# Fixing the Broken Compass: Diagnosing and Improving Inference-Time Reward Modeling

**Anonymous Author(s)**
Affiliation
Address
email

## Abstract

Inference-time scaling techniques have shown promise in enhancing the reasoning capabilities of large language models (LLMs). While recent research has primarily focused on training-time optimization, our work highlights inference-time reward model (RM)-based reasoning as a critical yet overlooked avenue. In this paper, we conduct a systematic analysis of RM behavior across downstream reasoning tasks, revealing three key limitations: (1) RM can impair performance on simple questions, (2) its discriminative ability declines with increased sampling, and (3) high search diversity undermines RM performance. To address these issues, we propose **CRISP** (Clustered Reward Integration with Stepwise Prefixing), a novel inference-time algorithm that clusters generated reasoning paths by final answers, aggregates reward signals at the cluster level, and adaptively updates prefix prompts to guide generation. Experimental results demonstrate that CRISP significantly enhances LLM reasoning performance, achieving up to **5%** accuracy improvement over other RM-based inference methods and an average of **10%** gain over advanced reasoning models.

## 1 Introduction

The remarkable achievements of OpenAI's o1 have sparked a wave of research into inference-time scaling techniques in reasoning tasks [21, 6, 42]. Some works aim to enhance models during the training phase, employing reinforcement learning (RL) [38, 23] or supervised fine-tuning (SFT) [41, 19] on high-quality data to equip models with the ability to generate long chains of thought (CoT). Other approaches focus on inference-time optimization, using reward model (RM)-based search strategies such as Monte Carlo Tree Search (MCTS) to guide the model toward more efficient solution paths [35, 29, 43].

Driven by the great success of the DeepSeek-R1 series [6], recent efforts have predominantly focused on reproducing its performance from a training-centric perspective [19, 41, 38], while largely overlooking inference optimization methods. Although R1-style works achieve strong performance on tasks such as math reasoning, they have been shown to suffer from serious issues such as overthinking [4, 31] and limited task generalization [44, 47]. These issues, however, can be mitigated through RM-based inference techniques. For example, on the commonsense reasoning task CSQA [33], DeepSeek-R1-7B [6] achieves 64.8 accuracy with an average of 3,613 tokens. In contrast, our RM-based inference method, applied to its base model Qwen2.5-Math-7B [40], reaches a higher accuracy of **72.0** using only **1,100 tokens** on average. Therefore, optimizing inference-time reasoning remains a critical direction, particularly for smaller models.

How can we further improve the reasoning performance of LLMs at inference time? Revisiting R1-style work, one key insight is their identification of the reward hacking issue during RL training, which they address using rule-based reward functions, ultimately improving performances [16, 6, 7].

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

This raises a natural question: **Can we similarly analyze the issues of the reward model at inference time and mitigate them to enhance the LLM's reasoning ability?**

In this work, we investigate the factors affecting reward model performance at inference time and propose methods to mitigate its limitations. Specifically, we begin by mathematically modeling the RM-based inference process to identify its key influencing factors: the input questions, the number of sampled responses, and the search parameters. Then, we conduct targeted experiments to analyze the impact of each factor on RM performance: **(1) Input question:** We test the performance of BoN and MCTS across different question difficulty levels and demonstrate that RM-based inference significantly impairs performance on simple questions. **(2) Sampling number:** We analyze the RM's discriminative ability under different numbers $n$ and observe that its performance deteriorates as $n$ increases. The statistical analysis attributes this degradation to an inverse long-tail phenomenon, wherein the RM tends to assign higher scores to low-frequency, incorrect responses. **(3) Search parameters:** We focus on parameters controlling search diversity, such as sampling temperature and MCTS tree structure. Our results show that RM performs best under moderate diversity, while excessive diversity undermines reasoning accuracy.

To mitigate the former issues in RM-based inference, we design a novel algorithm called **CRISP** (**C**lustered **R**eward **I**ntegration with **S**tepwise **P**refixing). CRISP operates in an iterative fashion, where each round begins by sampling reasoning paths conditioned on a dynamic prefix set. These paths are then clustered by their final answers, allowing the algorithm to aggregate reward signals at the cluster level and thereby attenuate the RM's tendency to mis-rank rare but incorrect outputs. We further incorporate an early termination mechanism based on cluster cardinality, which enables efficient inference on simple questions and alleviates RM instability in such cases. Finally, high-scoring paths from dominant clusters inform the construction of stepwise prefixes for the next sampling round, enabling tighter control over search diversity by limiting the number of intermediate states explored. We conduct extensive experiments to compare our method with other baselines. The results not only indicate that our method is effective in improving RM-based reasoning abilities, with accuracy gains of up to **5%**, but also validate the soundness of our earlier findings. Moreover, compared to DeepSeek-R1 models of the same scale, our method reduces average token usage by up to **90%**, while achieving an average accuracy improvement of **10%** on non-mathematical tasks.

Our main contributions are as follows: (1) We draw three critical findings based on a systematic analysis of RM behavior during inference: RM degrades performance on simple questions, fails to effectively distinguish low-frequency incorrect samples, and performs suboptimally under excessive search diversity. (2) We propose CRISP, a novel inference-time algorithm that clusters generated reasoning paths by final answers, aggregates reward signals at the cluster level, and adaptively updates prefix prompts to guide generation, effectively mitigating the shortcomings of reward models at inference time. (3) Extensive experiments demonstrate that CRISP consistently outperforms both inference-time and training-time baselines, with accuracy improvements of up to **5%** compared to other RM-based inference methods, and an average of **10%** over R1 models in non-mathematical reasoning tasks.

## 2 Overall Performance of Reward Models in Inference-Time

We first evaluate the overall performance of the reward model in inference time as our preliminary experiments. Here we compare the accuracy of Best-of-N (BoN), which generates multiple responses and selects the best one based on the reward score.

**Experimental Setup** For the policy model, we select some representative open-source models: Gemma2-9B [24], Llama3.1-8B [25], Qwen2.5-3B and Qwen2.5-14B [39]. For the evaluation of reward models, we consider several advanced works, including two outcome reward models (ORMs)—ArmoRM [34] and Skywork-Llama-3.1-8B [14]—and two process reward models (PRMs)—Shepherd-Mistral-7B-PRM [35] and Skywork-o1-PRM-Qwen-2.5-7B [20]. These models demonstrate commendable performance on related benchmarks (see Appendix A for details). As for the evaluation data, following previous works [30, 3, 22], we select MATH-500 [10, 12], which consists of high-school competition-level math problems. In addition to BoN, we also set two baselines: **SC** and **Oracle**. For the former, we select the major voting answer from $n$ responses. For the latter, we directly recall the existing correct answer from the generated samples, which serves as the performance ceiling.

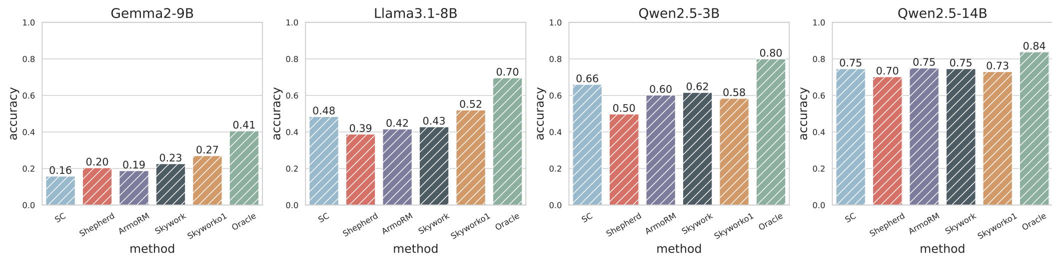



Figure 1: The performance of different policy models using various reward models for BoN inference on the MATH dataset ($n$ = 10).

**Main Results** Figure 1 shows the main results of the evaluation (more results, including more datasets and inference strategy in Appendix B). We can conclude that: **Advanced reward models have limited performance on the downstream math reasoning task.** For most LLMs, BoN only provides minor improvements over SC (<5%). Specifically, on Qwen2.5-3B, the BoN for all reward models exhibits lower accuracy than SC, indicating that the BoN inference method has limited reasoning performance. Besides, Oracle significantly outpaces other baselines, suggesting that the performance bottleneck lies in the RM's discriminative ability rather than the LLM's generative capability. Therefore, **identifying and mitigating the factors that impair the RM's performance during inference are crucial for enhancing LLM's reasoning ability**.

# 3 Probing RM-based Inference Issues

## 3.1 Mathematical Modeling

During the inference phase, the first step is to input the question $q$ and generate multiple responses $\mathcal{R}$:

$$\mathcal{R} = \mathcal{S}(\mathcal{M}(q), n; \Phi) \tag{1}$$

where $\mathcal{M}(q)$ denotes the output distribution of the policy model after inputting the question, $n$ denotes the number of samples and $\Phi$ denotes the parameters of the search strategy $\mathcal{S}$ (such as sampling temperature). After that, we use a scoring function $f$ to select the best response $\hat{r}$ from $\mathcal{R}$:

$$\hat{r} = \arg\max_{r \in \mathcal{R}} f(r) \tag{2}$$

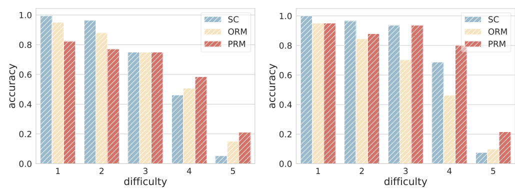
To analyze the performance of the reward model, we define $f$ as the score predicted by the RM. Our work focuses on identifying key factors that influence RM performance. To this end, we vary the components in Eq.1 to observe the accuracy of predicted $\hat{r}$ under different $\mathcal{R}$. Specifically, we study three main factors through probing experiments: the input question $q$, the sampling number $n$, and the search parameters $\Phi$.

## 3.2 Experimental Setup

For reward models, based on results in Figure 1, we select the best-performing Skywork and Skywork-o1 as the ORM and PRM for our subsequent experiments. Regarding policy models, we use Qwen2.5-3B and Llama3.1-8B throughout our experiments. To ensure that our findings are not specific to a particular strategy, we conduct all experiments using both BoN and MCTS. As for evaluation data, we employ the MATH-500 dataset in our main text, and provide additional results on GSM8K [5] and OlympiadBench [9] in the appendix.

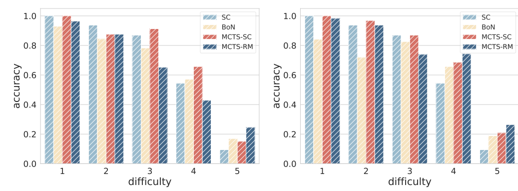
## 3.3 Input Question: Reward Model Underperforms on Easy Questions

**Question Difficulty Modeling** We first investigate how different questions affect the RM's performance. Following former works, we use question difficulty as a metric to classify different questions [12, 30]. We bin the policy model's pass@1 rate (estimated from 10 samples) on each question into five quantiles, each corresponding to increasing difficulty levels. For example, If the model answers correctly 0 or 1 time, the question is level 5 (hardest). If it answers correctly more than 8 times, the question is level 1 (easiest). Besides, we also study the difficulty approximation without the ground truth and report results in Appendix C.



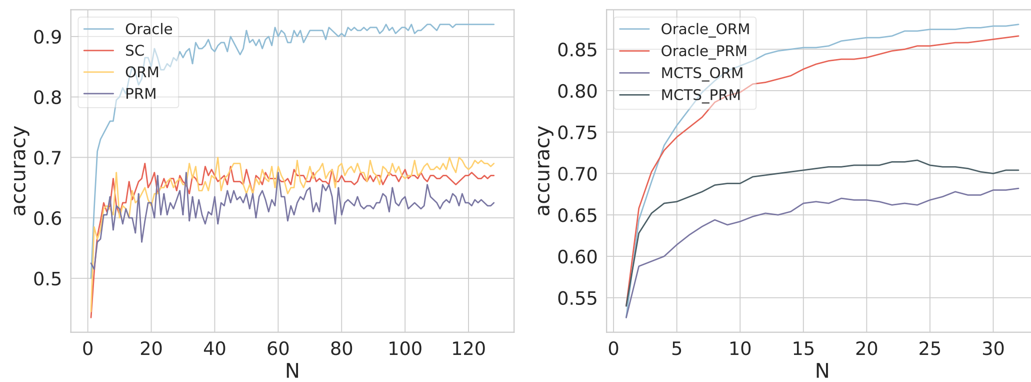

(a) Qwen2.5-3B (b) Llama3.1-8B

Figure 2: Performance of BoN inference across different question difficulty levels.




(a) ORM (b) PRM

Figure 3: Performance of MCTS inference across different question difficulty levels.




(a) BoN (b) MCTS

Figure 4: Two inference methods performance across difference sampling numbers.

**BoN Performance** After categorizing the data by difficulty, we analyze the BoN performance across different levels. We sample 32 examples from each question and illustrate the accuracy in Figure 2, from which we can conclude that: **Compared to SC, BoN performs worse on simple but better on difficult questions.** From the easiest level 1 to the hardest level 5, the accuracy of SC gradually declines, while BoN transitions from lagging behind SC to surpassing it. We also repeat the experiment on two more math reasoning benchmarks and present the results in Appendix D, further confirming our conclusion.

**MCTS Performance** In MCTS, we use two different scoring functions $f$ to select the final response for comparison: MCTS-SC and MCTS-RM (more functions in Appendix B). For the former, we employ a majority voting method for selection. For the latter, we choose the path with the highest reward score. We perform 32 rollouts over 200 questions, demonstrating the results in Figure 3. Although MCTS provides improvement over BoN, the accuracy of MCTS-RM still lags behind that of SC for low-difficulty problems (see levels 1 and 2 in Figure 3a). Besides, MCTS-SC achieves higher accuracy on easy questions but performs worse on harder questions compared to MCTS-RM. These indicate that: **(Cl.1) The introduction of the RM can hinder the LLM's reasoning performance on simple problems.** This pattern is not limited to specific inference strategies.

