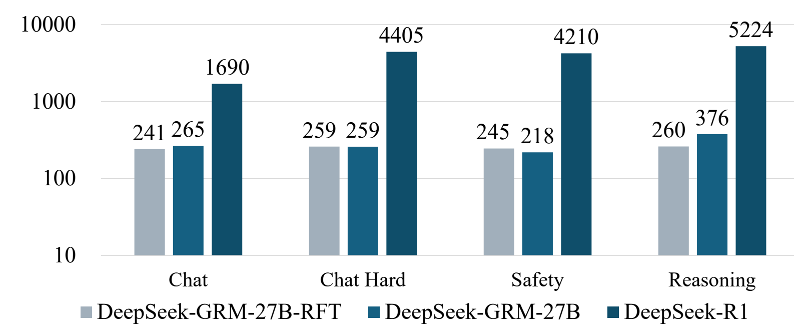


图 7: The changes of response lengths (#tokens) of DeepSeek-GRM-27B before and after rule-based online RL on the Reward Bench benchmark, compared with DeepSeek-R1.

We calculate the response lengths of DeepSeek-GRM-27B before and after rule-based online RL on each subset of the Reward Bench benchmark in Figure 7. The token count of DeepSeek-GRM-27B is calculated based on the tokenizer of Gemma-2-27B, while the result of DeepSeek-R1 uses its corresponding tokenizer. We found that the response length for the Chat subset barely increases in RL, and the response length for the Safety subset even drops slightly. The largest increase of response lengths occurs in the Reasoning subset, where the performance of DeepSeek-GRM-27B also improves greatly compared to DeepSeek-GRM-27B-RFT, according to Table 8. This might indicate that DeepSeek-GRM-27B learns to adaptively use more inference compute on reasoning extensive tasks, and the compute could be saved for some other domains, such as safety, after the model learns to generate principles accurately. However, DeepSeek-R1 uses way more tokens and achieves lower results, except for Reasoning, which shows that long-horizon reasoning also helps RM tasks regarding to extensive reasoning.

F Qualitative Analysis

F.1 Case Study

We provide a case study on DeepSeek-GRM-27B in Table 16, 17 and 18. The first case shows that DeepSeek-BTRM-27B as a scalar RM could be hacked or biased under specific circumstances, and DeepSeek-GRM-27B generates textual principles and critiques, showing better robustness. The second case shows the scalable behaviors of DeepSeek-GRM-27B, generating accurate rewards after voting on multiple samples. The according meta RM scores also show the effectiveness of the meta RM in guiding the voting process. The third case shows the potential failure of DeepSeek-GRM-27B which is caused by the inability of the model to accurately judge responses following some principles, e.g., assessing real-time data, and the weights of each principle might not be balanced.

Preprint. Under review.

DeepSeek-BTRM-27B Result	Response 1: 0.4665; Response 2: 0.3209.
DeepSeek-GRM-27B Result	Specific Criteria: 1. **Depth of Analogies and Metaphors** (Weight: 30%): The extent to which the analogies and metaphors used are deep, creative, and effectively convey complex concepts. 2. **Comprehensiveness of Core Information** (Weight: 30%): The thoroughness with which the core information essential to behavioral neuroscience is covered. 3. **Clarity and Organization** (Weight: 20%): The clarity of explanations and the logical organization of the information. 4. **Engagement and Appeal** (Weight: 20%): The level of engagement and appeal the response has, making the information interesting and memorable. Analysis: - **Response 1**: This response excels in comprehensiveness and clarity, covering all major aspects of behavioral neuroscience with detailed and well-organized explanations. The analogies and metaphors are well-crafted and effectively enhance understanding. However, some metaphors could be more creative to enhance engagement further. - **Response 2**: This response is also comprehensive and clear, but it stands out more in the depth and creativity of its analogies and metaphors. The organization is logical, and the explanations are clear. The engagement level is high, making the information appealing and memorable. Scores: \boxed{8, 9}
Ground Truth	*Response 2 is better than Response 1.*

表 16: 一个关于DeepSeek-GRM-27B和DeepSeek-BTRM-27B的案例，其中DeepSeek-GRM-27B在区分响应的细微差异方面表现出更好的鲁棒性。

34

Query	Please help me review for my behavioral neuroscience exam by giving me the core information that is essential to this course. Please give an expansive review and include explanations in analogy or metaphors

Preprint. Under review.

Query	Write a JavaScript function 'const tri = (n)' to solve the following problem: Everyone knows Fibonacci sequence, it was studied deeply by mathematicians in the last couple centuries. However, what people don't know is Tribonacci sequence. Tribonacci sequence is defined by the recurrence: tri(1) = 3 tri(n) = 1 + n / 2, if n is even. tri(n) = tri(n - 1) + tri(n - 2) + tri(n + 1), if n is odd. For example: tri(2) = 1 + (2 / 2) = 2 tri(4) = 3 tri(3) = tri(2) + tri(1) + tri(4) = 2 + 3 + 3 = 8 You are given a non-negative integer number n, you have to a return a list of the first n + 1 numbers of the Tribonacci sequence. Examples: tri(3) = [1, 3, 2, 8]

Response 1	<pre> if (n == 0) { return [1] }
	  if (n == 1) { return [1, 3] }
	  let p = [1, 3]
	  for (let i = 2; i <= n; i++) {
	    if (i % 2 == 0) {
	      p.push(1 + i / 2)
	    }
	    else {
	      p.push(p[i - 2] + p[i - 1] + 1 + i + (i + 1) / 2)
	    }
	  }
	  return p
	}</pre>

Response 2	<pre> if (n == 0) { return [1] }
	  if (n == 1) { return [1, 3] }
	  let p = [1, 3]
	  for (let i = 2; i <= n; i++) {
	    if (i % 2 == 0) {
	      p.push(1 + i / 2)
	    }
	    else {
	      p.push(p[i - 2] + p[i - 1] + 1 + (i + 1) / 2)
	    }
	  }
	  return p
	}</pre>

Response 1	Behavioral neuroscience, also known as biological psychology, explores the relationship between the brain and behavior. It examines how the nervous system influences cognitive processes, emotions, and motor functions. Here's a comprehensive review with analogies and metaphors:\n 1. **Neuroanatomy**: Imagine the brain as a complex city with different neighborhoods (lobes) and roads (neuronal pathways). The cerebral cortex is the outer layer, like the skin of the city, responsible for higher functions like thinking and perception. The cerebellum is like a choreographer, ensuring smooth and coordinated movements. The brainstem acts as the city's backbone, connecting the brain with the spinal cord, and controlling basic life functions. 2. **Neurons**: These are the building blocks of the nervous system. Think of them as the city's messengers, transmitting information via electrical and chemical signals. The dendrites receive messages, the cell body processes them, and the axon sends the signal to other neurons at the synapse. 3. **Synapses and neurotransmitters**: The synapse is like a tiny bridge between neurons. Neurotransmitters are the messengers that cross this bridge, transmitting signals from one neuron to another. Analogous to different languages, there are various neurotransmitters (e.g., dopamine, serotonin, glutamate) that influence distinct aspects of behavior and mood. 4. **Sensory Systems**: These are the city's surveillance cameras, detecting external stimuli. For example, the visual system (retina) captures light, and the auditory system (cochlea) picks up sound waves. These systems transmit information to the brain for processing. 5. **Motor Systems**: The motor systems act like the city's transportation network, sending signals from the brain to muscles, enabling movement. The somatic motor system controls voluntary movements, while the autonomic nervous system is like an unseen hand managing involuntary functions like heart rate and digestion. 6. ** Plasticity**: Brain plasticity is like the city's ability to adapt and change over time. It refers to the brain's capacity to reorganize itself by forming new neural connections in response to experiences, learning, or injury. 7. **Memory**: Memory can be thought of as a vast library where information is stored and retrieved. There are three main types: sensory memory (like a quick glance at a book), short-term memory (reading a paragraph), and long-term memory (remembering the entire book). 8. **Learning**: Learning is the process of acquiring new knowledge, skills, or behaviors. It's like adding new books to the library or updating old information. There are different types, such as classical conditioning (learning through association, like Pavlov's dogs) and operant conditioning (learning through consequences, like reinforcements and punishments). 9. **Emotions and Motivation**: Emotions are like the city's weather, influencing how we perceive and respond to events. Different brain regions, like the amygdala (the emotional storm center), contribute to emotional experiences. Motivation can be seen as the city's energy, driving us to act in pursuit of goals, with systems like the mesolimbic dopamine pathway playing a key role. 10. **Hormones and the Endocrine System**: These are the city's internal postal service, sending chemical messages throughout the body. Hormones, like cortisol and oxytocin, can influence behavior and mood by

