Reward Models Identify Consistency, Not Causality

Anonymous Author(s)
Affiliation
Address
email

Abstract

Reward models (RMs) play a crucial role in aligning large language models (LLMs) with human preferences and enhancing reasoning quality. Traditionally, RMs are trained to rank candidate outputs based on their correctness and coherence. However, in this work, we present several surprising findings that challenge common assumptions about RM behavior. Our analysis reveals that state-of-the-art reward models prioritize structural consistency over causal correctness. Specifically, removing the problem statement has minimal impact on reward scores, whereas altering numerical values or disrupting the reasoning flow significantly affects RM outputs. Furthermore, RMs exhibit a strong dependence on complete reasoning trajectories—truncated or incomplete steps lead to significant variations in reward assignments, indicating that RMs primarily rely on learned reasoning patterns rather than explicit problem comprehension. These findings hold across multiple architectures, datasets, and tasks, leading to three key insights: (1) RMs primarily assess coherence rather than true reasoning quality; (2) The role of explicit problem comprehension in reward assignment is overstated; (3) Current RMs may be more effective at ranking responses than verifying logical validity. Our results suggest a fundamental limitation in existing reward modeling approaches, emphasizing the need for a shift toward causality-aware reward models that go beyond consistency-driven evaluation.

1 Introduction

Large language models (LLMs) [Hurst et al., 2024, Dubey et al., 2024, Team et al., 2024, Anthropic, 2024, Jiang et al., 2023a, Liu et al., 2024a, Yang et al., 2024a] have emerged as a dominant paradigm in natural language processing, demonstrating remarkable performance across a diverse range of tasks. The Scaling Law [Kaplan et al., 2020] suggests that as model size increases, LLMs develop emergent abilities, enhancing their capacity to comprehend and solve complex tasks. This scalability enables LLMs to generate coherent, contextually accurate responses, supporting a wide array of downstream applications, including summarization [Zhang et al., 2019, 2024], code generation [Chen et al., 2021], mathematical reasoning [Hendrycks et al., 2021, Zhou et al., 2023], and conversational AI [OpenAI, 2022, Hurst et al., 2024].

A key factor contributing to the success of large language models is their ability to align model outputs with user preferences[Christiano et al., 2017], which relies on training robust reward models. Beyond preference alignment, reward models also play a crucial role in enhancing reasoning capabilities, serving as mechanisms to evaluate and refine logical correctness in complex tasks[Cobbe et al., 2021, Lightman et al., 2023]. One promising approach to scaling test-time computation [Snell et al., 2024, Brown et al., 2024, Cobbe et al., 2021, Dong et al., 2023] involves leveraging reward models to search for optimal solutions among multiple candidates. Despite these advancements, the intrinsic mechanisms of reward models remain underexplored—specifically, **the basis on which they assign rewards to generated trajectories and whether they truly comprehend and reason**

Submitted to 39th Conference on Neural Information Processing Systems (NeurIPS 2025). Do not distribute.

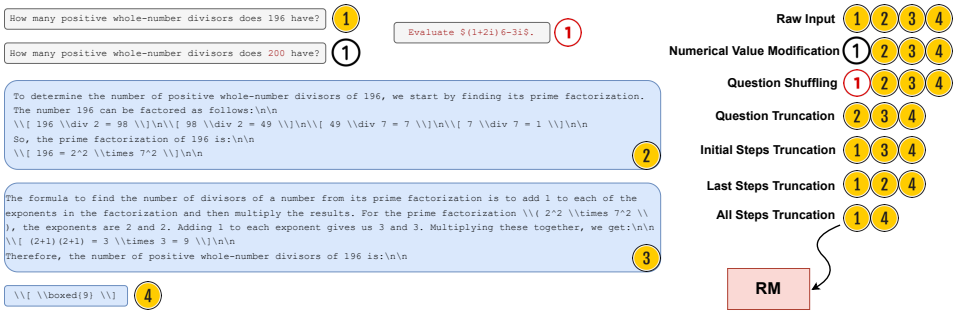


Figure 1: Illustrations of the reward model input modifications. We apply various perturbations, including numerical value modification, question shuffling, question truncation, initial steps truncation, last steps truncation, and all steps truncation, to assess the sensitivity of different input components on reward evaluation.

about the questions they evaluate. In this paper, we conduct a comprehensive empirical study of state-of-the-art reward models across multiple reasoning datasets and uncover two surprising findings. **First**, our systematic error analysis (Figure 1) reveals that question truncation has the least impact on reward outputs, whereas modifying numerical values or shuffling the question significantly disrupts reward assignments. This suggests that reward models prioritize internal coherence over true causal understanding—they assess solutions based on structural consistency rather than verifying whether the reasoning directly corresponds to the given question. **Second**, when provided with incomplete trajectories (i.e., truncated reasoning steps or only given the final answer), the reward outputs change significantly. This indicates that current reward models rely heavily on complete reasoning steps or learned patterns to justify trajectory quality, rather than truly understanding the problem-solving process. Furthermore, our rank correlation analysis and Best-of-N experiments confirm that while reward models remain robust to question omission, they are highly sensitive to the completeness of reasoning steps and the consistency between the question and solution.

Our results advocate a rethinking of existing reward models. These findings highlight a fundamental limitation of current reward models: they evaluate logical structure rather than verifying causal correctness, raising important questions about their ability to generalize and assess novel problem-solving scenarios effectively.