### 3.4 Sampling Number: RM struggles to distinguish low-frequency negatives

**Gap between Accuracy and Coverage** Recent works [3] demonstrate the LLM's coverage of correct answers increases as the sampling number grows, whereas the accuracy does not fully scale with $n$. Based on this, we further investigate whether introducing better RMs and inference strategies can reduce the gap between coverage and accuracy. The changes in accuracy and coverage are shown in Figure 4. The results demonstrate that: **Regardless of the reward model or inference strategy used, the model's accuracy does not improve as $n$ increases.** For both figures, the accuracy plateaus beyond a relatively small number of samples (approximately 30). In contrast, the Oracle setting consistently increases, leading to a persistently widening gap between accuracy and coverage.

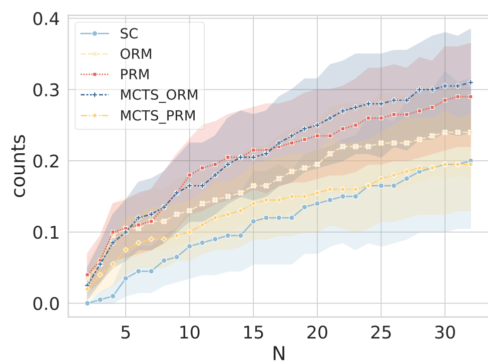



Figure 5: The number of times the model's selection changes from correct to incorrect.

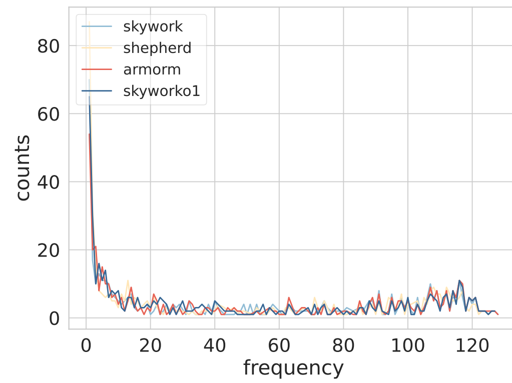


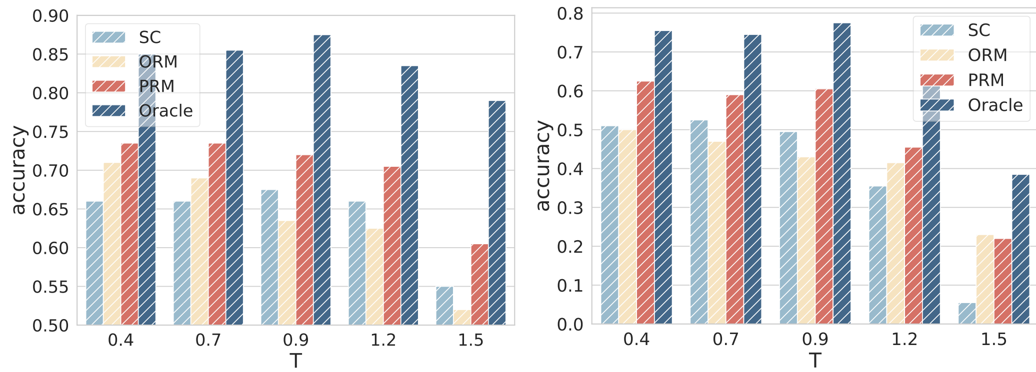

Figure 6: Frequency statistics of the highest-scored negative responses in BoN.

**Discriminative Performance** In the context of increasing coverage, the policy model's accuracy primarily depends on the reward model's discriminative capacity. Therefore, the plateau observed in Figure 4 is likely due to the reward model selecting incorrect answers as $n$ increases. To validate this claim, we begin with a case study, in which we randomly select a set of questions to examine the correctness of the RM's selections under different sampling numbers (see Appendix E for detailed results). We observe that in some cases, the reward model assigns the highest score to newly generated but incorrect responses, thereby causing originally correct answers to be replaced with incorrect ones as $n$ increases. Additionally, we record the number of instances in which the selected answer transitions from correct to incorrect and present the results in Figure 5. All methods exhibit a tendency for more incorrect transitions as $n$ increases. **This indicates that the model increasingly erroneous distinctions as the sampling size grows**. Moreover, compared to SC, RM-based inference methods show higher transition counts in Figure 5, which suggests that incorporating reward models introduces more incorrect selections.

**Inverse Long-tail Phenomenon** Why does the reward model perform worse as the sampling number grows? Reflecting on its training process [34, 14, 35], the training data primarily consists of paired responses (i.e., a correct one and an incorrect one). These pairs represent a constrained subset of the response space. We hypothesize that as $n$ grows, more low-frequency responses (those outside the training distribution) are sampled. The reward model struggles to generalize to these unfamiliar inputs, leading to incorrect responses occasionally receiving higher scores. To validate this hypothesis, we perform a statistical analysis of negative responses. For each question, we select the incorrect response with the highest RM score and compute the frequency of its answer across all samples. As shown in Figures 6 and 20, the RM displays an **inverse long-tail phenomenon** when scoring incorrect responses. For most questions, the top-scoring incorrect answers tend to have very low frequencies (frequency < 5 in Figure 6). Conversely, incorrect answers with high occurrence frequencies rarely achieved the highest scores. These findings support our hypothesis: **(Cl.2) RMs struggle to correctly score incorrect responses with low occurrence frequencies, making it difficult to distinguish incorrect responses from correct ones as $n$ grows.**

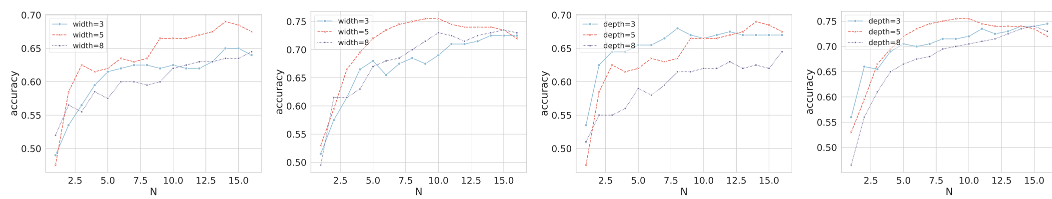
### 3.5 Search Parameters: RM performs worse on high-diversity distributions

**Search Diversity in BoN** The final influencing factor we investigate is the search parameters $\Phi$, which are primarily utilized to control the diversity of the policy model's search. For the BoN method, the temperature $T$ is the key parameter controlling the search diversity. We sweep $T$ and analyze its influence on the performance, as shown in Figure 7. For both policy models, BoN performance consistently degrades with increasing $T$, while SC and Oracle (i.e., coverage) remain stable except at high temperatures ($T$ > 0.9 in Figure 7). These results indicate that RM is more sensitive to sampling diversity than the policy model. **Higher diversity makes it challenging for the RM to distinguish between positive and negative responses.** To better understand this issue, we perform additional statistical analyses in Appendix F, which suggest that higher sampling temperatures cause the policy model to produce more low-frequency incorrect responses, thereby degrading discriminative accuracy.




(a) Qwen2.5-3B (b) Llama3.1-8B

Figure 7: Performance of BoN inference across different sampling temperatures.




(a) Width of ORM (b) Width of PRM (c) Depth of ORM (d) Depth of PRM

Figure 8: MCTS inference performance under different tree structures.

**Search Diversity in MCTS** In the MCTS algorithm, search diversity is primarily governed by the tree structure, determined by two key parameters: width and depth. The width refers to the number of child nodes at each node, whereas the depth denotes the length of the longest path from the root to a leaf node. A larger width indicates a broader search space during exploration, while a greater depth implies the model can traverse more intermediate states along a single trajectory. We evaluate MCTS performance under varying settings and present the results in Figure 8. The findings reveal: (1) For width, the best performance is observed at intermediate values (width = 5), too high widths lead to a decline in performance. (2) For depth, the best performance is achieved under settings with a lower value (e.g., depth = 3 or 5). These suggest that in MCTS, exploring too many intermediate states can harm performance. Notably, the optimal number of intermediate steps in search does not necessarily align with the number of steps a human would take to solve the same problem. We also analyze the impact of exploration weight on the diversity of MCTS, with consistent findings (see Appendix G). In summary, excessive diversity, such as width, depth, or temperature, can impair the performance of the reward model. Thus, we conclude: **(Cl.3) During inference, it is essential to constrain the diversity of the sampling distribution to maintain the optimal performance of the RM.**

# 4 Mitigating RM-based Inference Issues

## 4.1 Our Methodology

In the preceding sections, we uncover key patterns that affect the RM's performance and identify serval issues in RM-based reasoning. To mitigate these issues, we propose a novel RM-based inference algorithm called **Clustered Reward Integration with Stepwise Prefixing (CRISP)**. Figure 9 and Algorithm 1 demonstrate the main process of our method, which comprises five modules:

**Path Generation** Given a question $q$, during each iteration, we generate new reasoning paths based on the existing prefix set $\mathcal{P}$:

$$\mathcal{R} = \mathcal{R} \cup \mathcal{M}(q, n, \mathcal{P}) \tag{3}$$

In the generation process, the policy model generates $n$ complete sequences of remaining reasoning steps conditioned on $\mathcal{P}$ ($\mathcal{P} = \emptyset$ in the init iteration), rather than generating intermediate nodes step

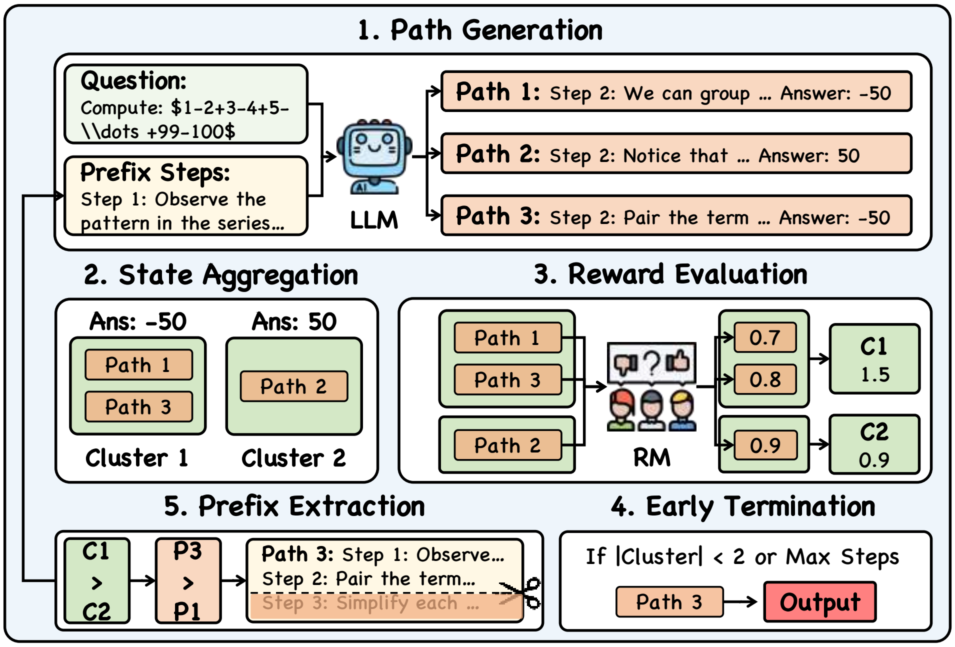



Figure 9: Main process of our CRISP method.

by step as in approaches like MCTS. This helps control the diversity of the search space and reduces the negative impact of excessive diversity on the reward model, as discussed in Cl.3.

**State Aggregation** To further reduce the complexity of the state space and mitigate the impact of low-frequency negative examples on the reward model's performance (as discussed in Cl.2), we define a final-answer-based state aggregation function $\psi$:

$$\psi : \mathcal{R} \rightarrow \mathcal{C} \tag{4}$$

where $\mathcal{C}$ is the set of final answer clusters (i.e., all responses leading to the same answer), and for any path $r_1, r_2 \in \mathcal{R}$, we have:

$$\psi(r_1) = \psi(r_2) \iff Answer(r_1) = Answer(r_2) \tag{5}$$

All paths that produce the same final answer are mapped to the same cluster $\mathcal{C}_j \in \mathcal{C}$. As an example, in Module 2 of Figure 9, paths 1 and 3, both with the answer of -50, are assigned to the same cluster.

**Reward Evaluation** After clustering the responses, we can convert the reward scores $f$ for each path into scores $\mathcal{F}$ for the corresponding clusters $\mathcal{C}_j$ (i.e., lines 17-20 in Algorithm 1):

$$\mathcal{F}(\mathcal{C}_j) = \sum_{x \in \mathcal{C}_j} f(x) \tag{6}$$

In the implementation, we normalize $f(x)$ before summing. By additionally considering the frequency of the answers associated with each path during scoring, we can prevent the reward model from assigning excessively high scores to low-frequency responses, thereby mitigating the issue identified in Cl.2. Although this may reduce the scores for some low-frequency correct responses, we will later demonstrate through ablation experiments that this design overall improves performance (see §4.4).

**Early Termination** This module controls when to exit the loop and return the final response. In addition to the standard exit condition of reaching the maximum number of iterations, we also control early termination by monitoring the number of clusters. If the number falls below a certain threshold (set to 2 in our work), it indicates that the question is relatively simple (as evidenced and discussed in Appendix C). In this case, the algorithm terminates, returning the answer corresponding to the most populated cluster, which is equivalent to SC. This not only reduces inference costs but also mitigates the issue of the reward model underperforming on simple questions (see Cl.1).