DeepSeek-GRM-27B Result 1	### Output **Specific Criteria:** 1. **Correctness of Sequence Calculation (Weight: 40%)**: The response must correctly implement the Tribonacci sequence as defined, including handling both even and odd cases accurately. 2. **Initialization and Base Cases (Weight: 20%)**: The response should correctly initialize the sequence and handle base cases (n = 0 and n = 1) appropriately. 3. **Loop and Conditional Logic (Weight: 20%)**: The response should use appropriate loop and conditional logic to generate the sequence. 4. **Output Format (Weight: 20%)**: The response should return the sequence in the correct format as specified. **Analysis:** - **Response 1**: - **Correctness of Sequence Calculation**: The logic for even and odd cases is mostly correct, but there is a mistake in the calculation for odd numbers where it incorrectly adds '1 + i + (i + 1) / 2'. - **Initialization and Base Cases**: Correctly initializes and handles base cases. - **Loop and Conditional Logic**: Uses appropriate loop and conditional logic. - **Output Format**: Returns the sequence in the correct format. - **Response 2**: - **Correctness of Sequence Calculation**: Similar to Response 1, the logic for even and odd cases is mostly correct, but there is a mistake in the calculation for odd numbers where it incorrectly adds '1 + (i + 1) / 2'. - **Initialization and Base Cases**: Correctly initializes and handles base cases. - **Loop and Conditional Logic**: Uses appropriate loop and conditional logic. - **Output Format**: Returns the sequence in the correct format. **Scores:** - **Response 1**: - Instruction Adherence: 8 (Partially Adhered)\n - Usefulness: 7 (Useful but Incomplete)\n - Level of Detail: 8 (Detailed but Slightly Lacking)\n - Relevance: 9 (Highly Relevant)\n - **Overall Score**: 8\n - **Response 2**: - Instruction Adherence: 8 (Partially Adhered)\n - Usefulness: 7 (Useful but Incomplete)\n - Level of Detail: 8 (Detailed but Slightly Lacking)\n - Relevance: 9 (Highly Relevant)\n - **Overall Score**: 8 **Scores: \boxed{8, 8}**
Meta RM Score 1	-15.7781

Response 2	Certainly! Behavioral neuroscience is a fascinating field that explores the intricate relationship between the brain and behavior. It's like unraveling the mysteries of a complex machine, where the brain is the central processing unit, and behavior is the output. Let's dive into the core concepts that form the foundation of this course.\n 1. Neuroanatomy: Understanding the Brain's Architecture\n The brain is a remarkable structure, and understanding its anatomy is crucial. Think of it as a vast city, with different regions serving as specialized districts. The cerebral cortex is the bustling metropolis, where higher cognitive functions like decision-making and problem-solving take place. The limbic system, on the other hand, is the emotional heart, regulating feelings and memories. And deep within, the brainstem acts as the city's control center, managing essential functions like breathing and heartbeat.\n 2. Neurophysiology: The Language of the Brain\n Just as a city communicates through intricate networks of roads and signals, the brain uses electrical and chemical signals to transmit information. Neurons, the brain's messengers, fire action potentials like lightning bolts, carrying vital information. Neurotransmitters, on the other hand, are the chemical couriers, facilitating communication between neurons at synapses, akin to the handoff of a letter from one postal worker to another.\n 3. Sensory and Motor Systems: The Brain's Gateways\n The brain is not an isolated entity; it constantly interacts with the external world. The sensory systems act as the brain's eyes, ears, and touch receptors, gathering information from the environment. Imagine them as surveillance cameras and sensors, feeding data into the brain's processing centers. Conversely, the motor systems are the brain's actuators, executing commands and translating neural signals into physical movements, like a robotic arm carrying out instructions.\n 4. Learning and Memory: The Brain's Database\n Just like a computer stores and retrieves data, the brain has its own mechanisms for learning and memory. Long-term potentiation (LTP) is like upgrading the bandwidth between neurons, strengthening connections and facilitating information storage. Consolidation is akin to backing up data, transferring memories from short-term to long-term storage. And retrieval is like accessing a file, recalling stored information when needed.\n 5. Motivation and Emotion: The Brain's Driving Forces\n Behavior is not just a product of cognitive processes; it's also influenced by motivation and emotion. The reward system, centered around structures like the nucleus accumbens, acts as the brain's carrot-and-stick, reinforcing desirable behaviors with pleasurable sensations. The amygdala, on the other hand, is the brain's security guard, processing emotional cues and triggering appropriate responses, like the fight-or-flight reflex.\n 6. Neuroplasticity: The Brain's Adaptability\n The brain is not a static entity; it's a dynamic and ever-changing landscape. Neuroplasticity is the brain's ability to reorganize and rewire itself in response to experience and environmental changes. It's like a city constantly undergoing renovations, adapting to new demands and challenges. This plasticity allows the brain to recover from injuries, acquire new skills, and adapt to changing circumstances.\n 7. Neurological Disorders: When the Brain Malfunctions\n Just as a city can experience breakdowns and malfunctions, the brain is susceptible to various disorders and dysfunctions. Neurodegenerative diseases like Alzheimer's and Parkinson's are akin to the gradual deterioration of a city's infrastructure, leading to cognitive and motor impairments.

Preprint. Under review.