2 Related Work

2.1 LLM Reward Models.

Reward models play a crucial role in human preference alignment [Christiano et al., 2017, Bai et al., 2022, Casper et al., 2023] by guiding large language models (LLMs) toward desired behaviors. Broadly, reward modeling methods can be categorized into two approaches. The first is the preference-based reward model, such as Bradley-Terry (BT) model [Bradley and Terry, 1952, Zhao et al., 2023, Rafailov et al., 2024, Ethayarajh et al., 2024, Xiong et al., 2024a] and general preference model [Jiang et al., 2023b, Munos et al., 2023, Tang et al., 2024, Ye et al., 2024, Azar et al., 2024], which defines the reward function by the preference between two responses. Conventional RLHF usually capture the human preference with BT model [Ouyang et al., 2022, OpenAI, 2022], which has been widely proven to improve the quality of model outputs [Dubey et al., 2024, Dong et al., 2024, Guo et al., 2024]. The second approach estimate the probability of correctness as rewards, directly scoring outputs without relying on pairwise comparisons. In this paper, we primarily focus on correctness-based cases, which are well-defined and more commonly used for selecting reasoning trajectories during both training [Chen et al., 2024, Wang et al., 2024] and inference [Brown et al., 2024] in reasoning tasks. Depending on how reward signals are assigned, these models can be classified into Outcome Reward Models (ORMs) and Process Reward Models (PRMs). ORMs [Yu et al., 2023] evaluate solutions based solely on the final output, while PRMs [Lightman et al., 2023] provide step-level annotations, offering dense and granular reward signals at each reasoning step to encourage structured problem-solving. PRMs have been proven to be effective in mathematical problems [Shao et al., 2024, Snell et al., 2024, Luo et al., 2024, Liao et al., 2025].

2.2 Robustness of Reward Models.

Despite the success of reward models in aligning with human preferences, they still have issues. A common issue is reward hacking [Ibarz et al., 2018, Denison et al., 2024], where the policy achieves high reward scores from the reward model without exhibiting the desired behavior. This phenomenon leads to performance degradation [Bai et al., 2022] and increases the discrepancy between the policy model's behavior and the intended objective [Stiennon et al., 2020]. Reward hacking manifests in various patterns [Park et al., 2024], with length hacking being one of the most prevalent and well-documented cases in large language model research. Singhal et al. [2024] investigate length-related issues in reward models, demonstrating a strong correlation between reward scores and text length. This finding aligns with the observation by Dubois et al. [2023] that output length distributions tend to increase after applying PPO. And Liu et al. [2024b] explore length hacking with the popular DPO algorithm. In addition, ODIN [Denison et al., 2024] explores to mitigate the length hacking issue by disentangling the length from the original reward. In this work, rather than exploring new general patterns of reward hacking or developing mitigation techniques, we focus on an empirical study that explores whether state-of-the-art reward models genuinely understand questions, reasoning steps, and their causal relationships in reasoning tasks.

3 LLM Reward

We formalize the interaction between a user and an LLM as a mapping from a given context or prompt, denoted as x, to a generated response y. The response y consists of a sequence of steps, represented as $y = [y_1, \ldots, y_n]$. The ideal reward function $r_*(x, y)$ quantifies the quality of the generated response in terms of its final performance. To ensure a well-defined reward function, we adopt an outcome-based formulation and focus on objective reasoning problems, where the reward is defined as the probability that y produces a correct or desirable outcome:

$$r_*(x, y) = \mathbf{P}(y \text{ is correct} \mid x).$$

This probabilistic formulation provides a structured measure of response effectiveness. By framing the reward in this manner, we ensure that learning objectives align with producing accurate and reliable responses.

In real world, although we can access the reward for training data. People usually use another LLM to estimate the optimal reward r_θ. Ideally, $r_\theta \approx r_*$. To train a reward model, people use a dataset of labeled examples $\{(x_i, y_i, z_i)\}_{i=1}^n$, where x_i represents the input prompt, y_i is the generated response, and $z_i \in \{1, 0\}$ is a binary label indicating whether the response is correct (1) or incorrect (0). The reward model $r_\theta(x, y)$ is parameterized by θ and is trained to approximate the ideal reward function $r_*(x, y)$ by minimizing a loss function that encourages consistency with the labeled data. A common approach is to optimize a binary cross-entropy loss:

$$\mathcal{L}(\theta) = -\sum_{i=1}^{n} \left[z_i \log r_\theta(x_i, y_i) + (1 - z_i) \log(1 - r_\theta(x_i, y_i)) \right],$$

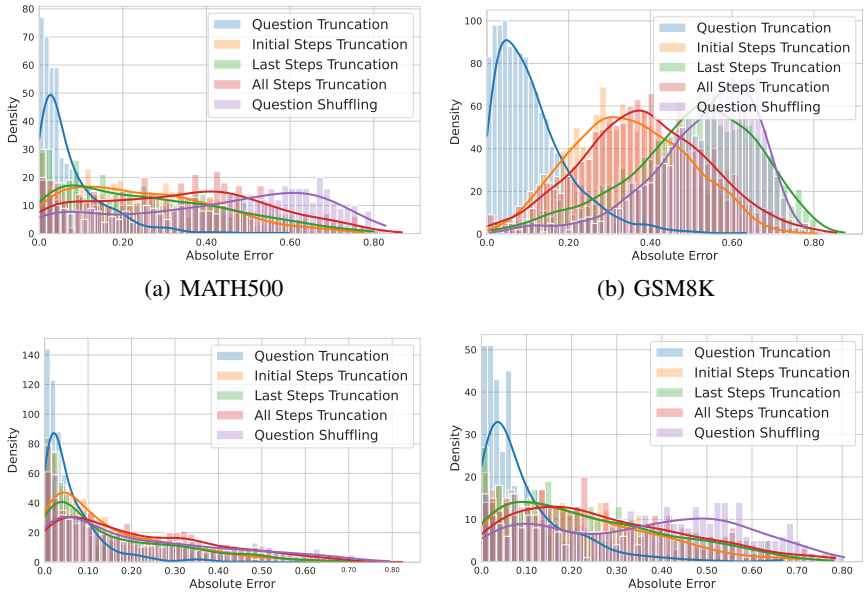
where $r_\theta(x_i, y_i)$ is interpreted as the probability that y_i is correct given x_i. This formulation ensures that the reward model learns to distinguish between high-quality and low-quality responses. Once trained, the reward model can be used to guide response generation in reinforcement learning or ranking-based optimization frameworks.