Table 1: Accuracy comparison in main experiments, the best results are highlighted in **bold**.

| Methods | | Qwen2.5-3B | | | Llama3.1-8B | | |
|---|---|---|---|---|---|---|---|
| | | *GSM8K* | *MATH* | *Olympiad* | *GSM8K* | *MATH* | *Olympiad* |
| CoT | | 0.78 | 0.46 | 0.24 | 0.85 | 0.38 | 0.11 |
| Self-Consistency | | 0.83 | 0.64 | 0.31 | 0.91 | 0.57 | 0.16 |
| Best-of-N | + ORM | 0.83 | 0.65 | 0.31 | 0.91 | 0.47 | 0.18 |
| | + PRM | 0.87 | 0.61 | 0.34 | 0.95 | 0.62 | 0.23 |
| BoN Weighted | + ORM | 0.83 | 0.67 | 0.31 | 0.89 | 0.53 | 0.20 |
| | + PRM | 0.86 | 0.60 | 0.36 | 0.94 | 0.62 | 0.24 |
| MCTS | + ORM | 0.92 | 0.67 | 0.34 | 0.90 | 0.43 | 0.13 |
| | + PRM | 0.95 | 0.71 | 0.31 | 0.95 | 0.57 | 0.19 |
| Beam Search | | 0.95 | 0.73 | 0.34 | 0.94 | 0.56 | 0.15 |
| **Ours** | + ORM | 0.91 | 0.70 | 0.36 | 0.89 | 0.49 | 0.18 |
| | + PRM | **0.96** | **0.76** | **0.39** | **0.95** | **0.67** | **0.26** |

**Prefix Extraction** In this module, we extract the top multiple prefixes as the new prefix set $\mathcal{P}$ for the next iteration, based on the scores of the paths and clusters. As illustrated in Module 5 of Figure 9, we first select the top-$k$ clusters with the highest scores (here, $k$=1, so we select Cluster 1). Then, from the selected cluster(s), we choose the path with the highest score (in this case, 0.8 > 0.7, so we select Path 3) to extract the prefix. Specifically, at the $i$-th generation, we extract the first $i$ steps of all paths as $\mathcal{P}$, and repeat the process until termination.

### 4.2 Main Experiments

**Experimental Setup** We compare the reasoning performance of our method with other advanced baselines, including: **CoT** [37], **Self-Consistency** [36], **Best-of-N**, **BoN Weighted** [30], **MCTS** [8] and **Beam Search** [30]. For datasets, in addition to MATH-500 [10, 12], we also validate our methods on GSM8K [5] and OlympiadBench [9]. For models, we continue to select Qwen2.5-3B and Llama3.1-8B as the policy model, while using Skywork-Llama-3.1-8B (ORM) and Skywork-o1-PRM-Qwen-2.5-7B (PRM) as the reward model. We present more details in Appendix H.

**Main Results** We demonstrate the result in Table 1, from which we can get the following conclusions: **(1) Our proposed CRISP method significantly improves RM's performance in reasoning tasks.** Across all benchmarks and both model backbones, CRISP consistently outperforms existing RM-based inference approaches. Notably, on the Llama3.1-8B model, CRISP achieves a performance gain of up to **5.0%** on the MATH dataset over the best-competing method. **(2) The findings from the preceding analysis are reasonable.** CRISP is specifically crafted to overcome the key issues of reward modeling revealed in §3. Its consistent and significant performance improvements provide strong empirical evidence that CRISP effectively mitigates these limitations, which are critical bottlenecks affecting the model's reasoning performance.

### 4.3 Training-Time vs. Inference-Time Optimization

To demonstrate the continued necessity of our inference-time optimization approach amid the rising dominance of RL and SFT techniques represented by the DeepSeek-R1 series, we compare our method against the R1 model across different reasoning tasks, including math reasoning (MATH-500), commonsense reasoning (CSQA [33]), social reasoning (SIQA [27]) and logical reasoning (LogiQA [15]). Specifically, given the same base model, we evaluate the accuracy and token consumption among its chat version (using CoT), the R1 distilled version, and our proposed method. From the results in Table 2, we can observe that: **(1) Our method enables more efficient reasoning across all tasks.** It achieves comparable reasoning tokens to the CoT method, while reducing output length by over **90%** compared to the R1 model in the best case. **(2) Our method exhibits stronger generalization capabilities.** Although it underperforms the R1 model on math tasks, it consistently outperforms R1 on other reasoning benchmarks, with average gains of **10%** and **5%** accuracy across two backbones. This highlights the advantage of our inference-time optimization in generalizing across diverse scenarios.

Table 2: Comparison between R1 models and our method, the best accuracy are highlighted in **bold**.

| Base Models | Methods | *Math* | | *Commonsense* | | *Social* | | *Logical* | |
|---|---|---|---|---|---|---|---|---|---|
| | | Acc | Length | Acc | Length | Acc | Length | Acc | Length |
| Qwen2.5-Math-1.5B | Chat | 0.52 | 1470 | 0.40 | 1400 | 0.46 | 1204 | 0.40 | 2790 |
| | R1-Distill | **0.79** | 13421 | 0.47 | 6066 | 0.52 | 6407 | 0.35 | 12352 |
| | Ours | 0.59 | 943 | **0.58** | 1004 | **0.61** | 1144 | **0.44** | 1143 |
| Qwen2.5-Math-7B | Chat | 0.74 | 1855 | 0.58 | 1479 | 0.58 | 1388 | 0.49 | 2133 |
| | R1-Distill | **0.88** | 9626 | 0.65 | 3612 | 0.66 | 2920 | 0.50 | 6492 |
| | Ours | 0.79 | 987 | **0.72** | 1100 | **0.66** | 1059 | **0.59** | 2058 |

### 4.4 Discussion and Future Work

**Ablation Study** We perform ablation experiments to validate the contribution of each module in the CRISP framework, with results summarized in Figure 23 of Appendix I. The results show that removing any single module leads to a decline in performance. As our design is informed by the analysis presented in §3 (i.e., Cl.1-Cl.3), the results provide further empirical support for our findings.

**Cost Analysis** As an inference-time method, in addition to accuracy, reasoning cost is also an important factor to consider. We therefore measure computational cost (e.g., number of generated tokens and inference time) in our evaluations and report the results in Figure 24 of Appendix J. It demonstrates that our CRISP method incurs lower costs compared to other advanced methods.

**Limitations & Future Work** While our work provides a thorough investigation of RM behavior during inference, it does not address potential issues that may arise during the training of models. In future work, we aim to extend our study to the training phase of reward models. Understanding how training dynamics (such as reward signal design and data sampling strategies) impact downstream reasoning performance could offer deeper insights and help improve the overall reliability of LLM.

## 5 Related Work

**Inference-time Optimization Technique in LLM's Reasoning** Recent studies have demonstrated that large language models (LLMs) can be effectively enhanced through search-based optimization at inference time [21, 42, 45]. These works primarily follow two approaches: optimizing the strategy for LLMs to search for answers [8, 30, 2, 22] or improving the reward model's ability to evaluate response quality [35, 43, 29]. However, most studies explore these two approaches separately, with limited research analyzing the impact of search factors on RM performance. Our work addresses this gap and proposes a new search strategy to mitigate RM's deficiencies.

**Reward Model in LLM's Reasoning** The reward model plays a crucial role in complex reasoning tasks of LLMs [42, 29, 35]. Existing works mainly investigate the RM from two perspectives: evaluation and optimization. For the former, researchers design various datasets to evaluate the RM's ability to distinguish between positive and negative responses [11, 17, 46]. For the latter, researchers focus on the training phase, improving the RM's ability by synthesizing high-quality data [35, 14] or optimizing the training algorithm [43, 1, 18]. There is a lack of in-depth analysis of the potential issues RM faces during inference, as well as methods to optimize RM's performance in the inference stage. Our work addresses the gaps left by these related studies.

## 6 Conclusion

In this work, we focus on analyzing key factors that influence the reward model's performance in reasoning tasks. We find that low question difficulty, large sampling number, and high search diversity can lead to issues in RM-based inference, with in-depth explanations provided. To address these issues, we propose CRISP, a cluster-based, prefix-guided inference algorithm that enhances the robustness and efficiency of the reward model. Experimental results demonstrate that our method is effective in enhancing LLM reasoning capabilities.

## References


[1] Z. Ankner, M. Paul, B. Cui, J. D. Chang, and P. Ammanabrolu. Critique-out-loud reward models. *CoRR*, abs/2408.11791, 2024.

[2] Z. Bi, K. Han, C. Liu, Y. Tang, and Y. Wang. Forest-of-thought: Scaling test-time compute for enhancing LLM reasoning. *CoRR*, abs/2412.09078, 2024.

[3] B. C. A. Brown, J. Juravsky, R. S. Ehrlich, R. Clark, Q. V. Le, C. Ré, and A. Mirhoseini. Large language monkeys: Scaling inference compute with repeated sampling. *CoRR*, abs/2407.21787, 2024.

[4] X. Chen, J. Xu, T. Liang, Z. He, J. Pang, D. Yu, L. Song, Q. Liu, M. Zhou, Z. Zhang, R. Wang, Z. Tu, H. Mi, and D. Yu. Do NOT think that much for 2+3=? on the overthinking of o1-like llms. *CoRR*, abs/2412.21187, 2024.

[5] K. Cobbe, V. Kosaraju, M. Bavarian, M. Chen, H. Jun, L. Kaiser, M. Plappert, J. Tworek, J. Hilton, R. Nakano, C. Hesse, and J. Schulman. Training verifiers to solve math word problems. *CoRR*, abs/2110.14168, 2021.

[6] DeepSeek-AI, D. Guo, D. Yang, H. Zhang, J. Song, R. Zhang, R. Xu, Q. Zhu, S. Ma, P. Wang, X. Bi, X. Zhang, X. Yu, Y. Wu, Z. F. Wu, Z. Gou, Z. Shao, Z. Li, Z. Gao, A. Liu, B. Xue, B. Wang, B. Wu, B. Feng, C. Lu, C. Zhao, C. Deng, C. Zhang, C. Ruan, D. Dai, D. Chen, D. Ji, E. Li, F. Lin, F. Dai, F. Luo, G. Hao, G. Chen, G. Li, H. Zhang, H. Bao, H. Xu, H. Wang, H. Ding, H. Xin, H. Gao, H. Qu, H. Li, J. Guo, J. Li, J. Wang, J. Chen, J. Yuan, J. Qiu, J. Li, J. L. Cai, J. Ni, J. Liang, J. Chen, K. Dong, K. Hu, K. Gao, K. Guan, K. Huang, K. Yu, L. Wang, L. Zhang, L. Zhao, L. Wang, L. Zhang, L. Xu, L. Xia, M. Zhang, M. Zhang, M. Tang, M. Li, M. Wang, M. Li, N. Tian, P. Huang, P. Zhang, Q. Wang, Q. Chen, Q. Du, R. Ge, R. Zhang, R. Pan, R. Wang, R. J. Chen, R. L. Jin, R. Chen, S. Lu, S. Zhou, S. Chen, S. Ye, S. Wang, S. Yu, S. Zhou, S. Pan, S. S. Li, S. Zhou, S. Wu, S. Ye, T. Yun, T. Pei, T. Sun, T. Wang, W. Zeng, W. Zhao, W. Liu, W. Liang, W. Gao, W. Yu, W. Zhang, W. L. Xiao, W. An, X. Liu, X. Wang, X. Chen, X. Nie, X. Cheng, X. Liu, X. Xie, X. Liu, X. Yang, X. Li, X. Su, X. Lin, X. Q. Li, X. Jin, X. Shen, X. Chen, X. Sun, X. Wang, X. Song, X. Zhou, X. Wang, X. Shan, Y. K. Li, Y. Q. Wang, Y. X. Wei, Y. Zhang, Y. Xu, Y. Li, Y. Zhao, Y. Sun, Y. Wang, Y. Yu, Y. Zhang, Y. Shi, Y. Xiong, Y. He, Y. Piao, Y. Wang, Y. Tan, Y. Ma, Y. Liu, Y. Guo, Y. Ou, Y. Wang, Y. Gong, Y. Zou, Y. He, Y. Xiong, Y. Luo, Y. You, Y. Liu, Y. Zhou, Y. X. Zhu, Y. Xu, Y. Huang, Y. Li, Y. Zheng, Y. Zhu, Y. Ma, Y. Tang, Y. Zha, Y. Yan, Z. Z. Ren, Z. Ren, Z. Sha, Z. Fu, Z. Xu, Z. Xie, Z. Zhang, Z. Hao, Z. Ma, Z. Yan, Z. Wu, Z. Gu, Z. Zhu, Z. Liu, Z. Li, Z. Xie, Z. Song, Z. Pan, Z. Huang, Z. Xu, Z. Zhang, and Z. Zhang. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning, 2025.

[7] L. Gao, J. Schulman, and J. Hilton. Scaling laws for reward model overoptimization. In *International Conference on Machine Learning*, pages 10835–10866. PMLR, 2023.

[8] S. Hao, Y. Gu, H. Ma, J. J. Hong, Z. Wang, D. Z. Wang, and Z. Hu. Reasoning with language model is planning with world model. In H. Bouamor, J. Pino, and K. Bali, editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pages 8154–8173. Association for Computational Linguistics, 2023.

[9] C. He, R. Luo, Y. Bai, S. Hu, Z. L. Thai, J. Shen, J. Hu, X. Han, Y. Huang, Y. Zhang, J. Liu, L. Qi, Z. Liu, and M. Sun. Olympiadbench: A challenging benchmark for promoting AGI with olympiad-level bilingual multimodal scientific problems. In L. Ku, A. Martins, and V. Srikumar, editors, *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024*, pages 3828–3850. Association for Computational Linguistics, 2024.

[10] D. Hendrycks, C. Burns, S. Kadavath, A. Arora, S. Basart, E. Tang, D. Song, and J. Steinhardt. Measuring mathematical problem solving with the MATH dataset. In J. Vanschoren and S. Yeung, editors, *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks 1, NeurIPS Datasets and Benchmarks 2021, December 2021, virtual*, 2021.

[11] N. Lambert, V. Pyatkin, J. Morrison, L. Miranda, B. Y. Lin, K. R. Chandu, N. Dziri, S. Kumar, T. Zick, Y. Choi, N. A. Smith, and H. Hajishirzi. Rewardbench: Evaluating reward models for language modeling. *CoRR*, abs/2403.13787, 2024.