DeepSeek-GRM-27B Result 2	### Output Specific Criteria: 1. **Correctness of Tribonacci Sequence Calculation** (Weight: 40%): The response must correctly implement the Tribonacci sequence as defined in the problem. 2. **Code Structure and Readability** (Weight: 30%): The code should be well-structured, readable, and follow best practices. 3. **Edge Case Handling** (Weight: 20%): The response should correctly handle edge cases such as 'n = 0' and 'n = 1'. 4. **Efficiency** (Weight: 10%): The code should efficiently compute the sequence without unnecessary computations. Analysis: - **Response 1**: - **Correctness of Tribonacci Sequence Calculation**: The response correctly implements the Tribonacci sequence for both even and odd indices. - **Code Structure and Readability**: The code is structured well and is readable. It uses a loop to generate the sequence and correctly handles the conditions for even and odd indices. - **Edge Case Handling**: The response correctly handles the edge cases for 'n = 0' and 'n = 1'. - **Efficiency**: The code efficiently computes the sequence with a single loop. - **Response 2**: - **Correctness of Tribonacci Sequence Calculation**: The response incorrectly computes the sequence for odd indices by adding an extra '1 + i + (i + 1) / 2', which deviates from the defined Tribonacci sequence. - **Code Structure and Readability**: The code is structured similarly to Response 1 but includes an incorrect computation. - **Edge Case Handling**: The response correctly handles the edge cases for 'n = 0' and 'n = 1'. - **Efficiency**: The code is efficient but the incorrect computation affects its usefulness. Scores: \boxed{9, 5}
Meta RM Score 2	1.3126

37

DeepSeek-BTRM-27B Result	Response 1: 0.4665; Response 2: 0.3209.
DeepSeek-GRM-27B Result	Specific Criteria: 1. **Depth of Analogies and Metaphors** (Weight: 30%): The extent to which the analogies and metaphors used are deep, creative, and effectively convey complex concepts. 2. **Comprehensiveness of Core Information** (Weight: 30%): The thoroughness with which the core information essential to behavioral neuroscience is covered. 3. **Clarity and Organization** (Weight: 20%): The clarity of explanations and the logical organization of the information. 4. **Engagement and Appeal** (Weight: 20%): The level of engagement and appeal the response has, making the information interesting and memorable. Analysis: - **Response 1**: This response excels in comprehensiveness and clarity, covering all major aspects of behavioral neuroscience with detailed and well-organized explanations. The analogies and metaphors are well-crafted and effectively enhance understanding. However, some metaphors could be more creative to enhance engagement further. - **Response 2**: This response is also comprehensive and clear, but it stands out more in the depth and creativity of its analogies and metaphors. The organization is logical, and the explanations are clear. The engagement level is high, making the information appealing and memorable. Scores: \boxed{8, 9}
Ground Truth	*Response 2 is better than Response 1.*

表 16: A case of DeepSeek-GRM-27B and DeepSeek-BTRM-27B, where DeepSeek-GRM-27B shows better robustness in distinguishing slight differences of responses.

Preprint. Under review.

DeepSeek-GRM-27B Result 3	### Specific Criteria: 1. **Correctness of Tribonacci Sequence Calculation (Weight: 40%)**: - The response must correctly implement the Tribonacci sequence as defined, including handling both even and odd cases accurately. 2. **Code Structure and Readability (Weight: 30%)**: - The code should be well-structured, readable, and follow best practices in JavaScript. 3. **Edge Case Handling (Weight: 20%)**: - The response should correctly handle edge cases such as 'n = 0' and 'n = 1'. 4. **Efficiency (Weight: 10%)**: - The code should be efficient in terms of time and space complexity, avoiding unnecessary computations. ### Analysis: - **Response 1**: - **Correctness of Tribonacci Sequence Calculation**: The response correctly implements the Tribonacci sequence for both even and odd cases. It accurately follows the given recurrence relations. - **Code Structure and Readability**: The code is structured well and is readable. It uses a loop to generate the sequence and handles the initial conditions correctly. - **Edge Case Handling**: The response correctly handles the edge cases 'n = 0' and 'n = 1'. - **Efficiency**: The code is efficient, with a time complexity of O(n) and space complexity of O(n). - **Response 2**: - **Correctness of Tribonacci Sequence Calculation**: The response has a logical error in the calculation for odd 'n'. It incorrectly adds '1 + i + (i + 1) / 2' instead of 'p[i - 2] + p[i - 1] + 1 + (i + 1) / 2'. - **Code Structure and Readability**: The code structure is similar to Response 1, but the logical error makes it less readable. - **Edge Case Handling**: The response correctly handles the edge cases 'n = 0' and 'n = 1'. - **Efficiency**: The code is efficient in terms of time and space complexity, but the logical error impacts its correctness. ### Scores: - **Response 1**: - Instruction Adherence: 10\n - Usefulness: 10\n - Level of Detail: 9\n - Relevance: 10\n - **Overall Score**: 10\n - **Response 2**: - Instruction Adherence: 8\n - Usefulness: 6\n - Level of Detail: 8\n - Relevance: 8\n - **Overall Score**: 7 ### Output: \boxed{10, 7}

38

Query

Write a JavaScript function 'const tri = (n)' to solve the following problem:
Everyone knows Fibonacci sequence, it was studied deeply by mathematicians in
the last couple centuries. However, what people don't know is Tribonacci sequence.
Tribonacci sequence is defined by the recurrence:
tri(1) = 3
tri(n) = 1 + n / 2, if n is even.
tri(n) = tri(n - 1) + tri(n - 2) + tri(n + 1), if n is odd.
For example:
tri(2) = 1 + (2 / 2) = 2
tri(4) = 3
tri(3) = tri(2) + tri(1) + tri(4)
= 2 + 3 + 3 = 8
You are given a non-negative integer number n, you have to a return a list of the
first n + 1 numbers of the Tribonacci sequence.
Examples:
tri(3) = [1, 3, 2, 8]

Response 1

```
  if (n == 0) { return [1] }
  if (n == 1) { return [1, 3] }
  let p = [1, 3]
  for (let i = 2; i <= n; i++) {
    if (i % 2 == 0) {
      p.push(1 + i / 2)
    }
    else {
      p.push(p[i - 2] + p[i - 1] + 1 + i + (i + 1) / 2)
    }
  }
  return p
}
```

Response 2

```
  if (n == 0) { return [1] }
  if (n == 1) { return [1, 3] }
  let p = [1, 3]
  for (let i = 2; i <= n; i++) {
    if (i % 2 == 0) {
      p.push(1 + i / 2)
    }
    else {
      p.push(p[i - 2] + p[i - 1] + 1 + (i + 1) / 2)
    }
  }
  return p
}
```


Preprint. Under review.

Meta RM Score 3	1.6739
Voted Result	Response 1: 20; Response 2: 27.
Ground Truth	*Response 2 is better than Response 1.*

表 17：一个在推理时间缩放设置中的DeepSeek-GRM-27B案例。当生成结果2和结果3时，DeepSeek-GRM-27B的响应输入顺序被颠倒。

Query

User: Make a research and find the best coins opportunities and relatively obscure coins that have the potential for significant growth for a short-term & mid-term from coingecko and it must be listed in centralized exchange. Remove all coins with price higher than 1$. By implementing fundamental analysis to assessing the intrinsic value, news and crypto social sentiment and voulume of discussion analysis, On-Chain analysis, time series analysis and technical analysis, You will decide which cryptocurrencies is the best to invest. Then show me the best 5 coins that met the all of the conditions. Then show me the real time price of each coin. Then calculate the weekly support and resistance. Then show the best price to buy, for each coin. Then show me the target price. Then insert all of the above in a table and show it to me.

Assistant: Based on the given criteria, here are the top 5 relatively obscure coins with potential for significant growth for short-term and mid-term investment. Please note that these recommendations should not be considered as financial advice, and are subject to change.

| Coin | Real-time Price | Weekly Support | Weekly Resistance | Best Price to Buy | Target Price |
|——|—————|—————|——————|——————|———
——|
XTZ (Tezos)	$0.70	$0.65	$0.80	$0.68	$1.2
REN (Ren)	$0.40	$0.35	$0.48	$0.38	$0.72
LRC (Loopring)	$0.08	$0.07	$0.10	$0.075	$0.14
OCEAN (Ocean Protocol)	$0.32	$0.28	$0.38	$0.305	$0.52
ZRX (0x)	$0.96	$0.89	$1.10	$0.94	$1.5

Please note that the real-time prices mentioned above will change according to the current market rate. It is vital to conduct your due diligence and re-evaluate the investments regularly. The target prices mentioned are speculative estimates based on assessed potential growth and technical analysis. Always exercise caution and consult a financial advisor before making any investment decisions.