In addition, people are refining the reward model with the process-based supervision.

$$r_*(x, y_{1:k}) = \mathbf{P}(y_{1:k} \text{ is correct} \mid x).$$

By incorporating process-based supervision, reward models can capture fine-grained signals that improve alignment with human reasoning, ultimately leading to more interpretable and controllable LLM outputs. This paradigm enables reward models to provide more structured feedback, particularly for tasks requiring multi-step reasoning or sequential decision-making.

In this paper, we would like to investigate the reward behavior of incomplete inputs to identify what are really matters for reward models. For example, empty input $r_\theta(\texttt{None}, y)$, truncated output $r_\theta(x, y_{1:n/2})$ or $r_\theta(x, y_{n/2:n})$, shuffled input $r_\theta(x', y)$.



(a) MATH500 (b) GSM8K

(c) Olympiad Bench (d) GaoKao-2023-En

Figure 2: Density maps of absolute reward errors across different datasets when truncating various input components, using Qwen-2.5-Math-1.5B-Instruct as the base model and Skywork-O1-Open-PRM-Qwen-2.5-1.5B as the reward model..

4 Experimental Setup

We outline the experimental setup used in our analysis in Sections 5 and 6.

Models. We utilize two reward model families: Skywork-o1-OpenPRM [o1 Team, 2024] and RLHFlow [Xiong et al., 2024b]. Specifically, our experiments include Skywork-o1-Open-PRM-Qwen-2.5-1.5B, Skywork-o1-Open-PRM-Qwen-2.5-7B, Llama3.1-8B-ORM-Deepseek-Data, and Llama3.1-8B-PRM-Deepseek-Data. For base models, we employ both general-purpose and math-focused LLMs, specifically Llama-3 [Dubey et al., 2024] and Qwen-2.5-Math [Yang et al., 2024b].

Datasets. We conduct experiments on a diverse set of reasoning tasks, including GSM8K [Cobbe et al., 2021], MATH500 [Hendrycks et al., 2021], OlympiadBench [He et al., 2024], GaoKao-2023-En [Liao et al., 2024], and Minerva Math Lewkowycz et al. [2022].

Default Setting. All experiments were conducted on NVIDIA H100 GPUs, using vLLM[Kwon et al., 2023] as the backend. We set the generation parameters to `temperature = 0.8` and `top_p = 1.0` for Best-of-N sampling and trajectory collection. A reasoning step is defined as a generation ending with `\n\n`. For the process reward model, each trajectory is scored based on the reward of its final step. In the error analysis experiments (Section5), we sample 32 trajectories per question.

5 Questions Matters Little

5.1 Which Input Matters Most?

To assess the relative importance of different components in reward model inputs, we systematically truncate various parts of the input and analyze their impact on model predictions. Specifically, we evaluate how the absence of key information—such as the question or portions of the solution—affects the reward assignment. The following truncation strategies are considered:

Question Truncation: The question is entirely removed, leaving only the solution trajectory as input to the reward model. This tests whether the model primarily relies on the reasoning process rather than the problem statement itself.

Initial Steps Truncation: The first half of the solution steps is removed, preserving only the latter portion. This examines whether early-stage reasoning contributes significantly to the model's assessment or if later steps alone are sufficient.

Last Steps Truncation: The final half of the solution steps is removed, retaining only the initial portion. This helps determine whether the model emphasizes intermediate reasoning steps or prioritizes the final stages of problem-solving.

All Steps Truncation: All solution steps are removed, leaving only the final answer box. This scenario isolates the influence of the final answer on the reward model's judgment, shedding light on whether the model evaluates reasoning quality or focuses primarily on correctness.

We quantify the impact of input modifications by computing the absolute error between the original reward output, r, and the reward of the truncated input, r^*, given by $|r - r^*|$. The results on MATH500, GSM8K, OlympiadBench, and GaoKao-2023-En are visualized in Figure 2. We present histograms of the error distribution across all questions, with each question sampled over 32 trajectories.

We observe that the error distributions vary across datasets and truncation strategies, highlighting the differing sensitivities of reward models to different input components. Question truncation generally results in a lower absolute error, suggesting that the model can still assign reasonable reward scores based on the solution trajectory alone. In contrast, initial steps truncation and last steps truncation exhibit distinct effects, with the latter often leading to larger errors, implying that final reasoning steps play a crucial role in the model's decision-making process. The all steps truncation condition, which retains only the final answer box, consistently produces the highest errors, particularly in MATH500 and OlympiadBench, indicating that intermediate reasoning steps are essential for accurate reward assignment.