[12] H. Lightman, V. Kosaraju, Y. Burda, H. Edwards, B. Baker, T. Lee, J. Leike, J. Schulman, I. Sutskever, and K. Cobbe. Let's verify step by step. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024.

[13] W. Ling, D. Yogatama, C. Dyer, and P. Blunsom. Program induction by rationale generation: Learning to solve and explain algebraic word problems. In R. Barzilay and M. Kan, editors, *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*, pages 158–167. Association for Computational Linguistics, 2017.

[14] C. Y. Liu, L. Zeng, J. Liu, R. Yan, J. He, C. Wang, S. Yan, Y. Liu, and Y. Zhou. Skywork-reward: Bag of tricks for reward modeling in llms. *CoRR*, abs/2410.18451, 2024.

[15] J. Liu, L. Cui, H. Liu, D. Huang, Y. Wang, and Y. Zhang. Logiqa: A challenge dataset for machine reading comprehension with logical reasoning. In C. Bessiere, editor, *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI 2020*, pages 3622–3628. ijcai.org, 2020.

[16] T. Liu, W. Xiong, J. Ren, L. Chen, J. Wu, R. Joshi, Y. Gao, J. Shen, Z. Qin, T. Yu, et al. Rrm: Robust reward model training mitigates reward hacking. *arXiv preprint arXiv:2409.13156*, 2024.

[17] Y. Liu, Z. Yao, R. Min, Y. Cao, L. Hou, and J. Li. Rm-bench: Benchmarking reward models of language models with subtlety and style. *CoRR*, abs/2410.16184, 2024.

[18] X. Lou, D. Yan, W. Shen, Y. Yan, J. Xie, and J. Zhang. Uncertainty-aware reward model: Teaching reward models to know what is unknown. *CoRR*, abs/2410.00847, 2024.

[19] N. Muennighoff, Z. Yang, W. Shi, X. L. Li, L. Fei-Fei, H. Hajishirzi, L. Zettlemoyer, P. Liang, E. J. Candès, and T. Hashimoto. s1: Simple test-time scaling. *CoRR*, abs/2501.19393, 2025.

[20] S. o1 Team. Skywork-o1 open series. `https://huggingface.co/Skywork`, November 2024.

[21] OpenAI. Introducing openai o1 preview., 2024. Accessed: 2025-01-24.

[22] Z. Qi, M. Ma, J. Xu, L. L. Zhang, F. Yang, and M. Yang. Mutual reasoning makes smaller llms stronger problem-solvers. *CoRR*, abs/2408.06195, 2024.

[23] Y. Qu, M. Y. R. Yang, A. Setlur, L. Tunstall, E. E. Beeching, R. Salakhutdinov, and A. Kumar. Optimizing test-time compute via meta reinforcement fine-tuning. *CoRR*, abs/2503.07572, 2025.

[24] M. Rivière, S. Pathak, P. G. Sessa, C. Hardin, S. Bhupatiraju, L. Hussenot, T. Mesnard, B. Shahriari, A. Ramé, J. Ferret, P. Liu, P. Tafti, A. Friesen, M. Casbon, S. Ramos, R. Kumar, C. L. Lan, S. Jerome, A. Tsitsulin, N. Vieillard, P. Stanczyk, S. Girgin, N. Momchev, M. Hoffman, S. Thakoor, J. Grill, B. Neyshabur, O. Bachem, A. Walton, A. Severyn, A. Parrish, A. Ahmad, A. Hutchison, A. Abdagic, A. Carl, A. Shen, A. Brock, A. Coenen, A. Laforge, A. Paterson, B. Bastian, B. Piot, B. Wu, B. Royal, C. Chen, C. Kumar, C. Perry, C. Welty, C. A. Choquette-Choo, D. Sinopalnikov, D. Weinberger, D. Vijaykumar, D. Rogozinska, D. Herbison, E. Bandy, E. Wang, E. Noland, E. Moreira, E. Senter, E. Eltyshev, F. Visin, G. Rasskin, G. Wei, G. Cameron, G. Martins, H. Hashemi, H. Klimczak-Plucinska, H. Batra, H. Dhand, I. Nardini, J. Mein, J. Zhou, J. Svensson, J. Stanway, J. Chan, J. P. Zhou, J. Carrasqueira, J. Iljazi, J. Becker, J. Fernandez, J. van Amersfoort, J. Gordon, J. Lipschultz, J. Newlan, J. Ji, K. Mohamed, K. Badola, K. Black, K. Millican, K. McDonell, K. Nguyen, K. Sodhia, K. Greene, L. L. Sjösund, L. Usui, L. Sifre, L. Heuermann, L. Lago, and L. McNealus. Gemma 2: Improving open language models at a practical size. *CoRR*, abs/2408.00118, 2024.

[25] M. Rivière, S. Pathak, P. G. Sessa, C. Hardin, S. Bhupatiraju, L. Hussenot, T. Mesnard, B. Shahriari, A. Ramé, J. Ferret, P. Liu, P. Tafti, A. Friesen, M. Casbon, S. Ramos, R. Kumar, C. L. Lan, S. Jerome, A. Tsitsulin, N. Vieillard, P. Stanczyk, S. Girgin, N. Momchev, M. Hoffman, S. Thakoor, J. Grill, B. Neyshabur, O. Bachem, A. Walton, A. Severyn, A. Parrish, A. Ahmad, A. Hutchison, A. Abdagic, A. Carl, A. Shen, A. Brock, A. Coenen, A. Laforge, A. Paterson, B. Bastian, B. Piot, B. Wu, B. Royal, C. Chen, C. Kumar, C. Perry, C. Welty, C. A. Choquette-Choo, D. Sinopalnikov, D. Weinberger, D. Vijaykumar, D. Rogozinska, D. Herbison, E. Bandy, E. Wang, E. Noland, E. Moreira, E. Senter, E. Eltyshev, F. Visin, G. Rasskin, G. Wei, G. Cameron, G. Martins, H. Hashemi, H. Klimczak-Plucinska, H. Batra, H. Dhand, I. Nardini, J. Mein, J. Zhou, J. Svensson, J. Stanway, J. Chan, J. P. Zhou, J. Carrasqueira, J. Iljazi, J. Becker, J. Fernandez, J. van Amersfoort, J. Gordon, J. Lipschultz, J. Newlan, J. Ji, K. Mohamed, K. Badola, K. Black, K. Millican, K. McDonell, K. Nguyen, K. Sodhia, K. Greene, L. L. Sjösund, L. Usui, L. Sifre, L. Heuermann, L. Lago, and L. McNealus. Gemma 2: Improving open language models at a practical size. *CoRR*, abs/2408.00118, 2024.

[26] K. Sakaguchi, R. L. Bras, C. Bhagavatula, and Y. Choi. Winogrande: An adversarial winograd schema challenge at scale. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 8732–8740. AAAI Press, 2020.

[27] M. Sap, H. Rashkin, D. Chen, R. L. Bras, and Y. Choi. Socialiqa: Commonsense reasoning about social interactions. *CoRR*, abs/1904.09728, 2019.

[28] A. Saparov and H. He. Language models are greedy reasoners: A systematic formal analysis of chain-of-thought. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023.

[29] A. Setlur, C. Nagpal, A. Fisch, X. Geng, J. Eisenstein, R. Agarwal, A. Agarwal, J. Berant, and A. Kumar. Rewarding progress: Scaling automated process verifiers for LLM reasoning. *CoRR*, abs/2410.08146, 2024.

[30] C. Snell, J. Lee, K. Xu, and A. Kumar. Scaling LLM test-time compute optimally can be more effective than scaling model parameters. *CoRR*, abs/2408.03314, 2024.

[31] Y. Sui, Y. Chuang, G. Wang, J. Zhang, T. Zhang, J. Yuan, H. Liu, A. Wen, S. Zhong, H. Chen, and X. B. Hu. Stop overthinking: A survey on efficient reasoning for large language models. *CoRR*, abs/2503.16419, 2025.

[32] O. Tafjord, B. Dalvi, and P. Clark. Proofwriter: Generating implications, proofs, and abductive statements over natural language. In C. Zong, F. Xia, W. Li, and R. Navigli, editors, *Findings of the Association for Computational Linguistics: ACL/IJCNLP 2021, Online Event, August 1-6, 2021*, volume ACL/IJCNLP 2021 of *Findings of ACL*, pages 3621–3634. Association for Computational Linguistics, 2021.

[33] A. Talmor, J. Herzig, N. Lourie, and J. Berant. Commonsenseqa: A question answering challenge targeting commonsense knowledge. In J. Burstein, C. Doran, and T. Solorio, editors, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4149–4158. Association for Computational Linguistics, 2019.

[34] H. Wang, W. Xiong, T. Xie, H. Zhao, and T. Zhang. Interpretable preferences via multi-objective reward modeling and mixture-of-experts. In Y. Al-Onaizan, M. Bansal, and Y. Chen, editors, *Findings of the Association for Computational Linguistics: EMNLP 2024, Miami, Florida, USA, November 12-16, 2024*, pages 10582–10592. Association for Computational Linguistics, 2024.

[35] P. Wang, L. Li, Z. Shao, R. Xu, D. Dai, Y. Li, D. Chen, Y. Wu, and Z. Sui. Math-shepherd: Verify and reinforce llms step-by-step without human annotations. In L. Ku, A. Martins, and V. Srikumar, editors, *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024*, pages 9426–9439. Association for Computational Linguistics, 2024.

[36] X. Wang, J. Wei, D. Schuurmans, Q. V. Le, E. H. Chi, S. Narang, A. Chowdhery, and D. Zhou. Self-consistency improves chain of thought reasoning in language models. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023.

[37] J. Wei, X. Wang, D. Schuurmans, M. Bosma, B. Ichter, F. Xia, E. H. Chi, Q. V. Le, and D. Zhou. Chain-of-thought prompting elicits reasoning in large language models. In S. Koyejo, S. Mohamed, A. Agarwal, D. 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.

[38] T. Xie, Z. Gao, Q. Ren, H. Luo, Y. Hong, B. Dai, J. Zhou, K. Qiu, Z. Wu, and C. Luo. Logic-rl: Unleashing LLM reasoning with rule-based reinforcement learning. *CoRR*, abs/2502.14768, 2025.

[39] A. Yang, B. Yang, B. Zhang, B. Hui, B. Zheng, B. Yu, C. Li, D. Liu, F. Huang, H. Wei, H. Lin, J. Yang, J. Tu, J. Zhang, J. Yang, J. Yang, J. Zhou, J. Lin, K. Dang, K. Lu, K. Bao, K. Yang, L. Yu, M. Li, M. Xue, P. Zhang, Q. Zhu, R. Men, R. Lin, T. Li, T. Xia, X. Ren, X. Ren, Y. Fan, Y. Su, Y. Zhang, Y. Wan, Y. Liu, Z. Cui, Z. Zhang, and Z. Qiu. Qwen2.5 technical report. *CoRR*, abs/2412.15115, 2024.

[40] A. Yang, B. Zhang, B. Hui, B. Gao, B. Yu, C. Li, D. Liu, J. Tu, J. Zhou, J. Lin, K. Lu, M. Xue, R. Lin, T. Liu, X. Ren, and Z. Zhang. Qwen2.5-math technical report: Toward mathematical expert model via self-improvement. *CoRR*, abs/2409.12122, 2024.

[41] Y. Ye, Z. Huang, Y. Xiao, E. Chern, S. Xia, and P. Liu. LIMO: less is more for reasoning. *CoRR*, abs/2502.03387, 2025.

[42] Z. Zeng, Q. Cheng, Z. Yin, B. Wang, S. Li, Y. Zhou, Q. Guo, X. Huang, and X. Qiu. Scaling of search and learning: A roadmap to reproduce o1 from reinforcement learning perspective. *CoRR*, abs/2412.14135, 2024.

[43] L. Zhang, A. Hosseini, H. Bansal, M. Kazemi, A. Kumar, and R. Agarwal. Generative verifiers: Reward modeling as next-token prediction. *CoRR*, abs/2408.15240, 2024.

[44] W. Zhang, S. Nie, X. Zhang, Z. Zhang, and T. Liu. S1-bench: A simple benchmark for evaluating system 1 thinking capability of large reasoning models. *arXiv preprint arXiv:2504.10368*, 2025.

[45] Y. Zhao, H. Yin, B. Zeng, H. Wang, T. Shi, C. Lyu, L. Wang, W. Luo, and K. Zhang. Marco-o1: Towards open reasoning models for open-ended solutions. *CoRR*, abs/2411.14405, 2024.

[46] C. Zheng, Z. Zhang, B. Zhang, R. Lin, K. Lu, B. Yu, D. Liu, J. Zhou, and J. Lin. Processbench: Identifying process errors in mathematical reasoning. *CoRR*, abs/2412.06559, 2024.

[47] T. Zheng, Y. Chen, C. Li, C. Li, Q. Zong, H. Shi, B. Xu, Y. Song, G. Y. Wong, and S. See. The curse of cot: On the limitations of chain-of-thought in in-context learning. *arXiv preprint arXiv:2504.05081*, 2025.

## A Performance of Selected RMs

To demonstrate that the RM issues identified in our experiments in Section §2 are not due to the selected RM's inherently low discriminative abilities, here we present the performance of our RM. For the two ORMs (e.g. ArmoRM-Llama3-8B and Skywork-Reward-Llama-3.1-8B), we report their performance on RewardBench [11] compared to other baselines in Table 3. For the two PRMs (e.g. Math-Shepherd-Mistral-7B-PRM and Skywork-o1-Open-PRM-Qwen-2.5-7B), we report their performance on ProcessBench [11] compared to other baselines in Table 4. From them, we can get that the performance of these models on relevant benchmarks is comparable to the advanced LLMs (e.g. gpt4), hence they are representative.

## B Additional Overall Experiments

In addition to the experiments in the main text, we also conduct the experiments in other settings.