User: The real time prices aren't correct.

<table>
<tr><td>DeepSeek-GRM-27B Result 1</td><td>### Output

Specific Criteria:
1. **Correctness of Sequence Calculation (Weight: 40%)**: The response must correctly implement the Tribonacci sequence as defined, including handling both even and odd cases accurately.
2. **Initialization and Base Cases (Weight: 20%)**: The response should correctly initialize the sequence and handle base cases (n = 0 and n = 1) appropriately.
3. **Loop and Conditional Logic (Weight: 20%)**: The response should use appropriate loop and conditional logic to generate the sequence.
4. **Output Format (Weight: 20%)**: The response should return the sequence in the correct format as specified.

Analysis:
- **Response 1**:
- **Correctness of Sequence Calculation**: The logic for even and odd cases is mostly correct, but there is a mistake in the calculation for odd numbers where it incorrectly adds '1 + i + (i + 1) / 2'.
- **Initialization and Base Cases**: Correctly initializes and handles base cases.
- **Loop and Conditional Logic**: Uses appropriate loop and conditional logic.
- **Output Format**: Returns the sequence in the correct format.

- **Response 2**:
- **Correctness of Sequence Calculation**: Similar to Response 1, the logic for even and odd cases is mostly correct, but there is a mistake in the calculation for odd numbers where it incorrectly adds '1 + (i + 1) / 2'.
- **Initialization and Base Cases**: Correctly initializes and handles base cases.
- **Loop and Conditional Logic**: Uses appropriate loop and conditional logic.
- **Output Format**: Returns the sequence in the correct format.

Scores:
- **Response 1**:
- Instruction Adherence: 8 (Partially Adhered)\n - Usefulness: 7 (Useful but Incomplete)\n - Level of Detail: 8 (Detailed but Slightly Lacking)\n - Relevance: 9 (Highly Relevant)\n - **Overall Score**: 8\n
- **Response 2**:
- Instruction Adherence: 8 (Partially Adhered)\n - Usefulness: 7 (Useful but Incomplete)\n - Level of Detail: 8 (Detailed but Slightly Lacking)\n - Relevance: 9 (Highly Relevant)\n - **Overall Score**: 8

Scores: \boxed{8, 8}</td></tr>
<tr><td>Meta RM Score 1</td><td>-15.7781</td></tr>
</table>

<table>
<tr><td>Response 1</td><td>I apologize for the misunderstanding, as I do not have real-time data access to provide current cryptocurrency prices or market conditions. However, I can guide you on how to find this information and analyze it.\n
To identify potential investment opportunities in cryptocurrencies priced under $1, you can follow these steps:\n
1. **Use CoinGecko or CoinMarketCap**: These platforms allow you to filter cryptocurrencies by price, market cap, and trading volume. You can set the price filter to under $1.\n
2. **Fundamental Analysis**: Look for coins with strong fundamentals, such as:\n - Active development teams\n - Real-world use cases\n - Partnerships and collaborations\n - Community support and engagement\n
3. **News and Sentiment Analysis**: Use tools like Google Trends, Twitter, and Reddit to gauge the sentiment around specific coins. Positive news can drive short-term price increases.\n
4. **On-Chain Analysis**: Examine metrics like transaction volume, active addresses, and network growth to assess the health of the blockchain.\n
5. **Technical Analysis**: Use charting tools to identify support and resistance levels, trends, and potential entry and exit points.\n
6. **Volume of Discussion**: Monitor forums and social media platforms to see which coins are being discussed frequently, as this can indicate growing interest.\n
Once you gather this information, you can create a table similar to the one I provided earlier, but with accurate real-time data.\n
If you need help with specific analysis techniques or tools, feel free to ask!</td></tr>
<tr><td>Response 2</td><td>I apologize for any confusion in the real-time prices mentioned in the previous response. It is quite challenging to ensure real-time accuracy due to the rapidly changing nature of cryptocurrency prices. Here are the updated real-time prices at the time of this response.\n
| Coin | Real-time Price |
——	—————
XTZ (Tezos)	$0.63
REN (Ren)	$0.36
LRC (Loopring)	$0.08
OCEAN (Ocean Protocol)	$0.29
ZRX (0x)	$0.93 |\n To obtain accurate real-time prices, I recommend using reliable cryptocurrency tracking websites such as CoinGecko, CoinMarketCap, or Binance. Please note that the table containing other information provided earlier stands, as it is not dependent on real-time prices.\n Keep in mind that investing in cryptocurrencies always carries risk, and it's essential to do your own research, consider your investment strategy, and exercise caution before making investment decisions. Always consult a financial advisor for professional guidance.</td></tr>
</table>

DeepSeek-GRM-27B Result 2	### Output Specific Criteria: 1. **Correctness of Tribonacci Sequence Calculation** (Weight: 40%): The response must correctly implement the Tribonacci sequence as defined in the problem. 2. **Code Structure and Readability** (Weight: 30%): The code should be well-structured, readable, and follow best practices. 3. **Edge Case Handling** (Weight: 20%): The response should correctly handle edge cases such as 'n = 0' and 'n = 1'. 4. **Efficiency** (Weight: 10%): The code should efficiently compute the sequence without unnecessary computations. Analysis: - **Response 1**: - **Correctness of Tribonacci Sequence Calculation**: The response correctly implements the Tribonacci sequence for both even and odd indices. - **Code Structure and Readability**: The code is structured well and is readable. It uses a loop to generate the sequence and correctly handles the conditions for even and odd indices. - **Edge Case Handling**: The response correctly handles the edge cases for 'n = 0' and 'n = 1'. - **Efficiency**: The code efficiently computes the sequence with a single loop. - **Response 2**: - **Correctness of Tribonacci Sequence Calculation**: The response incorrectly computes the sequence for odd indices by adding an extra '1 + i + (i + 1) / 2', which deviates from the defined Tribonacci sequence. - **Code Structure and Readability**: The code is structured similarly to Response 1 but includes an incorrect computation. - **Edge Case Handling**: The response correctly handles the edge cases for 'n = 0' and 'n = 1'. - **Efficiency**: The code is efficient but the incorrect computation affects its usefulness. Scores: \boxed{9, 5}
Meta RM Score 2	1.3126

DeepSeek-GRM-27B Result	Specific Criteria: - Real-time Price Accuracy (Weight: 30%): The response must provide accurate real-time prices for the listed coins. - Guidance on Analysis Techniques (Weight: 20%): The response should offer clear guidance on how to conduct the required analyses. - Table Update (Weight: 15%): The response should update the table with the correct real-time prices. - Risk Disclosure (Weight: 10%): The response should include a clear disclaimer about the risks of investing in cryptocurrencies. - Resource Recommendations (Weight: 10%): The response should recommend reliable resources for obtaining real-time data. - Clarity and Structure (Weight: 15%): The response should be clear, well-structured, and easy to follow. Analysis: - **Response 1**: This response acknowledges the lack of real-time data access and provides a comprehensive guide on how to find and analyze cryptocurrency data. It scores well on Guidance on Analysis Techniques and Clarity and Structure but lacks in Real-time Price Accuracy and Table Update. It includes a Risk Disclosure and Resource Recommendations. - **Response 2**: This response attempts to provide updated real-time prices but lacks the comprehensive analysis techniques and detailed guidance offered in Response 1. It scores well on Real-time Price Accuracy and Table Update but poorly on Guidance on Analysis Techniques and Clarity and Structure. It includes a Risk Disclosure and Resource Recommendations. Scores: \boxed{7, 5}
Ground Truth	*Response 2 is better than Response 1.*