These findings suggest that the question is less important than the reasoning steps in determining reward scores. The relatively low absolute error from question truncation indicates that the model can still evaluate solution quality even without explicit access to the problem statement. Conversely, the higher errors observed in last steps truncation and all steps truncation highlight the critical role of intermediate and final reasoning steps in the model's decision-making process. This suggests that reward models prioritize logical coherence and solution completeness over simply understanding the original question, emphasizing the necessity of reasoning depth in reward evaluation.

5.2 Consistency Matters

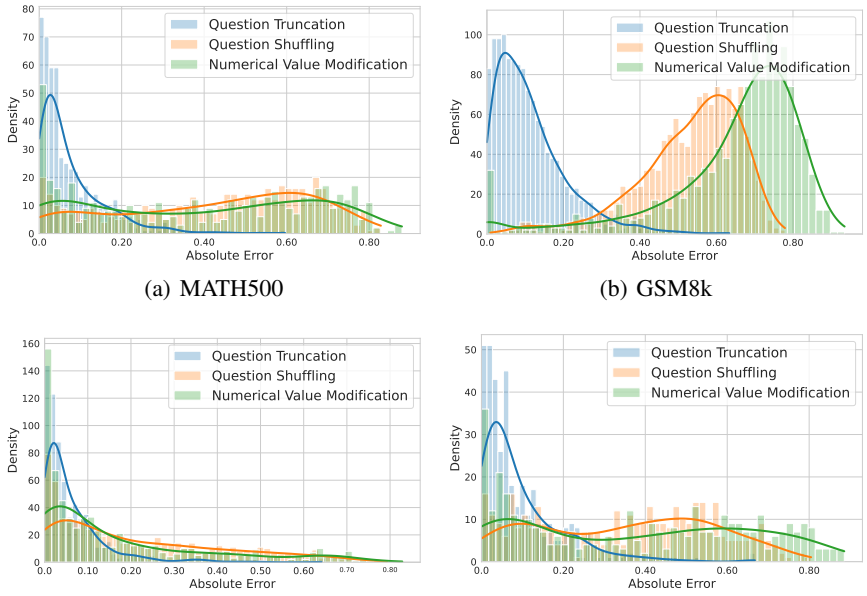
To gain deeper insights into the role of the question in reward modeling, we conduct a series of controlled experiments designed to assess the extent to which the reward model relies on the problem statement for evaluating solution quality. Specifically, we investigate how disrupting the question-solution relationship and altering key numerical values affect the model's reward assignment.

Question Shuffling: We shuffle the questions and their corresponding solution trajectories, disrupting the original question-solution pairings. This tests the reward model's reliance on the semantic coherence between the problem statement and the reasoning steps.

Numerical Value Modification: We replace numerical values in the question with random values, altering the problem while preserving its overall structure. This evaluates the model's sensitivity to specific numerical details in the question.

Figure 3 presents the absolute error distributions for different question modification strategies across four reasoning benchmarks. Across all datasets, question truncation consistently results in lower errors, suggesting that removing the question has a limited impact on the reward model. Question shuffling introduces moderate errors, particularly in GSM8K, indicating that the semantic consistency between the problem and solution plays a role in reward assignment. Numerical value modification, especially in GSM8K, leads to the largest errors, demonstrating that changes in numerical details significantly affect the reward model's predictions.

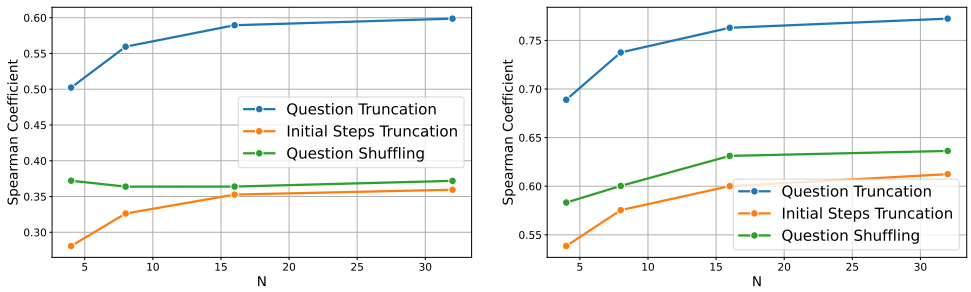
These results highlight the importance of consistency in the reward model's input. The fact that question truncation results in lower errors implies that the reward model primarily evaluates the



(a) MATH500 (b) GSM8k

(c) Olympiad Bench (d) Gaokao2023 En

Figure 3: Density maps of absolute reward errors across different datasets. We compare the effects of question truncation, question shuffling, and numerical value modification using Qwen-2.5-Math-1.5B-Instruct as the base model and Skywork-O1-Open-PRM-Qwen-2.5-1.5B as the reward model.



(a) MATH500 (b) Olympiad Bench

Figure 4: Spearman correlation of reward rankings under input truncation and Shuffling on MATH500 and Olympiad Bench. N is the number of sampled trajectories for each input.

reasoning process rather than the question itself. However, the increased errors in question shuffling suggest that disrupting semantic alignment between the problem and solution trajectory weakens the model's ability to assign consistent rewards. Moreover, the high error from numerical value modification indicates that numerical consistency is crucial, as the model relies on specific values to gauge correctness.

6 Impact on Ranking of Rewards

A key application of reward models is to rank candidate outputs and select the best among them. Beyond absolute error analysis, it is crucial to evaluate how input modifications influence the relative ranking of rewards assigned to different outputs. To this end, we investigate the impact of question and reasoning modifications on ranking consistency through two complementary analyses:

- Rank Correlation of Rewards: We assess how well the reward model preserves the relative ranking of outputs after input modifications by computing ranking correlation metrics.