Firstly, while the main text compares different RMs using BoN methods, we now replicate this comparison using the MCTS approach. Our settings are as follows:

- **SC:** Using the self-consistency method for comparison;
- **Reward:** Using the reward score as $f$ in MCTS (e.g. MCTS-Reward in §3.3);
- **Maj_vote:** Using the major voting as $f$ in MCTS (e.g. MCTS-SC in §3.3);
- **Q_value:** Using the sum of Q-value in each path as $f$ in MCTS;
- **N_greedy:** At each step, select the node with the most frequent visits N and perform a top-down greedy search on the tree to obtain the final selected path;
- **Q_greedy:** At each step, select the node with the highest Q-value and perform a top-down greedy search on the tree to obtain the final selected path;
- **Oracle:** The coverage of the MCTS method.

In addition, we also use the consistency of the final answer output by the policy model itself as the source of the reward, denoted as 'Self'. The results are demonstrated in Figure 10. We can conclude that: (1) Even with the MCTS framework, the improvement in model reasoning brought by the RM is still minimal, further validating our conclusions in §2. (2) In Skywork and Skyworko1, the average performance of Reward is the best among all scoring functions. Therefore, in the MCTS-related experiments presented in the main text, we default to using it as the scoring function $f$.

Secondly, we focus on math reasoning in the main text, here we repeat our experiments on other types of reasoning tasks. Specifically, for math reasoning, we select another dataset: AQuA [13]. For commonsense reasoning, we select WinoGrande (WINO) [26] and CSQA [33]; For logical reasoning, we select ProofWriter [32] and ProntoQA [28] The results are demonstrated in Figure 11, 12, 13, 14 and 15. Lastly, we only use discriminative RM in the main text. All of these results are consistent with the conclusion in the main text.

## C Additional Experiments on Question Difficulty Approximation

In the main text, we calculate the question difficulty with assuming oracle access to a ground truth. However, in real-world applications, we are only given access to test prompts and do not know the true answers. Thus, we need to find a function that effectively estimates the problem difficulty without requiring ground truth. Specifically, we propose the following functions:

- **Length:** The average length of all responses to the question;
- **Count:** The count of different answers to the question;
- **Null:** The number of responses that fail to correctly generate the answer.

We classify the problems according to the difficulty levels as outlined in the main text and calculate the above three metrics across different levels of problem difficulty to compare the degree of correlation. The results are illustrated in Figure 16, 17 and 18. We can observe that, comparatively, the Count function is most directly proportional to difficulty. Therefore, we use this function to estimate difficulty when designing the CRISP method in §4.1.

## D Additional Experiments across Different Difficulty Levels

In the main text, we only analyze the impact of question difficulty on the MATH dataset. To demonstrate the generalizability of our conclusions, we repeat this experiment on GSM8K [5] and Olympiadbench [9]. The former dataset contains 8.5K linguistically diverse elementary school math problems designed to evaluate arithmetic reasoning consistency, while the latter is an Olympiad-level bilingual multimodal scientific benchmark. Compared to MATH, the former is simpler, while the latter is more challenging. The results are illustrated in Table 5, 6 and 7. We can observe that the issues identified in Cl.1 are prevalent across various reasoning datasets.

## E Case Analysis of Sampling Numbers Experiment

We start with a case analysis to uncover the issues inherent in the reward model. In the analysis, we randomly select five questions from different methods and examine the correctness of answers as $n$ scales. If a question is answered correctly, it indicates that the RM can accurately distinguish the positive examples from the negative ones, otherwise, it cannot. The results of this experiment are demonstrated in Figure 19, from which we can deduce that: **As $n$ increases, LLMs can generate incorrect responses that become increasingly challenging for the reward model to differentiate.** For some cases (like index 3 and 4 in Figure 19), RM assigns the highest score to newly generated incorrect responses, transforming the originally correct answers into incorrect ones.

## F Cause Analysis of Temperature-Induced Accuracy Drop

We further conduct statistical analyses to uncover the reasons for this issue. For each $T$, we calculate the information entropy of incorrect answers across 16 samplings and report the distribution over 200 questions in Figure 21. As the temperature rises, the entropy for both models shows a gradually increasing trend, hence, the distribution of these negative samples becomes more random. This indicates that the policy model generates a greater number of low-frequency incorrect answers at higher temperatures. According to Cl.2, RM struggles to differentiate these negative examples from correct ones, leading to lower inference accuracy. This result not only elucidates the reasons behind the subpar performance of BoN under high diversity conditions but also further corroborates the inverse long-tail phenomenon of the RM.

## G Diversity Experiment on Exploration Constant

In MCTS, apart from the tree structure, the explore weight $c$ also plays a crucial role in balancing the trade-off between exploitation (i.e. choosing actions that are known to yield high rewards) and exploration. A higher value of $c$ encourages more exploration, increasing the weight of the uncertain actions in the UCB formula. A lower value of $c$ favors exploitation, as it prioritizes actions with known higher rewards. We compare the MCTS performance under different $c$ and present the result in Figure 22. We can observe that an excessively large $c$ reduces performance (e.g. $c = 10.0$), indicating that overly high sampling diversity impairs reasoning accuracy, which is consistent with Cl.3 in our main text.

## H Implementation Details in the Main Experiments

Here we provide a detailed account of the implementation specifics from the main experiments:

For Self-Consistency, we generate 32 samples and choose the major voting answer as the final prediction. For BoN, we set the temperature to 0.7 to control the diversity and choose the best answer from 32 samples. For BoN Weighted, we normalize the RM's scoring and use this score as a weight to conduct a weighted vote among different answers, selecting the final prediction. For MCTS, we set the rollout number to 16, the width to 5, the max depth to 5, and the explore weight to 0.1. For Beam Search, we set the Beam numbers to 8, the beam width to 5, and the max depth to 5.

For our method, we generate 16 samples with a temperature setting of 0.7 in the first iteration. In subsequent iterations, we set the sampling numbers to 8 for ORM, 4 for PRM, and the max depth to

3. In prefix extraction, for ORM, we select the top-1 path, for PRM, we select the top-2 paths. For the evaluation data, we sample 500 questions from GSM8K and MATH-500, while sampling 200 questions from OlympiadBench.

We release the prompts we use in Table 8, 9, 10, 11, 12 and 13. All experiments were conducted on NVIDIA A100 GPUs.

# I Ablation Study

To verify the effectiveness of each module of CRSIP, we conduct ablation experiments using 200 samples from GSM8K and MATH generated by Qwen2.5-3B. The experimental settings are as follows:

- **w/o Termination:** Disable the early termination condition based on the number of clusters;
- **w/o Aggregation:** Eliminate the clustering operation and use the score of each path instead of cluster scores for selection (similar to MCTS);
- **w/o Prefixing:** Cancel the operation of directly generating the remaining steps according to the prefix set, and instead generate intermediate nodes layer by layer (similar to MCTS and Beam).

Figure 23 shows the result of the ablation study. Removing each component leads to a decline in performance. Specifically, although w/o termination causes only a small drop, its inclusion not only improves performance but also reduces inference time.

# J Cost Analysis

We use Qwen2.5-3B as the policy model and Shepherd-PRM [35] as the reward model, and compare the inference time and token usage of different algorithms across various tasks. Each algorithm is required to perform 5 rollouts on the same devices, and the average is computed across all test instances. The results in Figure 24 demonstrate that our method is highly efficient. It achieves up to a **66%** reduction in inference time compared to advanced RM-integrated methods like MCTS and Beam Search, while preserving the runtime and token efficiency of the basic BoN method.

| Reward Model | Score | Chat | Chat Hard | Safety | Reasoning |
|---|---|---|---|---|---|
| Skywork-Reward-Llama-3.1-8B | 93.1 | 94.7 | 88.4 | 92.7 | 96.7 |
| ArmoRM-Llama3-8B-v0.1 | 89.0 | 96.9 | 76.8 | 92.2 | 97.3 |
| Gemini-1.5-pro-0514 | 88.1 | 92.3 | 80.6 | 87.5 | 92.0 |
| gpt-4-0125-preview | 84.3 | 95.3 | 74.3 | 87.2 | 86.9 |
| Meta-Llama-3-70B-Instruct | 75.4 | 97.6 | 58.9 | 69.2 | 78.5 |

Table 3: Comparison of RM's performance on RewardBench.

| Model | GSM8K | MATH | OlympiadBench | OmniMATH | Average |
|---|---|---|---|---|---|
| Shepherd-PRM-7B | 47.9 | 29.5 | 24.8 | 23.8 | 31.5 |
| Skyworko1-PRM-7B | 70.8 | 53.6 | 22.9 | 21.0 | 42.1 |
| Meta-Llama-3-70B-Instruct | 52.2 | 22.8 | 21.2 | 20.0 | 29.1 |
| Llama-3.1-70B-Instruct | 74.9 | 48.2 | 46.7 | 41.0 | 52.7 |
| Qwen2-72B-Instruct | 67.6 | 49.2 | 42.1 | 40.2 | 49.8 |

Table 4: Comparison of RM's performance on ProcessBench.

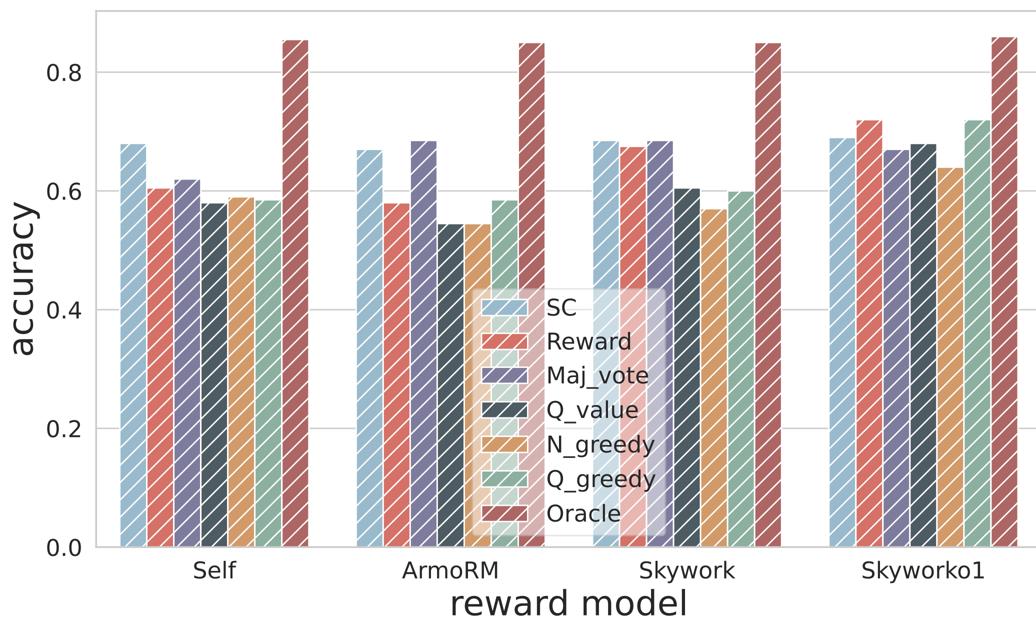



Figure 10: The performance of different reward models using the MCTS inference on the MATH dataset ($n$ = 16, Qwen-2.5-3B).

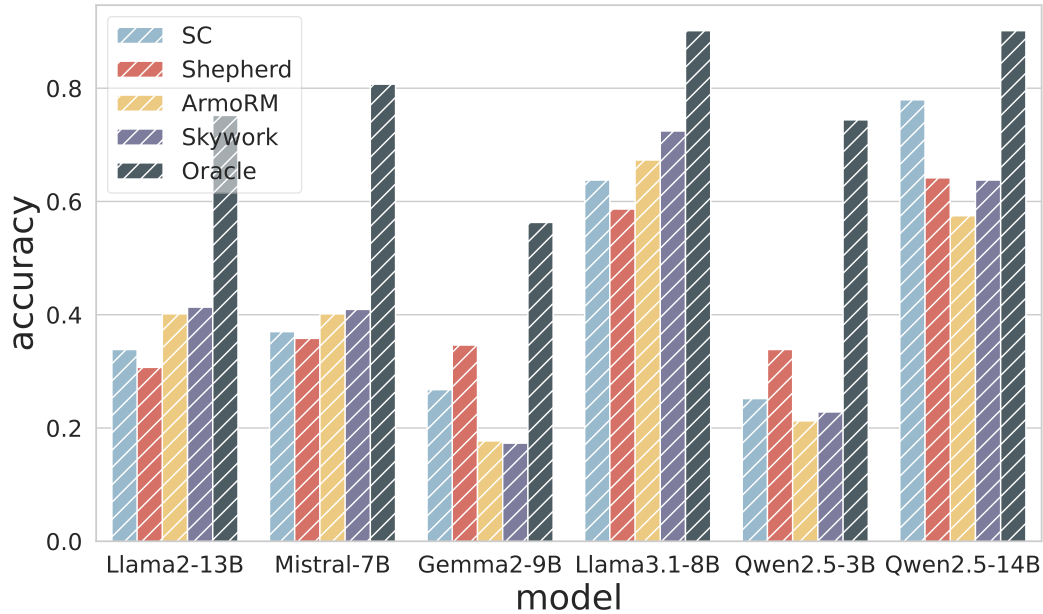



Figure 11: The performance of different policy models using various reward models for BoN inference on the AQuA dataset ($n$ = 10).