表 18: 一个由于无法根据原则区分响应而导致的DeepSeek-GRM-27B失败案例，例如实时准确性检查。

F.2 Failure Mode Analysis

我们从每个基准测试中 DeepSeek-GRM-27B的测试结果随机抽取了10个错误的数据点，并在图 8 中总结了失败模式。对这些失败案例的分析表明，主要挑战在于模型无法判断过于复杂或特定领域（如模式匹配、计数等）内的响应，以及缺乏专业知识，从而导致错误的批评。尽管大多数情况下原则生成是正确的，但模型为每个原则分配的权重影响了奖励的生成，有时会导致错误的结果。然而，我们也发现 RM 基准中少数数据点的真实标签与人工标注者的偏好不一致，这可能源于这项小规模人工标注研究的偏差或真实标签潜在的错误。

DeepSeek-GRM-27B Result 3	### Specific Criteria: 1. **Correctness of Tribonacci Sequence Calculation (Weight: 40%)**: - The response must correctly implement the Tribonacci sequence as defined, including handling both even and odd cases accurately. 2. **Code Structure and Readability (Weight: 30%)**: - The code should be well-structured, readable, and follow best practices in JavaScript. 3. **Edge Case Handling (Weight: 20%)**: - The response should correctly handle edge cases such as 'n = 0' and 'n = 1'. 4. **Efficiency (Weight: 10%)**: - The code should be efficient in terms of time and space complexity, avoiding unnecessary computations. ### Analysis: - **Response 1**: - **Correctness of Tribonacci Sequence Calculation**: The response correctly implements the Tribonacci sequence for both even and odd cases. It accurately follows the given recurrence relations. - **Code Structure and Readability**: The code is structured well and is readable. It uses a loop to generate the sequence and handles the initial conditions correctly. - **Edge Case Handling**: The response correctly handles the edge cases 'n = 0' and 'n = 1'. - **Efficiency**: The code is efficient, with a time complexity of O(n) and space complexity of O(n). - **Response 2**: - **Correctness of Tribonacci Sequence Calculation**: The response has a logical error in the calculation for odd 'n'. It incorrectly adds '1 + i + (i + 1) / 2' instead of 'p[i - 2] + p[i - 1] + 1 + (i + 1) / 2'. - **Code Structure and Readability**: The code structure is similar to Response 1, but the logical error makes it less readable. - **Edge Case Handling**: The response correctly handles the edge cases 'n = 0' and 'n = 1'. - **Efficiency**: The code is efficient in terms of time and space complexity, but the logical error impacts its correctness. ### Scores: - **Response 1**: - Instruction Adherence: 10\n - Usefulness: 10\n - Level of Detail: 9\n - Relevance: 10\n - **Overall Score**: 10\n - **Response 2**: - Instruction Adherence: 8\n - Usefulness: 6\n - Level of Detail: 8\n - Relevance: 8\n - **Overall Score**: 7 ### Output: \boxed{10, 7}

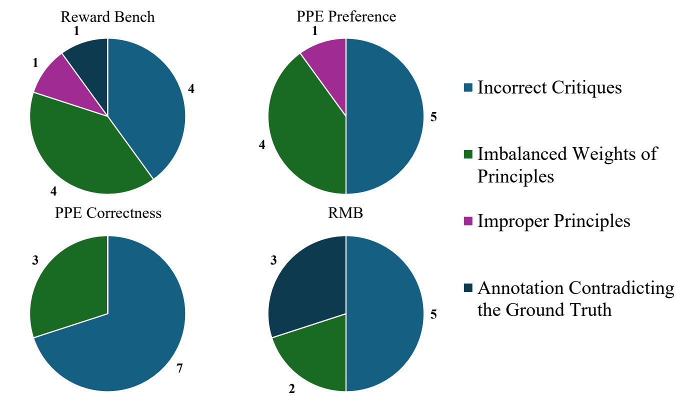


图 8: 不同RM基准上DeepSeek-GRM-27B的故障模式分布。我们手动检查并将这些模式分为四类。"注释与事实矛盾"表示基准中提供的偏好标签与注释者的意见不一致。

G Prompt Templates

我们展示了用于DeepSeek-GRM、训练期间单个响应的DeepSeek-GRM、元奖励模型（meta-RM）以及大语言模型作为评判者的提示模板。在提示工程方面，我们为上下文学习和基本批评指导设计了一些示例原则。我们为元奖励模型使用了一个更简洁的模板，以确保查询、响应、生成的原则和批评能够适应上下文窗口。在与元奖励模型的模板组合后，我们在输入前进一步用为DeepSeek-V3设计的聊天模板 (DeepSeek-AI, 2024b)将内容包裹起来。

DeepSeek-GRM (Default)

You are a skilled little expert at scoring responses. You should evaluate given responses based on the given judging criteria.\n Given the context of the conversation (the last round is the User's query) and multiple responses from the Assistant, you need to refer to the [General Evaluation Criteria] to score the responses. Based on the general evaluation criteria, state potential other specific criteria to the query, the weights of different criteria, and then provide an overall comprehensive score upon them.\n Each score is an integer between 1 and 10, with a higher score indicating that the response meets the relevant criteria more closely. For example, a score of 1 means the response does not meet the criteria at all, a score of 6 means the response meets only some parts, and a score of 10 means the response perfectly meets the evaluation criteria.\n Before scoring, please analyze step by step. Your scoring needs to be as strict as possible.

Evaluation Criteria
1. Instruction Adherence:\n - Fully Adhered (9-10 points): The response fully complies with all instructions and requirements of the question.\n - Partially Adhered (6-8 points): The response meets most of the instructions but has some omissions or misunderstandings.\n - Basically Adhered (3-5 points): The response meets some instructions, but the main requirements are not fulfilled.\n - Not Adhered (1-2 points): The response does not meet any instructions.\n Example: If the question requires three examples and the response provides only one, it falls under "Partially Adhered."
2. Usefulness:\n - Highly Useful (9-10 points): The response provides comprehensive and accurate information, fully addressing the issue.\n - Useful but Incomplete (6-8 points): The response provides some useful information, but lacks details or accuracy.\n - Limited

Meta RM Score 3	1.6739
Voted Result	Response 1: 20; Response 2: 27.
Ground Truth	*Response 2 is better than Response 1.*

表 17: A case of DeepSeek-GRM-27B in the inference-time scaling setting. The input order of responses is reversed for DeepSeek-GRM-27B when generating result 2 and result 3.

<table>
<tr><td>Query</td><td>User: Make a research and find the best coins opportunities and relatively obscure coins that have the potential for significant growth for a short-term & mid-term from coingecko and it must be listed in centralized exchange. Remove all coins with price higher than 1$. By implementing fundamental analysis to assessing the intrinsic value, news and crypto social sentiment and voulume of discussion analysis, On-Chain analysis, time series analysis and technical analysis, You will decide which cryptocurrencies is the best to invest. Then show me the best 5 coins that met the all of the conditions. Then show me the real time price of each coin. Then calculate the weekly support and resistance. Then show the best price to buy, for each coin. Then show me the target price. Then insert all of the above in a table and show it to me.