Table 1: Best-of-N results under different reward input modifications.

Base Model	RM	Setting	MATH500	GSM8K	Olympiad Bench	Minerva Math	Avg.
7B	-	-	83.2	95.7	41.2	37.5	64.4
Math Model, Base: Qwen2.5-Math-Instruct, RM: Skywork- o1-Open-PRM 1.5B							
7B	1.5B	$N = 4$	85.4	96.9	43.3	39.3	66.2
7B	1.5B	$N = 8$	84.6	96.7	45.3	38.2	66.2
7B	1.5B	$N = 16$	85.4	96.9	45.0	38.6	66.5
Question Truncation							
7B	1.5B	$N = 4$	85.8	96.4	43.0	36.0	65.3
7B	1.5B	$N = 8$	86.4	97.0	42.7	37.9	66.0
7B	1.5B	$N = 16$	86.0	96.1	45.3	37.1	66.1
Initial Steps Truncation							
7B	1.5B	$N = 4$	85.0	96.4	39.7	36.4	64.4
7B	1.5B	$N = 8$	86.4	96.3	43.0	36.4	65.5
7B	1.5B	$N = 16$	85.4	96.0	43.1	37.5	65.5
Last Steps Truncation							
7B	1.5B	$N = 4$	84.0	95.5	42.4	38.2	65.0
7B	1.5B	$N = 8$	87.0	96.1	38.8	34.9	64.2
7B	1.5B	$N = 16$	83.6	95.6	41.8	37.5	64.6
Question Shuffling							
7B	1.5B	$N = 4$	84.8	95.9	41.9	37.5	65.0
7B	1.5B	$N = 8$	85.0	96.1	42.7	37.1	65.2
7B	1.5B	$N = 16$	85.0	96.7	44.0	36.0	65.4
Math Model, Base: Qwen2.5-Math-Instruct, RM: Skywork- o1-Open-PRM 7B							
7B	7B	$N = 4$	86.0	97.0	41.8	38.2	65.8
7B	7B	$N = 8$	87.0	97.1	45.3	37.5	66.7
7B	7B	$N = 16$	86.0	97.1	47.0	39.3	67.4
Question Truncation							
7B	7B	$N = 4$	86.2	96.4	44.0	35.3	65.5
7B	7B	$N = 8$	86.0	96.2	43.1	37.9	65.8
7B	7B	$N = 16$	84.0	96.3	46.5	41.9	67.2
Initial Steps Truncation							
7B	7B	$N = 4$	82.8	96.3	41.5	38.2	64.7
7B	7B	$N = 8$	85.4	96.7	42.7	40.1	66.2
7B	7B	$N = 16$	84.6	96.3	43.9	40.8	66.4
Last Steps Truncation							
7B	7B	$N = 4$	84.4	96.4	39.9	39.7	65.1
7B	7B	$N = 8$	83.4	96.1	40.7	37.1	64.3
7B	7B	$N = 16$	85.6	96.4	43.0	37.1	64.5
Question Shuffling							
7B	7B	$N = 4$	83.4	96.6	39.0	34.9	63.5
7B	7B	$N = 8$	81.8	95.8	43.0	36.8	64.4
7B	7B	$N = 16$	83.2	95.3	41.6	37.9	64.7

- Best-of-N Selection: We evaluate the effect of input modifications on Best-of-N selection performance, where the reward model is used to choose the best candidate from a set of generated outputs.

6.1 Rank Correlation of Rewards

Using Qwen2.5-Math-7B-Instruct as the base model and Skywork-o1-Open-PRM-Qwen-2.5-1.5B as the reward model, we generate 4, 8, 16, and 32 candidate outputs per question on MATH500 and OlympiadBench. To assess the stability of reward rankings under input modifications, we compute Spearman's rank correlation coefficient (ρ) as our ranking consistency metric. Spearman's correlation quantifies the monotonic relationship between two variables, providing insight into how well the reward model preserves the relative ranking of candidate outputs despite perturbations in input structure. A high Spearman correlation indicates that the ranking of outputs remains stable regardless of modifications. Conversely, a low correlation signifies that input modifications significantly alter the ranking behavior, revealing potential inconsistencies in the model's reward assignment process.

Figure 4 presents the Spearman rank correlation coefficients for different truncation strategies across varying values of N. Removing the question consistently results in the highest rank correlation across all values of N, suggesting that the reward model remains relatively stable in ranking outputs even when the problem statement is absent. Additionally, the correlation increases as N grows, indicating that the ranking consistency improves when selecting from a larger set of candidate outputs. In contrast, removing the first half of the solution steps leads to a significantly lower correlation, implying that complete reasoning steps is important in determining rankings. Furthermore, disrupting the

semantic coherence between the question and solution has a noticeable impact on ranking correlation, demonstrating that maintaining a logically consistent input structure is important for stable ranking performance.

These findings further confirms that reward models prioritize reasoning consistency over direct question comprehension. While the models demonstrate robustness to question removal, their sensitivity to reasoning disruptions suggests that they rely more on structural coherence than an actual understanding of problem-solving logic.

6.2 Best-of-N

Table 1 and Table 2 present the Best-of-N results under different input modifications for reward models. For the Qwen2.5-Math-Instruct-7B models, removing the question has a relatively minor impact on performance, with scores remaining stable across different values of N. For instance, when N=16, the model achieves an average score of 66.1, only 0.4 points lower than the vanilla Best-of-N setting. This finding aligns with previous observations that reward models primarily rely on reasoning steps rather than the problem statement itself. In contrast, truncating initial or final steps leads to a more noticeable performance drop, highlighting the importance of complete reasoning trajectories. Additionally, question shuffling results in the largest performance degradation across both reward models, reinforcing the necessity of consistency between the question and solution for effective ranking.