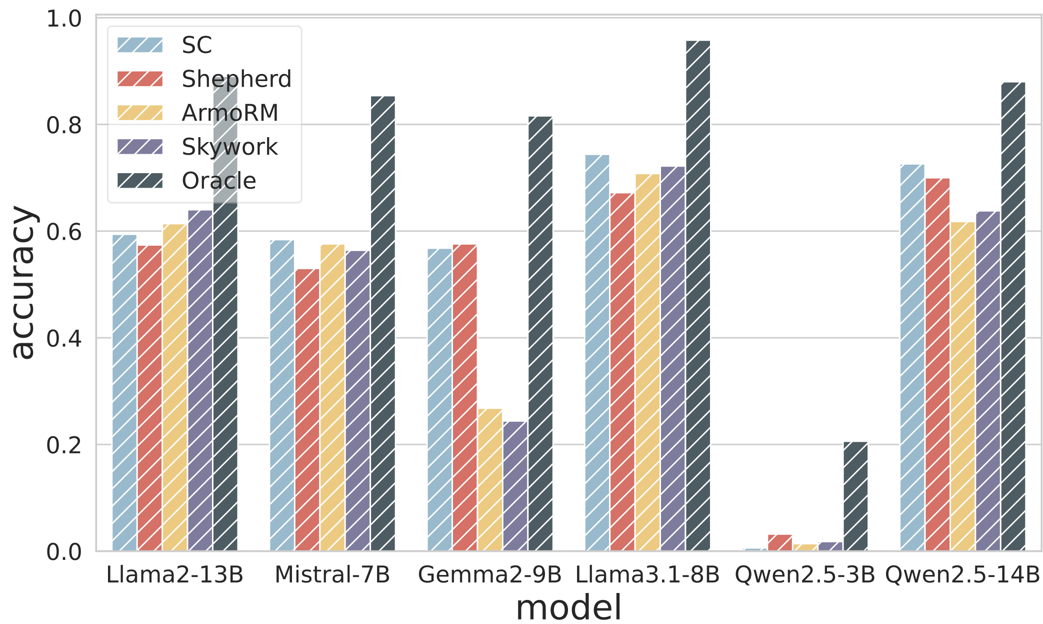



Figure 12: The performance of different policy models using various reward models for BoN inference on the WinoGrande dataset ($n$ = 10).

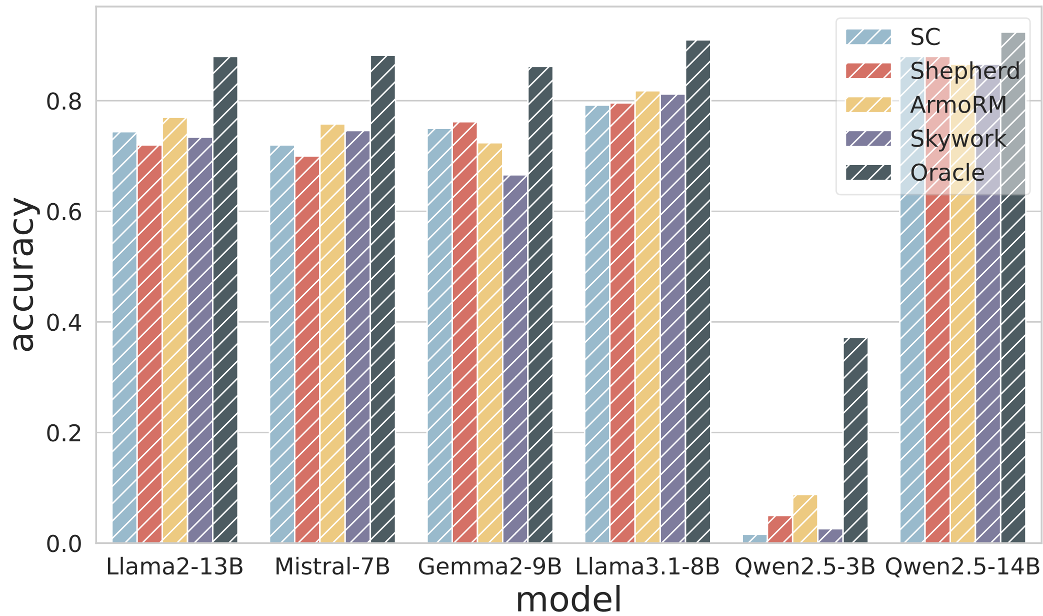



Figure 13: The performance of different policy models using various reward models for BoN inference on the CSQA dataset ($n$ = 10).

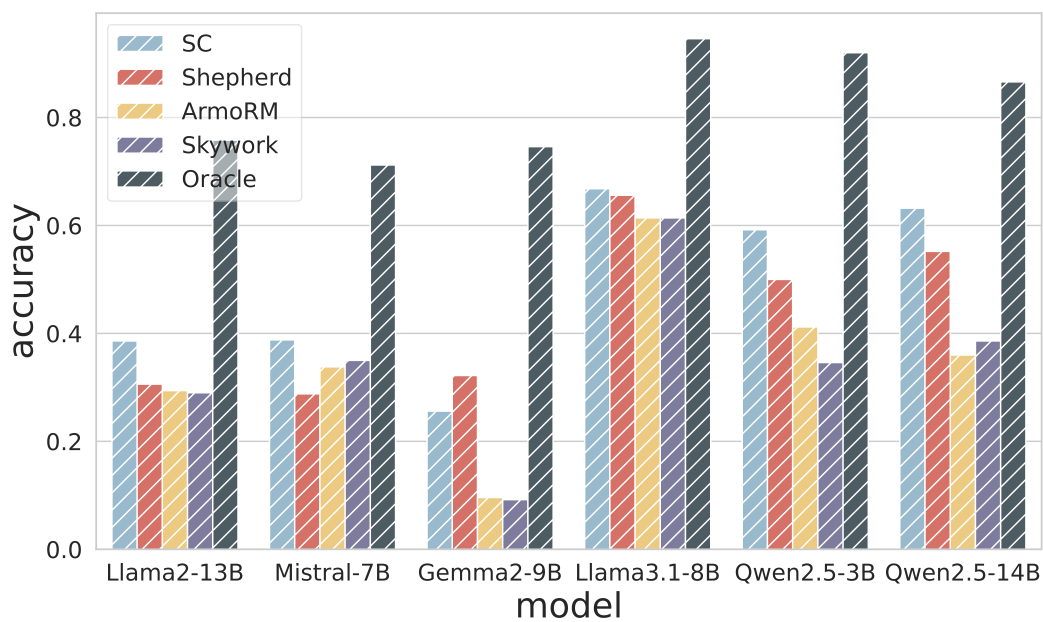



Figure 14: The performance of different policy models using various reward models for BoN inference on the ProofWriter dataset ($n$ = 10).

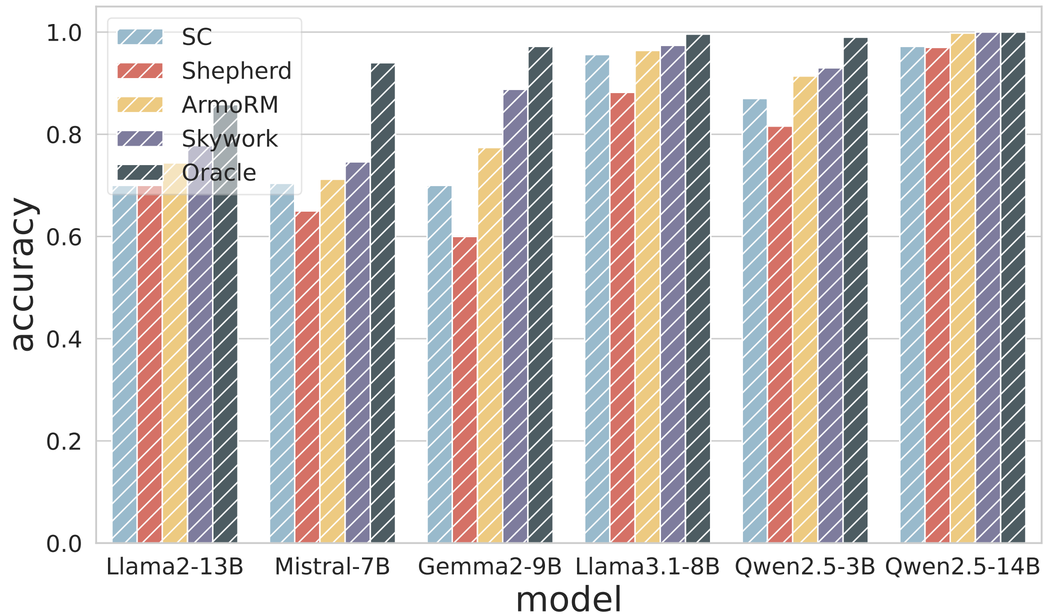



Figure 15: The performance of different policy models using various reward models for BoN inference on the ProntoQA dataset ($n$ = 10).

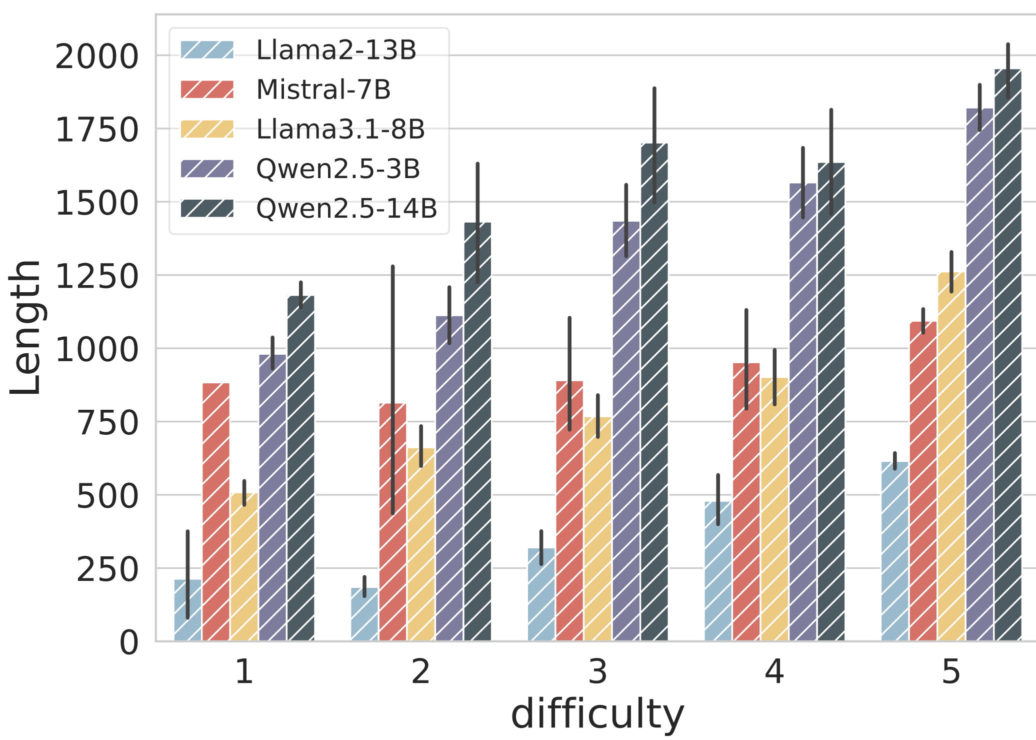



Figure 16: The correlation between output length and the question difficulty.

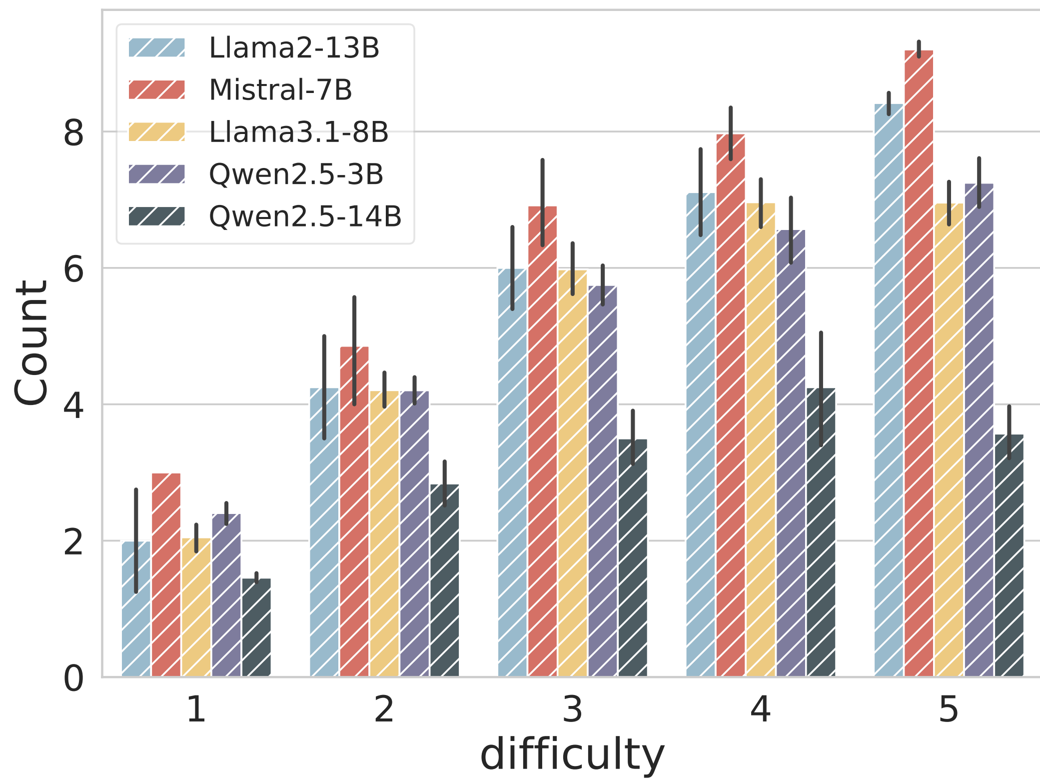



Figure 17: The correlation between the count of answers and the question difficulty.