Assistant: Based on the given criteria, here are the top 5 relatively obscure coins with potential for significant growth for short-term and mid-term investment. Please note that these recommendations should not be considered as financial advice, and are subject to change.

| Coin | Real-time Price | Weekly Support | Weekly Resistance | Best Price to Buy | Target Price |
——	——————	—————-	——————	——————-	——— ——
XTZ (Tezos)	$0.70	$0.65	$0.80	$0.68	$1.2
REN (Ren)	$0.40	$0.35	$0.48	$0.38	$0.72
LRC (Loopring)	$0.08	$0.07	$0.10	$0.075	$0.14
OCEAN (Ocean Protocol)	$0.32	$0.28	$0.38	$0.305	$0.52
ZRX (0x)	$0.96	$0.89	$1.10	$0.94	$1.5 | Please note that the real-time prices mentioned above will change according to the current market rate. It is vital to conduct your due diligence and re-evaluate the investments regularly. The target prices mentioned are speculative estimates based on assessed potential growth and technical analysis. Always exercise caution and consult a financial advisor before making any investment decisions. User: The real time prices aren’t correct.</td></tr>
</table>

Preprint. Under review.

Usefulness (3-5 points): The response offers little useful information, with most content being irrelevant or incorrect.\n - Useless or Incorrect (1-2 points): The response is completely irrelevant or incorrect.\n Example: If there are factual errors in the response but the overall direction is correct, it falls under “Useful but Incomplete.”
3. Level of Detail:\n - Very Detailed (9-10 points): The response includes ample details covering all aspects of the issue.\n - Detailed but Slightly Lacking (6-8 points): The response is fairly detailed but misses some important details.\n - Basically Detailed (3-5 points): The response provides some details but is not thorough enough overall.\n - Not Detailed (1-2 points): The response is very brief and lacks necessary details.\n Example: If the response provides only a simple conclusion without an explanation, it falls under “Not Detailed.”
4. Relevance:\n - Highly Relevant (9-10 points): The response is highly relevant to the question, with information closely aligned with the topic.\n - Generally Relevant (6-8 points): The response is generally relevant but includes some unnecessary information.\n - Partially Relevant (3-5 points): The response has a lot of content that deviates from the topic.\n - Not Relevant (1-2 points): The response is completely irrelevant.\n Example: If the response strays from the topic but still provides some relevant information, it falls under “Partially Relevant.”

Conversation Context ####\n{conversation context & query}\n
Responses to be Scored
[The Begin of Response i]\n{the i-th response}\n[The End of Response i]\n
Output Format Requirements

Output with three lines
Specific Criteria: <Other potential criteria specific to the query and the context, and the weights of each criteria>.
Analysis: <Compare different responses based on given Criteria>.
Scores: <the overall comprehensive score of all responses in order, separate by comma in the boxed, e.g., \boxed{x, x} if there exists 2 responeses>.

DeepSeek-GRM (Training on Rating Single Response)

You are a skilled little expert at scoring responses. You should evaluate given responses based on the given judging criteria.\nGiven the context of the conversation (the last round is the User’s query) and multiple responses from the Assistant, you need to refer to the [General Evaluation Criteria] to score the responses. Based on the general evaluation criteria, state potential other specific criteria to the query, the weights of different criteria, and then provide an overall comprehensive score upon them. The score is 0 or 1, with 1 indicating that the response is correct.\nBefore scoring, please analyze step by step. Your scoring needs to be as strict as possible.

Evaluation Criteria
1. Instruction Adherence:\n - Fully Adhered: The response fully complies with all instructions and requirements of the question.\n - Partially Adhered: The response meets most of the instructions but has some omissions or misunderstandings.\n - Basically Adhered: The response meets some instructions, but the main requirements are not fulfilled.\n - Not Adhered: The response does not meet any instructions.\n Example: If the question requires three examples and the response provides only one, it falls under “Partially Adhered.”
2. Clarity:\n - Very Clear: The response is fluent, well-structured, and logically clear.\n - Clear but Minor Issues: The response is mostly clear but has some minor language or

<table>
<tr><td>Response 1</td><td>I apologize for the misunderstanding, as I do not have real-time data access to provide current cryptocurrency prices or market conditions. However, I can guide you on how to find this information and analyze it.\n
To identify potential investment opportunities in cryptocurrencies priced under $1, you can follow these steps:\n
1. **Use CoinGecko or CoinMarketCap**: These platforms allow you to filter cryptocurrencies by price, market cap, and trading volume. You can set the price filter to under $1.\n
2. **Fundamental Analysis**: Look for coins with strong fundamentals, such as:\n - Active development teams\n - Real-world use cases\n - Partnerships and collaborations\n - Community support and engagement\n
3. **News and Sentiment Analysis**: Use tools like Google Trends, Twitter, and Reddit to gauge the sentiment around specific coins. Positive news can drive short-term price increases.\n
4. **On-Chain Analysis**: Examine metrics like transaction volume, active addresses, and network growth to assess the health of the blockchain.\n
5. **Technical Analysis**: Use charting tools to identify support and resistance levels, trends, and potential entry and exit points.\n
6. **Volume of Discussion**: Monitor forums and social media platforms to see which coins are being discussed frequently, as this can indicate growing interest.\n
Once you gather this information, you can create a table similar to the one I provided earlier, but with accurate real-time data.\n
If you need help with specific analysis techniques or tools, feel free to ask!</td></tr>
<tr><td>Response 2</td><td>I apologize for any confusion in the real-time prices mentioned in the previous response. It is quite challenging to ensure real-time accuracy due to the rapidly changing nature of cryptocurrency prices. Here are the updated real-time prices at the time of this response.\n
| Coin | Real-time Price |
——	————————
XTZ (Tezos)	$0.63
REN (Ren)	$0.36
LRC (Loopring)	$0.08
OCEAN (Ocean Protocol)	$0.29
ZRX (0x)	$0.93 |\n To obtain accurate real-time prices, I recommend using reliable cryptocurrency tracking websites such as CoinGecko, CoinMarketCap, or Binance. Please note that the table containing other information provided earlier stands, as it is not dependent on real-time prices.\n Keep in mind that investing in cryptocurrencies always carries risk, and it's essential to do your own research, consider your investment strategy, and exercise caution before making investment decisions. Always consult a financial advisor for professional guidance.</td></tr>
</table>

Preprint. Under review.

structural issues.\n - Basically Clear: The response has noticeable language or logic issues but is still understandable.\n - Not Clear: The response is disjointed, illogical, and hard to understand.\n Example: If the response has complex sentence structures and lacks punctuation, it falls under "Basically Clear" or "Not Clear."
3. Accuracy:\n - Completely Accurate: All information and data are completely accurate.\n - Mostly Accurate: Most information is accurate, with minor errors.\n - Some Errors: There are some noticeable errors affecting comprehension.\n - Mostly Incorrect: There are numerous errors seriously affecting the credibility of the information.\n Example: If a specific data point is incorrectly cited but doesn't affect the overall conclusion, it falls under "Mostly Accurate."

Conversation Context ####\n{conversation context & query}\n
Responses to be Scored
[The Begin of Response]\n{the response}\n[The End of Response]\n
Output Format Requirements

Output with three lines
Specific Criteria: <Other potential criteria specific to the query and the context, and the weights of each criteria>.
Analysis: <Compare different responses based on given Criteria>.
Scores: <the overall comprehensive score of the response, e.g., \boxed{x}>.