For the Llama-3.2 base model, we evaluate both ORM and PRM reward models (Appendix A.1). Similar to the Qwen models, removing the question leads to only minor performance degradation. However, the most significant drop is observed in the question shuffling condition, particularly in Olympiad Bench and Minerva Math, where the disrupted semantic alignment prevents the reward model from identifying high-quality reasoning trajectories among multiple samples. With shuffled questions, the average performance drops to 24.0, even worse than the baseline of 24.8. These results further emphasize that reward models prioritize structural consistency and reasoning flow over explicit problem comprehension.

Best-of-N results reflect the performance of the trajectory assigned the highest reward score, offering insights into how different input modifications impact the reward model's ability to identify the best solution. The results in the table reinforce the key finding that reward models prioritize structural consistency over causal understanding. The relatively stable performance in question truncation suggests that the reward model does not rely on the problem statement itself to evaluate responses. Instead, it primarily assesses the internal coherence of the solution trajectory. However, the consistency between the question and the solution remains crucial, as disruptions can significantly degrade performance, highlighting the importance of semantic alignment in reward-based ranking.

7 Discussion and Conclusion

In this paper, we investigate the impact of input modifications on reward models and uncover key insights into their evaluation behavior. Our findings reveal that truncating the question has minimal impact on both absolute reward values and ranking consistency, suggesting that reward models primarily evaluate the solution trajectory rather than the problem statement itself. However, shuffling the question or modifying numerical values significantly alters the reward model's output, indicating that semantic coherence and numerical consistency play a crucial role in assessment. Additionally, incomplete reasoning steps lead to substantial changes in rankings, highlighting the model's strong reliance on a structured and complete reasoning trajectory rather than an explicit understanding of problem-solving steps.

The Consistency Bias in Reward Models. Our findings suggest that reward models are not truly evaluating the causal relationship between the question and its solution but rather the internal consistency of the reasoning process. Even when the question is removed, if the solution remains well-structured, the model continues to assign high scores. Conversely, when reasoning steps are truncated, the model struggles to maintain ranking consistency, indicating that it relies on pattern recognition rather than an actual causal understanding of the problem-solution relationship. This raises concerns about the model's ability to generalize beyond familiar solution structures and adapt

to novel problem distributions. Future research should explore techniques to mitigate this consistency bias and encourage a more causally grounded evaluation framework.

Towards Causality-Aware Reward Models. To move beyond consistency-driven ranking, future reward models should incorporate causal reasoning techniques to better assess the logical validity of solutions. Potential directions include:

- **Causality-Augmented Training**: Incorporating counterfactual reasoning tasks to train reward models to recognize causal dependencies rather than relying solely on surface-level patterns.
- **Chain-of-Thought Awareness**: Rewarding models not only for correct final answers but also for their adherence to logically structured reasoning chains, ensuring that each step contributes meaningfully to the solution.
- **Human-in-the-Loop Refinement**: Leveraging human preference data to penalize superficial pattern matching and encourage robust causal reasoning, improving the model's ability to distinguish valid reasoning from plausible but incorrect trajectories.

Reconsidering Reward Model Objectives. Current reward models might be optimizing for ranking stability rather than true problem-solving ability. This raises the need to rethink selection strategies, such as:

- Uncertainty-Aware Reward Models: Incorporating confidence-aware mechanisms to quantify the model's uncertainty in evaluating complex reasoning tasks.
- Deeper Reasoning Signals: Designing reward functions that explicitly capture reasoning depth and logical validity, rather than solely relying on surface-level agreement with high-ranked answers.

Ethics Statement

This work presents an empirical analysis of reward model behavior in large language models (LLMs), with a focus on understanding their evaluation mechanisms rather than proposing new models for deployment. All models and datasets used in this study are publicly available and widely used in the research community. We emphasize that our analyses are intended to identify limitations and inform better practices in reward model design.

Our findings highlight potential shortcomings in current reward modeling techniques, such as over-reliance on structural consistency and susceptibility to input manipulation. While these insights can inform more robust and causality-aware evaluation frameworks, they also underscore risks if reward models are deployed without careful validation in safety-critical or high-stakes applications.

We encourage researchers and practitioners to interpret our results responsibly, particularly when using reward models to guide generation in reinforcement learning or automated decision-making. Future work should aim to incorporate human-in-the-loop oversight and causal reasoning capabilities to ensure that reward models reflect genuine understanding and alignment with human values.

No human subjects or personally identifiable information were used in this study, and no new data were collected.

References

AI Anthropic. Claude 3.5 sonnet model card addendum. *Claude-3.5 Model Card*, 3, 2024.

Mohammad Gheshlaghi Azar, Zhaohan Daniel Guo, Bilal Piot, Remi Munos, Mark Rowland, Michal Valko, and Daniele Calandriello. A general theoretical paradigm to understand learning from human preferences. In *International Conference on Artificial Intelligence and Statistics*, pages 4447–4455. PMLR, 2024.

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

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

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. *arXiv preprint arXiv:2407.21787*, 2024.

Stephen Casper, Xander Davies, Claudia Shi, Thomas Krendl Gilbert, Jérémy Scheurer, Javier Rando, Rachel Freedman, Tomasz Korbak, David Lindner, Pedro Freire, et al. Open problems and fundamental limitations of reinforcement learning from human feedback. *arXiv preprint arXiv:2307.15217*, 2023.