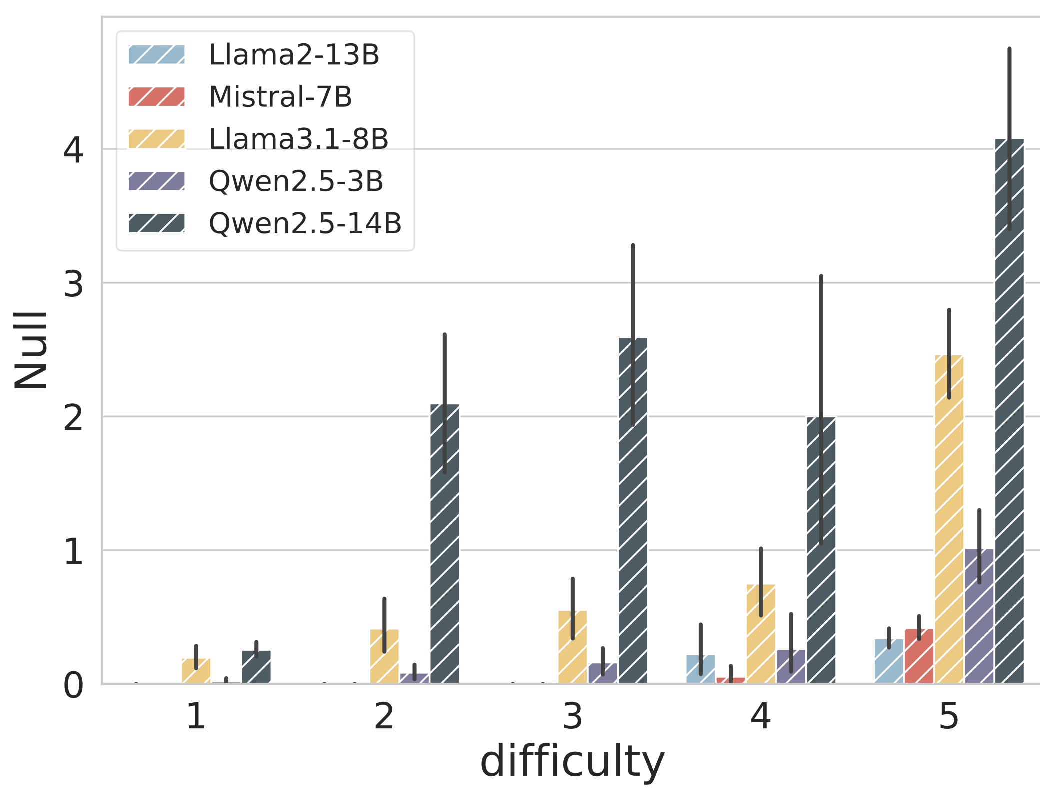
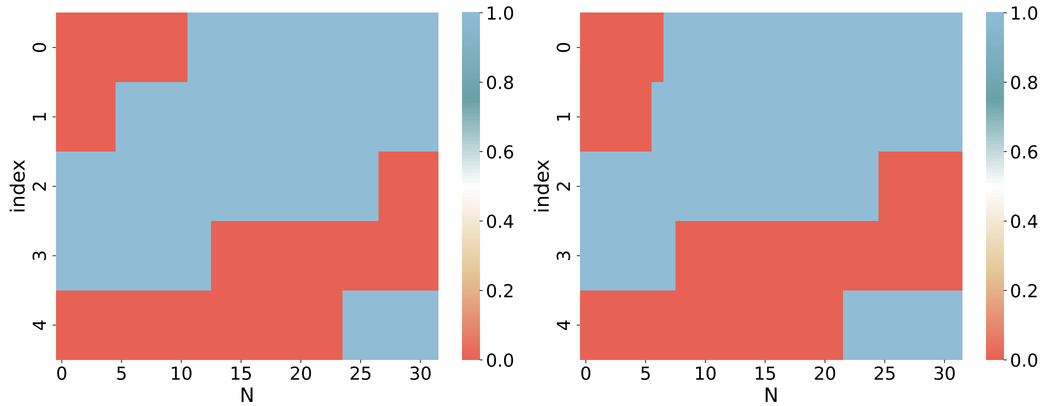



Figure 18: The correlation between the count of no answers and the question difficulty.




(a) BoN (b) MCTS

Figure 19: The variation in question answering correctness as the sampling number changes. Blue indicates a correct answer, while red indicates an incorrect answer.

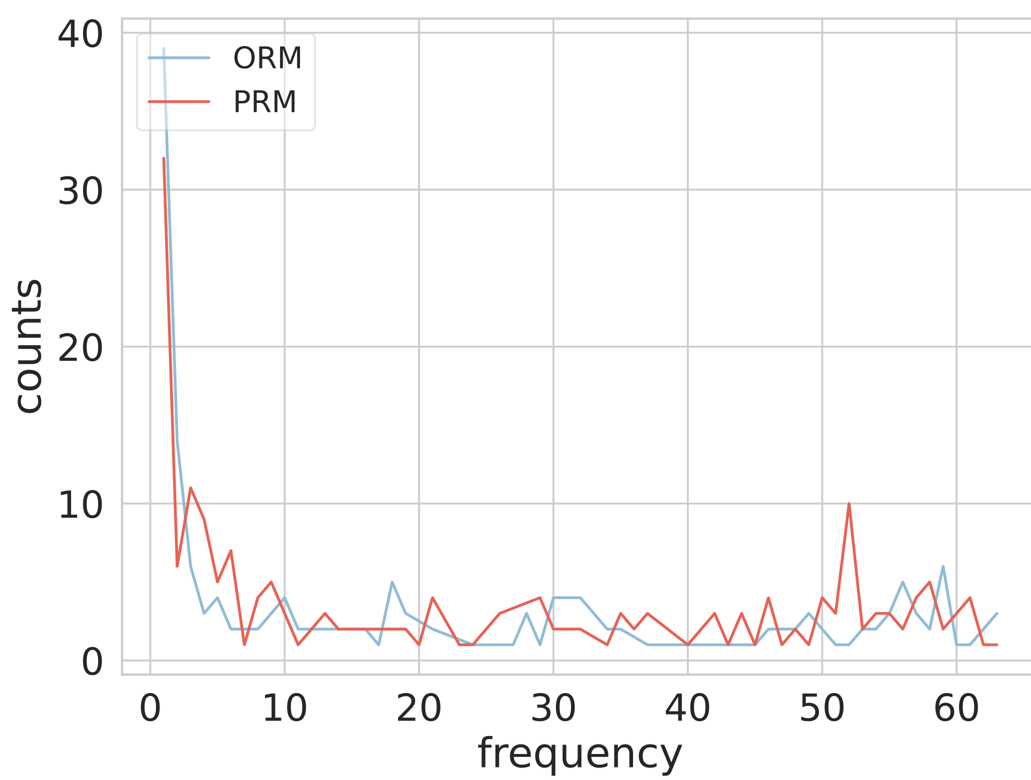



Figure 20: Frequency statistics of the highest-scored negative responses in MCTS.

Table 5: Comparison of performance across different difficulty levels on 500 samples of GSM8K (Qwen2.5-3B).

| Method | Level 1 | Level 2 | Level 3 | Level 4 | Level 5 | All |
|---|---|---|---|---|---|---|
| Self-Consistency(@128) | **99.7** | 96.8 | 80.0 | 34.6 | 3.2 | 83.2 |
| Best-of-128 + ORM | 98.0 | 87.1 | 72.0 | **65.4** | 12.9 | 83.8 |
| - SC | -1.7 | -9.7 | -8.0 | 30.8 | 9.7 | 0.6 |
| Best-of-128 + PRM | 98.3 | **100.0** | **96.0** | 57.7 | **30.6** | **87.8** |
| - SC | -1.4 | 3.2 | 16.0 | 23.1 | 27.4 | 4.6 |
| Count | 356 | 31 | 25 | 26 | 62 | 500 |

Table 6: Comparison of performance across different difficulty levels on MATH-500 (Qwen2.5-3B).

| Method | Level 1 | Level 2 | Level 3 | Level 4 | Level 5 | All |
|---|---|---|---|---|---|---|
| Self-Consistency(@128) | 98.8 | **98.8** | **80.4** | 49.2 | 5.3 | 65.4 |
| Best-of-128 + ORM | **99.4** | 92.8 | 69.6 | **58.5** | 17.3 | **67.8** |
| - SC | 0.6 | -6.0 | -9.8 | 9.3 | 12.0 | 2.4 |
| Best-of-128 + PRM | 88.3 | 71.1 | 78.6 | 53.8 | **21.8** | 62.2 |
| - SC | -10.5 | -27.7 | -1.8 | 4.6 | 16.5 | -3.2 |
| Count | 163 | 83 | 56 | 65 | 133 | 500 |

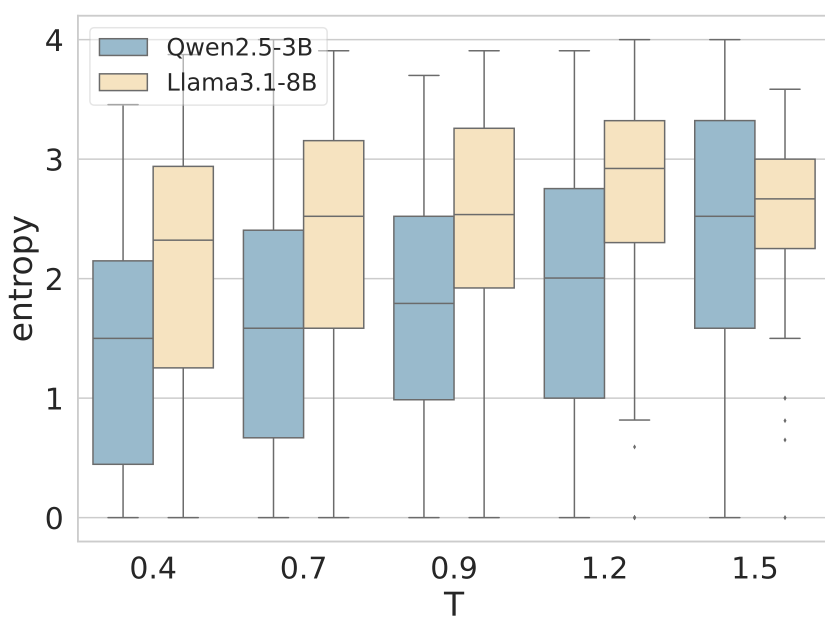
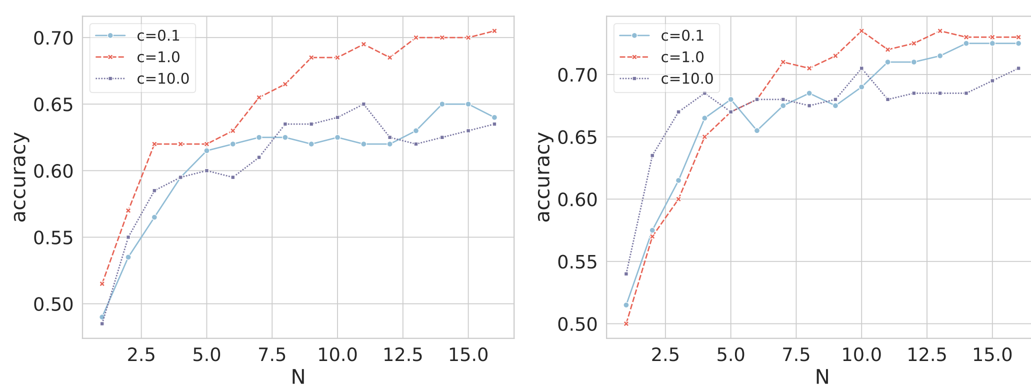



Figure 21: Information entropy of incorrect answers under different sampling temperatures.

Table 7: Comparison of performance across different difficulty levels on 200 samples of Olympiad-Bench (Qwen2.5-3B).

| Method | Level 1 | Level 2 | Level 3 | Level 4 | Level 5 | All |
|---|---|---|---|---|---|---|
| Self-Consistency(@32) | **100.0** | **100.0** | 64.3 | 50.0 | 0.8 | 30.5 |
| Best-of-32 + ORM | 100.0 | 80.0 | **78.6** | 40.0 | 3.8 | 31.5 |
| - SC | 0.0 | -20.0 | 14.3 | -10.0 | 3.0 | 1.0 |
| Best-of-32 + PRM | 100.0 | 100.0 | **78.6** | **50.0** | **6.9** | **34.0** |
| - SC | 0.0 | 0.0 | 14.3 | 0.0 | 6.1 | 3.5 |
| Count | 31 | 15 | 14 | 10 | 130 | 200 |




(a) BoN

(b) MCTS

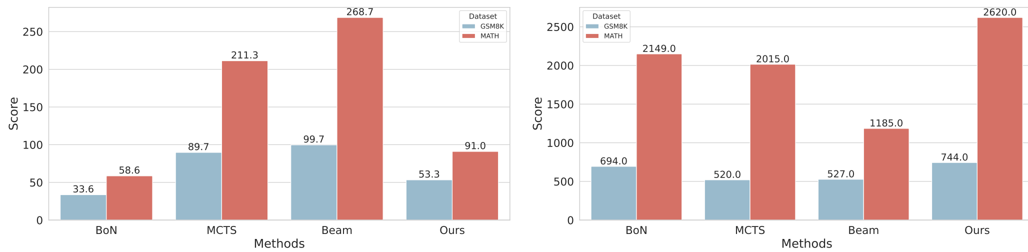
Figure 22: Performance comparison across different explore weight $c$ on Qwen2.5-3B.

**Algorithm 1** Clustered Reward Integration with Stepwise Prefixing

**Require:** Policy model $\mathcal{M}$, reward score $f$, question $q$, max steps $m$, sampling numbers $n$, top-$k$ parameter $k$

$i \leftarrow 0$
$\mathcal{R} \leftarrow \emptyset$ ▷ All responses
$\mathcal{P} \leftarrow \emptyset$ ▷ Response prefixes
$\mathcal{F} \leftarrow \emptyset$ ▷ Score map
$\mathcal{C} \leftarrow \emptyset$ ▷ Clusters
**while** $i < n$ **do**
    **if** $i = 0$ **then**
        $\mathcal{R} \leftarrow \mathcal{M}(q, n)$ ▷ Generate $n$ initial responses
        **if** $|\,\mathrm{Cluster}(\mathcal{R})| = 1$ **then**
            **return** $\mathcal{R}[0]$ ▷ Early exit if only one cluster
        **end if**
    **else**
        $\mathcal{R}_{\text{top}} \leftarrow \{\arg\max_{r \in \mathcal{C}_j} f(r) \,|\, \mathcal{C}_j \in \mathcal{C}_{\text{top}}\}$
        $\mathcal{P} \leftarrow \{r[:i{+}1] \mid r \in \mathcal{R}_{\text{top}}\}$ ▷ Truncate top responses
        $\mathcal{R} \leftarrow \mathcal{R} \cup \mathcal{M}(q, n, \mathcal{P})$ ▷ Decode more based on prefixes
    **end if**
    $\mathcal{C} \leftarrow \mathrm{Cluster}(\mathcal{R})$ ▷ Cluster current responses
    **for all** $\mathcal{C}_j \in \mathcal{C}$ **do**
        $\mathcal{F}(\mathcal{C}_j) \leftarrow \sum_{x \in \mathcal{C}_j} f(x)$ ▷ Assign cluster-wise reward
    **end for**
    $\mathcal{C}_{\text{top}} \leftarrow$ top-$k$ responses in $\mathcal{C}$ by $\mathcal{F}$
    $i \leftarrow i + 1$
**end while**
**return** $\mathcal{R}_{\text{top}}[0]$


Figure 23: Results of our ablation study.