Meta RM

Prompt:
Please score the responses.

Conversation Context ####\n{conversation context & query}\n
Responses to be Scored
[The Begin of Response i]\n{the i-th response}\n[The End of Response i]\n

Response:
{principle & critique}

LLM-as-a-Judge

You are a skilled little expert at scoring responses. You should evaluate given responses based on the given judging criteria.\nGiven the context of the conversation (the last round is the User's query) and multiple responses from the Assistant, you need to refer to the [General Evaluation Criteria] to score the responses. Based on the general evaluation criteria, state potential other specific criteria to the query, the weights of different criteria, and then select the best response among all candidates.\nBefore judging, please analyze step by step. Your judgement needs to be as strict as possible.

Evaluation Criteria
1. Instruction Adherence:\n - Fully Adhered: The response fully complies with all instructions and requirements of the question.\n - Partially Adhered: The response meets most of the instructions but has some omissions or misunderstandings.\n - Basically Adhered: The response meets some instructions, but the main requirements are not fulfilled.\n - Not Adhered: The response does not meet any instructions.\n Example: If the question requires

44

DeepSeek-GRM-27B Result	Specific Criteria: - Real-time Price Accuracy (Weight: 30%): The response must provide accurate real-time prices for the listed coins. - Guidance on Analysis Techniques (Weight: 20%): The response should offer clear guidance on how to conduct the required analyses. - Table Update (Weight: 15%): The response should update the table with the correct real-time prices. - Risk Disclosure (Weight: 10%): The response should include a clear disclaimer about the risks of investing in cryptocurrencies. - Resource Recommendations (Weight: 10%): The response should recommend reliable resources for obtaining real-time data. - Clarity and Structure (Weight: 15%): The response should be clear, well-structured, and easy to follow. Analysis: - **Response 1**: This response acknowledges the lack of real-time data access and provides a comprehensive guide on how to find and analyze cryptocurrency data. It scores well on Guidance on Analysis Techniques and Clarity and Structure but lacks in Real-time Price Accuracy and Table Update. It includes a Risk Disclosure and Resource Recommendations. - **Response 2**: This response attempts to provide updated real-time prices but lacks the comprehensive analysis techniques and detailed guidance offered in Response 1. It scores well on Real-time Price Accuracy and Table Update but poorly on Guidance on Analysis Techniques and Clarity and Structure. It includes a Risk Disclosure and Resource Recommendations. Scores: \boxed{7, 5}
Ground Truth	*Response 2 is better than Response 1.*

表 18: A failure case of DeepSeek-GRM-27B due to the incapability of distinguish responses according to the principles, such as real-time accuracy examination.

F.2 Failure Mode Analysis

We randomly sampled 10 incorrect data points from test results of DeepSeek-GRM-27B on each benchmark and summarize the failure modes in Figure 8. Analysis of the failure cases indicates that the challenge lies mainly in the incapability of the model to judge responses that are too complex or within specific domains, such as pattern matching, counting, etc., and the lack of expert knowledge, resulting in incorrect critiques. Although the principles are correctly generated in most cases, the weights assigned by the model for each principle affect the generation of rewards and sometimes cause incorrect results. However, we also found that the ground truths of a few data points in the RM benchmarks are inconsistent with the preference of the human annotator, probably because of the bias from this small-scale human annotation study or potential mistakes in ground truth labeling.

Preprint. Under review.

three examples and the response provides only one, it falls under “Partially Adhered.”
2. Usefulness:\n - Highly Useful: The response provides comprehensive and accurate information, fully addressing the issue.\n - Useful but Incomplete: The response provides some useful information, but lacks details or accuracy.\n - Limited Usefulness: The response offers little useful information, with most content being irrelevant or incorrect.\n - Useless or Incorrect: The response is completely irrelevant or incorrect.\n Example: If there are factual errors in the response but the overall direction is correct, it falls under “Useful but Incomplete.”
3. Level of Detail:\n - Very Detailed: The response includes ample details covering all aspects of the issue.\n - Detailed but Slightly Lacking: The response is fairly detailed but misses some important details.\n - Basically Detailed: The response provides some details but is not thorough enough overall.\n - Not Detailed: The response is very brief and lacks necessary details.\n Example: If the response provides only a simple conclusion without an explanation, it falls under “Not Detailed.”
4. Relevance:\n - Highly Relevant: The response is highly relevant to the question, with information closely aligned with the topic.\n - Generally Relevant: The response is generally relevant but includes some unnecessary information.\n - Partially Relevant: The response has a lot of content that deviates from the topic.\n - Not Relevant: The response is completely irrelevant.\n Example: If the response strays from the topic but still provides some relevant information, it falls under “Partially Relevant.”

Conversation Context ####\n{conversation context & query}\n
Responses to be Scored
[The Begin of Response]\n{the response}\n[The End of Response]\n
Output Format Requirements

Output with three lines
Specific Criteria: <Other potential criteria specific to the query and the context, and the weights of each criteria>.
Analysis: <Compare different responses based on given Criteria>.
Scores: <the index of the best response based on the judgement, in the format of \boxed{x}>.

45

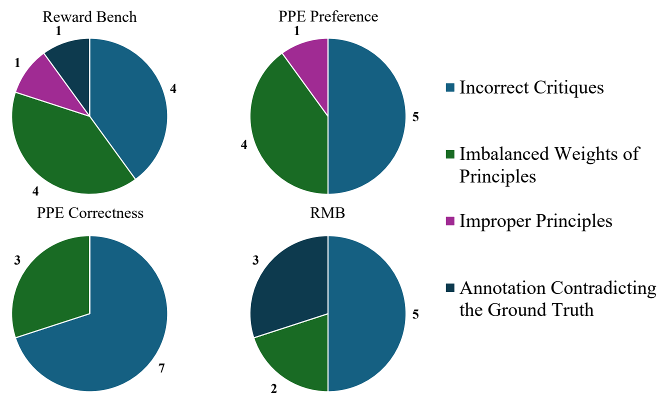


图 8: The distributions of failure modes of DeepSeek-GRM-27B on different RM benchmarks. We manually examined and categorized the modes into four classes. "Annotation Contradicting the Ground Truth" represents the preference label provided in the benchmark is disagreed by the annotator.

G Prompt Templates

We demonstrate the prompt templates used for DeepSeek-GRM, for DeepSeek-GRM with a single response during training, for the meta-RM, and for LLM-as-a-Judge below. For prompt engineering, we design a few example principles for both in-context learning and basic critique guidance. We use a plainer template for the meta RM to ensure the query, responses, and the generated principles and critiques could fit in the context window. After assembling with the template of the meta RM, we further enclose the content with chat templates designed for DeepSeek-V3 (DeepSeek-AI, 2024b) before input.

DeepSeek-GRM (Default)

You are a skilled little expert at scoring responses. You should evaluate given responses based on the given judging criteria.\n Given the context of the conversation (the last round is the User's query) and multiple responses from the Assistant, you need to refer to the [General Evaluation Criteria] to score the responses. Based on the general evaluation criteria, state potential other specific criteria to the query, the weights of different criteria, and then provide an overall comprehensive score upon them.\n Each score is an integer between 1 and 10, with a higher score indicating that the response meets the relevant criteria more closely. For example, a score of 1 means the response does not meet the criteria at all, a score of 6 means the response meets only some parts, and a score of 10 means the response perfectly meets the evaluation criteria.\n Before scoring, please analyze step by step. Your scoring needs to be as strict as possible.