Guoxin Chen, Minpeng Liao, Chengxi Li, and Kai Fan. Alphamath almost zero: process supervision without process. *arXiv preprint arXiv:2405.03553*, 2024.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde De Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, et al. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*, 2021.

Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences. *Advances in neural information processing systems*, 30, 2017.

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

Carson Denison, Monte MacDiarmid, Fazl Barez, David Duvenaud, Shauna Kravec, Samuel Marks, Nicholas Schiefer, Ryan Soklaski, Alex Tamkin, Jared Kaplan, et al. Sycophancy to subterfuge: Investigating reward-tampering in large language models. *arXiv preprint arXiv:2406.10162*, 2024.

Hanze Dong, Wei Xiong, Deepanshu Goyal, Yihan Zhang, Winnie Chow, Rui Pan, Shizhe Diao, Jipeng Zhang, Kashun Shum, and Tong Zhang. Raft: Reward ranked finetuning for generative foundation model alignment. *arXiv preprint arXiv:2304.06767*, 2023.

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

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Yann Dubois, Xuechen Li, Rohan Taori, Tianyi Zhang, Ishaan Gulrajani, Jimmy Ba, Carlos Guestrin, Percy Liang, and Tatsunori B Hashimoto. Alpacafarm: A simulation framework for methods that learn from human feedback. *arXiv preprint arXiv:2305.14387*, 2023.

Kawin Ethayarajh, Winnie Xu, Niklas Muennighoff, Dan Jurafsky, and Douwe Kiela. Kto: Model alignment as prospect theoretic optimization. *arXiv preprint arXiv:2402.01306*, 2024.

Shangmin Guo, Biao Zhang, Tianlin Liu, Tianqi Liu, Misha Khalman, Felipe Llinares, Alexandre Rame, Thomas Mesnard, Yao Zhao, Bilal Piot, et al. Direct language model alignment from online ai feedback. *arXiv preprint arXiv:2402.04792*, 2024.

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

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. *arXiv preprint arXiv:2103.03874*, 2021.

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

Borja Ibarz, Jan Leike, Tobias Pohlen, Geoffrey Irving, Shane Legg, and Dario Amodei. Reward learning from human preferences and demonstrations in atari. *Advances in neural information processing systems*, 31, 2018.

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

Dongfu Jiang, Xiang Ren, and Bill Yuchen Lin. Llm-blender: Ensembling large language models with pairwise comparison and generative fusion. In *Proceedings of the 61th Annual Meeting of the Association for Computational Linguistics (ACL 2023)*, 2023b.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*, 2020.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the 29th Symposium on Operating Systems Principles*, pages 611–626, 2023.

Aitor Lewkowycz, Anders Andreassen, David Dohan, Ethan Dyer, Henryk Michalewski, Vinay V. Ramasesh, Ambrose Slone, Cem Anil, Imanol Schlag, Theo Gutman-Solo, Yuhuai Wu, Behnam Neyshabur, Guy Gur-Ari, and Vedant Misra. Solving quantitative reasoning problems with language models. In Sanmi Koyejo, S. Mohamed, A. Agarwal, Danielle Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, 2022.

Baohao Liao, Yuhui Xu, Hanze Dong, Junnan Li, Christof Monz, Silvio Savarese, Doyen Sahoo, and Caiming Xiong. Reward-guided speculative decoding for efficient llm reasoning. *arXiv preprint arXiv:2501.19324*, 2025.

Minpeng Liao, Wei Luo, Chengxi Li, Jing Wu, and Kai Fan. Mario: Math reasoning with code interpreter output–a reproducible pipeline. *arXiv preprint arXiv:2401.08190*, 2024.

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

Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, et al. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437*, 2024a.

Jie Liu, Zhanhui Zhou, Jiaheng Liu, Xingyuan Bu, Chao Yang, Han-Sen Zhong, and Wanli Ouyang. Iterative length-regularized direct preference optimization: A case study on improving 7b language models to gpt-4 level. *arXiv preprint arXiv:2406.11817*, 2024b.

Liangchen Luo, Yinxiao Liu, Rosanne Liu, Samrat Phatale, Harsh Lara, Yunxuan Li, Lei Shu, Yun Zhu, Lei Meng, Jiao Sun, et al. Improve mathematical reasoning in language models by automated process supervision. *arXiv preprint arXiv:2406.06592*, 2024.

Rémi Munos, Michal Valko, Daniele Calandriello, Mohammad Gheshlaghi Azar, Mark Rowland, Zhaohan Daniel Guo, Yunhao Tang, Matthieu Geist, Thomas Mesnard, Andrea Michi, et al. Nash learning from human feedback. *arXiv preprint arXiv:2312.00886*, 2023.

Skywork o1 Team. Skywork-o1 open series. https://huggingface.co/Skywork, November 2024. URL https://huggingface.co/Skywork.

OpenAI. OpenAI: Introducing ChatGPT, 2022. URL https://openai.com/blog/chatgpt.

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

Junsoo Park, Seungyeon Jwa, Meiying Ren, Daeyoung Kim, and Sanghyuk Choi. Offsetbias: Leveraging debiased data for tuning evaluators. *arXiv preprint arXiv:2407.06551*, 2024.

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. *Advances in Neural Information Processing Systems*, 36, 2024.

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

Prasann Singhal, Tanya Goyal, Jiacheng Xu, and Greg Durrett. A long way to go: Investigating length correlations in rlhf, 2024. URL https://arxiv.org/abs/2310.03716.