(a) Time Consumption Comparison (s)

(b) Token Consumption Comparison

Figure 24: Results of our cost analysis.

Table 8: Prompts used to sample reasoning paths on the GSM8K dataset.

**Prompt**

Please act as a math teacher and solve the math problem step by step. At the final step, a conclusive answer is given in the format of “The answer is: \boxed{<ANSWER>}.”, where <ANSWER> should be a numeric answer.

**# Question:**
Mr. Ruther sold $\frac{3}{5}$ of his land and had 12.8 hectares left. How much land did he have at first?
**# Reasoning:**
Step 1: Mr. Ruther is left with $1 - \frac{3}{5} = \frac{2}{5}$ of his land.
Step 2: Since $\frac{2}{5}$ equals 12.8 hectares, then $\frac{1}{5} = \frac{12.8}{2} = 6.4$ hectares.
Step 3: Total land = $6.4 \times 5 = 32$ hectares.
Step 4: The answer is: $\boxed{32}$

**# Question:**
The Doubtfire sisters are driving home with 7 kittens adopted from the local animal shelter when their mother calls to inform them that their two house cats have just had kittens. She says that Patchy, the first cat, has had thrice the number of adopted kittens, while Trixie, the other cat, has had 12. How many kittens does the Doubtfire family now have?
**# Reasoning:**
Step 1: Patchy has had $3 \times 7 = 21$ kittens.
Step 2: Trixie has had 12 kittens. Total from both cats = $21 + 12 = 33$.
Step 3: Total kittens including adopted = $7 + 33 = 40$.
Step 4: The answer is: $\boxed{40}$

**# Question:**
After transferring to a new school, Amy made 20 more friends than Lily. If Lily made 50 friends, how many friends do Lily and Amy have together?
**# Reasoning:**
Step 1: Amy made $50 + 20 = 70$ friends.
Step 2: Total friends = $70 + 50 = 120$.
Step 3: The answer is: $\boxed{120}$
**# Question:**
{current question}
**# Reasoning:**

Table 9: Prompts used to sample reasoning paths on the MATH dataset.

**Prompt**

Please act as a math teacher and give step-by-step solutions to the user's questions. At the final step, a conclusive answer is given in the format of "The answer is: <ANSWER>.", where <ANSWER> should be a numeric answer.

**# Question:**

How many 3-letter words can we make from the letters A, B, C, and D, if we are allowed to repeat letters, and we must use the letter A at least once? (Here, a word is an arbitrary sequence of letters.)

**# Reasoning:**

Step 1: There are $4^3$ three-letter words from A, B, C, and D, and there are $3^3$ three-letter words from just B, C, and D.

Step 2: There must, then, be $4^3 - 3^3 = 64 - 27 = \boxed{37}$ words from A, B, C, and D containing at least one A.

Step 3: The answer is: $\boxed{37}$

**# Question:**

In the diagram, square $ABCD$ has sides of length 4, and $\triangle ABE$ is equilateral. Line segments $BE$ and $AC$ intersect at $P$. Point $Q$ is on $BC$ so that $PQ$ is perpendicular to $BC$ and $PQ = x$.

**# Reasoning:**

Step 1: Since $\triangle ABE$ is equilateral, we know that $\angle ABE = 60°$.

Step 2: Therefore,

$$\begin{aligned}\angle PBC &= \angle ABC - \angle ABE \\ &= 90° - 60° = 30°.\end{aligned}$$

Step 3: Since $AB = BC$, we know that $\triangle ABC$ is a right isosceles triangle and $\angle BAC = \angle BCA = 45°$.

Step 4: Then, $\angle BCP = \angle BCA = 45°$ and

$$\begin{aligned}\angle BPC &= 180° - \angle PBC - \angle BCP \\ &= 180° - 30° - 45° = \boxed{105°}.\end{aligned}$$

Step 5: The answer is: $\boxed{105}$

**# Question:**

Find the *positive* real number(s) $x$ such that

$$\frac{1}{2}(3x^2 - 1) = (x^2 - 50x - 10)(x^2 + 25x + 5).$$

**# Reasoning:**

Step 1: Write $a = x^2 - 50x - 10$ and $b = x^2 + 25x + 5$.

Step 2: Then the equation given becomes

$$\frac{a + 2b - 1}{2} = ab,$$

so $0 = 2ab - a - 2b + 1 = (a - 1)(2b - 1)$.

Step 3: Then $a - 1 = x^2 - 50x - 11 = 0$ or $2b - 1 = 2x^2 + 50x + 9 = 0$.

Step 4: The former has a positive root, $x = \boxed{25 + 2\sqrt{159}}$, while the latter does not.

Step 5: The answer is: $\boxed{25 + 2\sqrt{159}}$

**# Question:**

{current question}

**# Reasoning:**

Table 10: Prompts used to sample reasoning paths on the Olympiadbench dataset.

**Prompt**

Please act as a math teacher and give step-by-step solutions to the user's questions. At the final step, a conclusive answer is given in the format of "The answer is: \boxed{<ANSWER>}.", where <ANSWER> should be a numeric answer.

**# Question:**
Let $T$ be a rational number. Compute $\sin^2 \frac{T\pi}{2} + \sin^2 \frac{(5-T)\pi}{2}$.
**# Reasoning:**
Step 1: Note that $\sin \frac{(5-T)\pi}{2} = \cos\left(\frac{\pi}{2} - \frac{(5-T)\pi}{2}\right) = \cos\left(\frac{T\pi}{2} - 2\pi\right) = \cos \frac{T\pi}{2}$.
Step 2: Thus the desired quantity is $\sin^2 \frac{T\pi}{2} + \cos^2 \frac{T\pi}{2} = \boxed{1}$.
Step 3: The answer is: $\boxed{1}$

**# Question:**
Let $T = 11$. Compute the value of $x$ that satisfies $\sqrt{20 + \sqrt{T + x}} = 5$.
**# Reasoning:**
Step 1: Squaring both sides gives $20 + \sqrt{T + x} = 25$, so $\sqrt{T + x} = 5$.
Step 2: Squaring again gives $T + x = 25$, so $x = 25 - T = 14$.
Step 3: The answer is: $\boxed{14}$

**# Question:**
The sum of the interior angles of an $n$-gon equals the sum of the interior angles of a pentagon plus the sum of the interior angles of an octagon. Compute $n$.
**# Reasoning:**
Step 1: The sum of interior angles of an $n$-gon is $180°(n-2)$.
Step 2: A pentagon has sum $180°(5-2) = 540°$, and an octagon has sum $180°(8-2) = 1080°$.
Step 3: So $180(n-2) = 540 + 1080 = 1620$, hence $n - 2 = 9$, so $n = 11$.
Step 4: The answer is: $\boxed{11}$
**# Question:**
{current question}
**# Reasoning:**

Table 11: Prompts used to sample reasoning paths on the CSQA dataset.

| **Prompt** |
|---|
| Please act as a commonsense teacher and solve the commonsense reasoning problem step by step.<br><br>**# Question:**<br>Google Maps and other highway and street GPS services have replaced what?<br>**# Options:**<br>(A) atlas (B) mexico (C) countryside (D) united states (E) oceans<br>**# Reasoning:**<br>Step 1: Electronic maps and GPS services are the modern version of paper atlas.<br>Step 2: In that case, the atlas have been replaced by Google Maps and other highway and street GPS services.<br>Step 3: The answer is: **A**<br><br>**# Question:**<br>You can share files with someone if you have a connection to a what?<br>**# Options:**<br>(A) freeway (B) radio (C) wires (D) computer network (E) electrical circuit<br>**# Reasoning:**<br>Step 1: Files usually can be stored in the computers.<br>Step 2: In that case, we can share them over the Internet.<br>Step 3: Thus, if we connect to a computer network, we can share the file with others.<br>Step 4: The answer is: **D**<br><br>**# Question:**<br>The fox walked from the city into the forest, what was it looking for?<br>**# Options:**<br>(A) pretty flowers (B) hen house (C) natural habitat (D) storybook (E) dense forest<br>**# Reasoning:**<br>Step 1: Since the fox walk from the city into the forest, he may looks for something in the forest but not in the city.<br>Step 2: From all of the options, the natural habitat are usually away from cities.<br>Step 3: The answer is: **C**<br>**# Question:**<br>{current question}<br>**# Options:**<br>{current options}<br>**# Reasoning:** |

Table 12: Prompts used to sample reasoning paths on the SIQA dataset.

| **Prompt** |
| --- |
| Please act as a commonsense teacher and solve the commonsense reasoning problem step by step.<br><br>**# Question:**<br>Quinn wanted to help me clean my room up because it was so messy. What will Quinn want to do next?<br>**# Options:**<br>(A) Eat messy snacks (B) help out a friend (C) Pick up the dirty clothes<br>**# Reasoning:**<br>Step 1: Quinn wants to clean the room up.<br>Step 2: Picking up the dirty clothes is one way to clean the room.<br>Step 3: Thus, Quinn will want to pick up the dirty clothes next.<br>Step 4: The answer is: C<br><br>**# Question:**<br>Sydney had so much pent up emotion, they burst into tears at work. How would Sydney feel afterwards?<br>**# Options:**<br>(A) affected (B) like they released their tension (C) worse<br>**# Reasoning:**<br>Step 1: Crying is often a way to release tension.<br>Step 2: Sydney burst into tears at work.<br>Step 3: Thus, she would release the tension.<br>Step 4: The answer is: B<br><br>**# Question:**<br>Their cat kept trying to escape out of the window, so Jan placed an obstacle in the way. How would Jan feel afterwards?<br>**# Options:**<br>(A) scared of losing the cat (B) normal (C) relieved for fixing the problem<br>**# Reasoning:**<br>Step 1: The cat tried to escape so Jan needed to stop it to avoid losing the cat.<br>Step 2: Jan placed an obstacle in the way so the cat could not escape.<br>Step 3: The problem has been solved.<br>Step 4: Thus, Jan will feel relieved for fixing the problem.<br>Step 5: The answer is: C<br>**# Question:**<br>{current question}<br>**# Options:**<br>{current options}<br>**# Reasoning:** |

Table 13: Prompts used to sample reasoning paths on the LogiQA dataset.

| **Prompt** |
|---|
| Please act as a logical teacher and reason step by step to solve the logical reasoning problem.<br><br>**# Context:**<br>There are 90 patients with a disease T that is very difficult to treat and has taken the same routine drug. The patients were divided into two equal groups. The first group was given an experimental drug W, which is used to treat T, and the second group was given a placebo without W. Statistics ten years later showed that 44 people died in both groups, so the experimental drug was ineffective.<br>**# Question:**<br>Based on the above information, which of the following options, if correct, will best weaken the above argument?<br>**# Options:**<br>(A) Among the patients who died above, the average year of death in the second group was two years earlier than that in the first group. (B) Among the patients who died, the average life span of the second group was two years younger than that of the first group. (C) Among the above-mentioned living patients, the condition of the second group was more serious than that of the first group. (D) Among the above-mentioned living patients, those in the second group were older than those in the first group.<br>**# Reasoning:**<br>Step 1: Analyzing each option: A suggests drug W might extend life since the average death year in the drug W group is later than the placebo, directly challenging the drug's perceived ineffectiveness.<br>Step 2: B, similar to A, implies longer life in the drug W group but doesn't directly link to post-treatment lifespan.<br>Step 3: C indicates drug W may reduce disease severity but doesn't address lifespan or mortality, the main focus.<br>Step 4: D, about age differences, lacks direct relevance to drug effectiveness.<br>Step 5: Therefore, A most effectively weakens the argument against drug W's effectiveness.<br>Step 6: The answer is: A<br>**# Question:**<br>{current question}<br>**# Options:**<br>{current options}<br>**# Reasoning:** |

# NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: Please refer to the last paragraph of the Introduction.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Please refer to the 'Limitations & Future Work' paragraph in §4.4.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

Justification: This paper primarily relies on empirical studies rather than theoretical proofs.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We provide a detailed discussion of the implementation details for each experiment in the Appendix, including the datasets, model versions, and experimental parameters, etc.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: We include our experimental code in the supplementary materials (note that no new dataset was constructed in this work).

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: We provide a detailed discussion of the experimental settings of each experiment in the Appendix.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: We report confidence intervals for some experiments in our paper, such as the results in Figure 5.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We discuss the devices used in our work in Appendix H.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research conducted in this paper conforms in every respect to the NeurIPS Code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [NA]

    Justification: This work focuses on algorithmic improvements to reward-model-based inference and does not involve deployment, user interaction, or real-world data, thus posing no direct societal impact.

    Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: This paper does not involve the release of any models or datasets that pose a high risk of misuse. All experiments are conducted using standard benchmarks.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We cite all the original papers that produced the dataset we use for evaluation.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.

- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: The paper does not release new assets.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.

- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: [Yes]

    Justification: Since our work focuses on enhancing reasoning with LLMs, we explicitly specify the LLMs and their corresponding versions used in our experiments (see §2, §3.2 and §4.2).

    Guidelines:

    - The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.