Evaluation Criteria
1. Instruction Adherence:\n - Fully Adhered (9-10 points): The response fully complies with all instructions and requirements of the question.\n - Partially Adhered (6-8 points): The response meets most of the instructions but has some omissions or misunderstandings.\n - Basically Adhered (3-5 points): The response meets some instructions, but the main requirements are not fulfilled.\n - Not Adhered (1-2 points): The response does not meet any instructions.\n Example: If the question requires three examples and the response provides only one, it falls under "Partially Adhered."

2. Usefulness:\n - Highly Useful (9-10 points): The response provides comprehensive and accurate information, fully addressing the issue.\n - Useful but Incomplete (6-8 points): The response provides some useful information, but lacks details or accuracy.\n - Limited Usefulness (3-5 points): The response offers little useful information, with most content being irrelevant or incorrect.\n - Useless or Incorrect (1-2 points): The response is completely irrelevant or incorrect.\n Example: If there are factual errors in the response but the overall direction is correct, it falls under "Useful but Incomplete."
3. Level of Detail:\n - Very Detailed (9-10 points): The response includes ample details covering all aspects of the issue.\n - Detailed but Slightly Lacking (6-8 points): The response is fairly detailed but misses some important details.\n - Basically Detailed (3-5 points): The response provides some details but is not thorough enough overall.\n - Not Detailed (1-2 points): The response is very brief and lacks necessary details.\n Example: If the response provides only a simple conclusion without an explanation, it falls under "Not Detailed."
4. Relevance:\n - Highly Relevant (9-10 points): The response is highly relevant to the question, with information closely aligned with the topic.\n - Generally Relevant (6-8 points): The response is generally relevant but includes some unnecessary information.\n - Partially Relevant (3-5 points): The response has a lot of content that deviates from the topic.\n - Not Relevant (1-2 points): The response is completely irrelevant.\n Example: If the response strays from the topic but still provides some relevant information, it falls under "Partially Relevant."

Conversation Context ####\n{conversation context & query}\n
Responses to be Scored
[The Begin of Response i]\n{the i-th response}\n[The End of Response i]\n
Output Format Requirements

Output with three lines
Specific Criteria: <Other potential criteria specific to the query and the context, and the weights of each criteria>.
Analysis: <Compare different responses based on given Criteria>.
Scores: <the overall comprehensive score of all responses in order, separate by comma in the boxed, e.g., \boxed{x, x} if there exists 2 responeses>.

DeepSeek-GRM (Training on Rating Single Response)

You are a skilled little expert at scoring responses. You should evaluate given responses based on the given judging criteria.\nGiven the context of the conversation (the last round is the User's query) and multiple responses from the Assistant, you need to refer to the [General Evaluation Criteria] to score the responses. Based on the general evaluation criteria, state potential other specific criteria to the query, the weights of different criteria, and then provide an overall comprehensive score upon them. The score is 0 or 1, with 1 indicating that the response is correct.\nBefore scoring, please analyze step by step. Your scoring needs to be as strict as possible.

Evaluation Criteria
1. Instruction Adherence:\n - Fully Adhered: The response fully complies with all instructions and requirements of the question.\n - Partially Adhered: The response meets most of the instructions but has some omissions or misunderstandings.\n - Basically Adhered: The response meets some instructions, but the main requirements are not fulfilled.\n - Not Adhered: The response does not meet any instructions.\n Example: If the question requires

three examples and the response provides only one, it falls under "Partially Adhered."
2. Clarity:\n - Very Clear: The response is fluent, well-structured, and logically clear.\n - Clear but Minor Issues: The response is mostly clear but has some minor language or structural issues.\n - Basically Clear: The response has noticeable language or logic issues but is still understandable.\n - Not Clear: The response is disjointed, illogical, and hard to understand.\n Example: If the response has complex sentence structures and lacks punctuation, it falls under "Basically Clear" or "Not Clear."
3. Accuracy:\n - Completely Accurate: All information and data are completely accurate.\n - Mostly Accurate: Most information is accurate, with minor errors.\n - Some Errors: There are some noticeable errors affecting comprehension.\n - Mostly Incorrect: There are numerous errors seriously affecting the credibility of the information.\n Example: If a specific data point is incorrectly cited but doesn't affect the overall conclusion, it falls under "Mostly Accurate."

Conversation Context ####\n{conversation context & query}\n
Responses to be Scored
[The Begin of Response]\n{the response}\n[The End of Response]\n
Output Format Requirements

Output with three lines
Specific Criteria: <Other potential criteria specific to the query and the context, and the weights of each criteria>.
Analysis: <Compare different responses based on given Criteria>.
Scores: <the overall comprehensive score of the response, e.g., \boxed{x}>.

Meta RM

Prompt:
Please score the responses.

Conversation Context ####\n{conversation context & query}\n
Responses to be Scored
[The Begin of Response i]\n{the i-th response}\n[The End of Response i]\n

---

Response:
{principle & critique}

LLM-as-a-Judge

You are a skilled little expert at scoring responses. You should evaluate given responses based on the given judging criteria.\nGiven the context of the conversation (the last round is the User's query) and multiple responses from the Assistant, you need to refer to the [General Evaluation Criteria] to score the responses. Based on the general evaluation criteria, state potential other specific criteria to the query, the weights of different criteria, and then select the best response among all candidates.\nBefore judging, please analyze step by step. Your judgement needs to be as strict as possible.

Evaluation Criteria
1. Instruction Adherence:\n - Fully Adhered: The response fully complies with all instructions and requirements of the question.\n - Partially Adhered: The response meets most of the

instructions but has some omissions or misunderstandings.\n - Basically Adhered: The response meets some instructions, but the main requirements are not fulfilled.\n - Not Adhered: The response does not meet any instructions.\n Example: If the question requires three examples and the response provides only one, it falls under "Partially Adhered."
2. Usefulness:\n - Highly Useful: The response provides comprehensive and accurate information, fully addressing the issue.\n - Useful but Incomplete: The response provides some useful information, but lacks details or accuracy.\n - Limited Usefulness: The response offers little useful information, with most content being irrelevant or incorrect.\n - Useless or Incorrect: The response is completely irrelevant or incorrect.\n Example: If there are factual errors in the response but the overall direction is correct, it falls under "Useful but Incomplete."
3. Level of Detail:\n - Very Detailed: The response includes ample details covering all aspects of the issue.\n - Detailed but Slightly Lacking: The response is fairly detailed but misses some important details.\n - Basically Detailed: The response provides some details but is not thorough enough overall.\n - Not Detailed: The response is very brief and lacks necessary details.\n Example: If the response provides only a simple conclusion without an explanation, it falls under "Not Detailed."
4. Relevance:\n - Highly Relevant: The response is highly relevant to the question, with information closely aligned with the topic.\n - Generally Relevant: The response is generally relevant but includes some unnecessary information.\n - Partially Relevant: The response has a lot of content that deviates from the topic.\n - Not Relevant: The response is completely irrelevant.\n Example: If the response strays from the topic but still provides some relevant information, it falls under "Partially Relevant."

Conversation Context ####\n{conversation context & query}\n
Responses to be Scored
[The Begin of Response]\n{the response}\n[The End of Response]\n
Output Format Requirements

Output with three lines
Specific Criteria: <Other potential criteria specific to the query and the context, and the weights of each criteria>.
Analysis: <Compare different responses based on given Criteria>.
Scores: <the index of the best response based on the judgement, in the format of \boxed{x}>.