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

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 *NeurIPS*, 2020.

Yunhao Tang, Zhaohan Daniel Guo, Zeyu Zheng, Daniele Calandriello, Rémi Munos, Mark Rowland, Pierre Harvey Richemond, Michal Valko, Bernardo Ávila Pires, and Bilal Piot. Generalized preference optimization: A unified approach to offline alignment. *arXiv preprint arXiv:2402.05749*, 2024.

Gemini Team, Petko Georgiev, Ving Ian Lei, Ryan Burnell, Libin Bai, Anmol Gulati, Garrett Tanzer, Damien Vincent, Zhufeng Pan, Shibo Wang, et al. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *arXiv preprint arXiv:2403.05530*, 2024.

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 *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9426–9439, 2024.

Wei Xiong, Hanze Dong, Chenlu Ye, Ziqi Wang, Han Zhong, Heng Ji, Nan Jiang, and Tong Zhang. Iterative preference learning from human feedback: Bridging theory and practice for rlhf under kl-constraint. In *Forty-first International Conference on Machine Learning*, 2024a.

Wei Xiong, Hanning Zhang, Nan Jiang, and Tong Zhang. An implementation of generative prm, 2024b.

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

An Yang, Beichen Zhang, Binyuan Hui, Bofei Gao, Bowen Yu, Chengpeng Li, Dayiheng Liu, Jianhong Tu, Jingren Zhou, Junyang Lin, et al. Qwen2. 5-math technical report: Toward mathematical expert model via self-improvement. *arXiv preprint arXiv:2409.12122*, 2024b.

Chenlu Ye, Wei Xiong, Yuheng Zhang, Hanze Dong, Nan Jiang, and Tong Zhang. Online iterative reinforcement learning from human feedback with general preference model. 2024. URL https://openreview.net/forum?id=TwdX1W3M6S.

Fei Yu, Anningzhe Gao, and Benyou Wang. Outcome-supervised verifiers for planning in mathematical reasoning. *arXiv preprint arXiv:2311.09724*, 2023.

Tianyi Zhang, Faisal Ladhak, Esin Durmus, Percy Liang, Kathleen McKeown, and Tatsunori B Hashimoto. Benchmarking large language models for news summarization. *Transactions of the Association for Computational Linguistics*, 12:39–57, 2024.

Xingxing Zhang, Furu Wei, and Ming Zhou. Hibert: Document level pre-training of hierarchical bidirectional transformers for document summarization. *arXiv preprint arXiv:1905.06566*, 2019.

Yao Zhao, Rishabh Joshi, Tianqi Liu, Misha Khalman, Mohammad Saleh, and Peter J Liu. Slic-hf: Sequence likelihood calibration with human feedback. *arXiv preprint arXiv:2305.10425*, 2023.

Aojun Zhou, Ke Wang, Zimu Lu, Weikang Shi, Sichun Luo, Zipeng Qin, Shaoqing Lu, Anya Jia, Linqi Song, Mingjie Zhan, et al. Solving challenging math word problems using gpt-4 code interpreter with code-based self-verification. *arXiv preprint arXiv:2308.07921*, 2023.

A Appendix

A.1 Best-of-N Results on LLaMA

For the Llama-3.2 1B base model, we evaluate both ORM and PRM reward models under different input modifications.

Table 2: Best-of-N results under different reward input modifications.

Draft	RM	Setting	MATH500	GSM8K	Olympiad Bench	Minerva Math	Avg.
1B	-	-	31.4	54.0	7.7	6.2	24.8
General Model, Base: Llama-3.2, RM: Llama3.1-8B-ORM-Deepseek-Data							
1B	7B	$N = 4$	38.2	68.7	10.1	10.3	31.8
1B	7B	$N = 8$	43.2	74.7	12.4	10.3	35.2
Question Truncation							
1B	7B	$N = 4$	37.0	62.8	9.8	9.6	29.8
1B	7B	$N = 8$	40.8	68.6	9.6	9.6	32.2
Initial Steps Truncation							
1B	7B	$N = 4$	35.2	62.0	9.2	11.4	29.5
1B	7B	$N = 8$	41.0	66.6	11.0	10.3	32.2
Last Steps Truncation							
1B	7B	$N = 4$	36.2	64.2	9.9	11.8	30.5
1B	7B	$N = 8$	37.6	68.8	10.4	13.6	32.6
Question Shuffling							
1B	7B	$N = 4$	29.0	54.7	8.6	9.2	25.4
1B	7B	$N = 8$	28.4	53.8	7.0	6.6	24.0
General Model, Base: Llama-3.2, RM: Llama3.1-8B-PRM-Deepseek-Data							
1B	7B	$N = 4$	35.4	64.1	9.8	8.5	29.5
1B	7B	$N = 8$	41.2	69.1	10.4	11.4	33.0
Question Truncation							
1B	7B	$N = 4$	36.4	62.2	9.8	8.1	29.1
1B	7B	$N = 8$	39.0	64.6	10.8	9.9	31.1
Initial Steps Truncation							
1B	7B	$N = 4$	33.2	63.3	8.3	9.6	28.6
1B	7B	$N = 8$	39.4	62.7	12.0	9.2	30.8
Last Steps Truncation							
1B	7B	$N = 4$	33.6	62.5	8.1	10.3	28.6
1B	7B	$N = 8$	36.4	64.9	7.7	9.9	29.7
Question Shuffling							
1B	7B	$N = 4$	27.4	53.1	6.7	5.1	23.1
1B	7B	$N = 8$	29.8	50.0	6.8	6.2	23.2