ReasonerRank: Redefining Language Model Evaluation with Ground-Truth-Free Ranking Frameworks

Jiamu Zhang[1], Jiayi Yuan[1], Andrew Wen[1,2], Hoang Anh Duy Le[1], Yu-Neng Chuang[1], Soo-Hyun Choi[3], Rui Chen[3], Xia Hu[1]

[1] Rice University, [2] UTHealth Houston, [3] Samsung Electronics America

Correspondence: mz81@rice.edu, xia.hu@rice.edu

Abstract

Large Language Models (LLMs) are increasingly adopted across real-world applications, yet traditional evaluations rely on expensive, domain-specific ground-truth labels that are often unavailable or infeasible. We introduce a ground-truth-free evaluation framework focused on reasoning consistency and instruction following, shifting the emphasis from correctness—which is elusive without labels—to transparent, coherent, evidence-based reasoning. Each model response includes a direct answer, a structured multi-step explanation, and supporting evidence, all assessed via semantic similarity and output adherence checks. We further propose TopK-ReRank, which refines rankings by constructing a consensus answer from the most reliable models, reducing ambiguity across diverse reasoning styles. Experiments show that our framework outperforms existing label-free methods, including majority voting, triplet ranking, and peer-review approaches, providing a more interpretable and efficient alternative for evaluating LLMs in the absence of ground-truth labels. Our code is available at https://github.com/MorrisZJ/ReasonerRank.

1 Introduction

Large Language Models (LLMs) have demonstrated exceptional proficiency in a variety of language-related tasks across multiple domains, and their success has led to rapid adoption in real-world applications (Achiam et al., 2023; Yin et al., 2024; Yuan et al., 2024). This widespread use has spurred a growing interest in understanding and comparing their capabilities (Chang et al., 2024; Wang et al., 2024; Yuan et al., 2025). Traditionally, such evaluations rely on meticulously curated datasets with ground-truth labels to assess the correctness of model outputs. However, obtaining labeled data is expensive, time-consuming, and often requires domain expertise—particularly challenging in specialized areas like healthcare, law, and scientific research. This labeling bottleneck impedes efficient model evaluation and hence slows the integration of new LLMs into evolving application domains.

In many scenarios, labeling is simply infeasible. Emerging ("fresh") or zero-shot data can require subjective or multi-faceted responses, making it difficult to define a single "correct" answer. Dynamic environments—such as finance, cybersecurity, or breaking news—can also evolve too rapidly for human annotators to keep pace. Consequently, researchers urgently need methods to evaluate LLMs without relying on ground-truth data, a setting that raises fundamental questions about how to gauge "better" or "worse" performance when correctness itself remains elusive. Existing label-free approaches, including LLM-as-a-judge (Zheng et al., 2024b), multi-agent debate (Chan et al., 2024), and majority voting (a.k.a. the most common answer (Mienye and Sun, 2022)) among peer models, can reduce human labor but introduce new pitfalls. Voting schemes may embed biases, and LLM-as-a-judge can hallucinate or impose arbitrary standards, thus lacking transparent or reliable criteria for discerning which outputs are truly more coherent, logical, or helpful.

In this paper, we propose a new **ground-truth-free evaluation framework** for comparing LLMs, centered on reasoning consistency and instruction-following ability rather than on correctness—an inherently inaccessible notion without labeled data. Our approach is motivated by how humans intuitively evaluate responses in real-world contexts: rather than focusing exclusively on the final answer, we also consider whether the reasoning process is transparent, coherent, and grounded in evidence. Building on insights from cognitive science and argumentation theory, we show that logical consistency and clarity of thought often serve as more reliable indicators of model quality compared to

Findings of the Association for Computational Linguistics: ACL 2025, pages 13623–13639
July 27 - August 1, 2025

merely using a single "correct" label.

Concretely, we require each model to produce: *(i) a direct answer, (ii) a structured, multi-step reasoning explanation, and (iii) supporting evidence.* We utilize our proposed semantic similarity-based metric to assess and rank models' reasoning ability, then weight it with instruction-following ability to get the raw "reasoner" scores. Finally, we introduce TopK-ReRank, an adaptive method that generates a refined "consensus answer" through majority agreement among the most reliable models. This step mitigates the ambiguity arising from different reasoning styles, making it easier to identify and reward genuinely coherent and well-supported arguments. Thorough experiments and ablation studies demonstrate the effectiveness and robustness of our framework in comparing LLMs without relying on ground-truth labels.

2 Motivation

2.1 When Quick Evaluation Is Necessary

Obtaining labeled data is problematic in many real-world scenarios, but evaluating LLMs is still necessary. Because relying on manual annotations slows the progress and creates bottlenecks, it is essential to develop label-free evaluation methods. Several key challenges highlight this need.

- **Emerging or Zero-Shot Tasks:** LLMs are frequently applied to novel datasets and tasks where ground-truth labels do not yet exist. Waiting for manual annotations delays deployment.
- **Subjective or Ambiguous Responses:** Some tasks, such as creative writing and legal interpretations, involve responses that even domain experts may debate. A single "correct" label is neither feasible nor meaningful.
- **Dynamic and Rapidly Changing Fields:** In fast-evolving domains like finance, cybersecurity, or breaking news, information updates too quickly for human annotators to keep pace. A static ground-truth dataset quickly becomes obsolete, requiring adaptable evaluation methods.

2.2 From Snail-Paced Humans to Hallucinating LLM Judges: Why Existing Label-Free LLM Evaluations Are a Mess

LLM-as-a-judge A widely used shortcut in LLM evaluation is the LLM-as-a-Judge approach, where a supposedly stronger model is used to evaluate and rank the response of other models (Zheng et al., 2024b). This seems efficient, but when the task demands nuanced reasoning, the cracks begin to show. LLM-as-a-judge is prone to hallucinations that confidently justify incorrect responses with elaborate yet fabricated reasoning. What's more, in settings like ours—where the optimal model is unknown—this method becomes problematic. Selecting a model to serve as the evaluator inherently assumes that it is a reliable judge, thereby introducing unverified biases into the process.

Most Common Answer (MCA) The MCA treats the most frequent response among LLMs as the "ground truth," assuming consensus implies correctness (Mienye and Sun, 2022). This works for simple factual queries but easily fails on complex tasks, because models with shared biases tend to amplify incorrect answers instead of correcting them. MCA also suppresses diversity, penalizing less common yet insightful responses while rewarding conformity over correctness (see Section 4.2.1).

Peer Review Peer review ranking framework, such as PiCO (Ning et al., 2025), refine the MCA/majority voting by weighting peer evaluations and assigning greater influence to models deemed more capable. While this improves ranking stability, it assumes that stronger models make better judges, which is fragile when biases are shared. Instead of correcting errors, agreement amplifies them, creating a self-reinforcing bias loop. This method also faces a cold-start problem, where the initial bias-weighted scores might influence future rankings, reinforcing early misjudgments as well as gradually reducing evaluation diversity.

Triplet Ranking Triplet Ranking ("SelfRank" in the experiment) ranks the models through triplet-based comparisons, iteratively identifying the weakest performer (Dhurandhar et al., 2024). However, if two models share biases, they can systematically overrule the third, leading to a "Triplet Kangaroo Court" effect where rankings reinforce errors rather than correct them. Like PiCO, this method employs reputation-weighted influence, amplifying early misjudgments and locking models into positions shaped more by perception than capability (see Section 4.2.2).

2.3 Our Intuition

With ground truth, traditional evaluation metrics for LLMs primarily focus on the correctness of the answer using predefined ground-truth labels, without considering the underlying reasoning process that

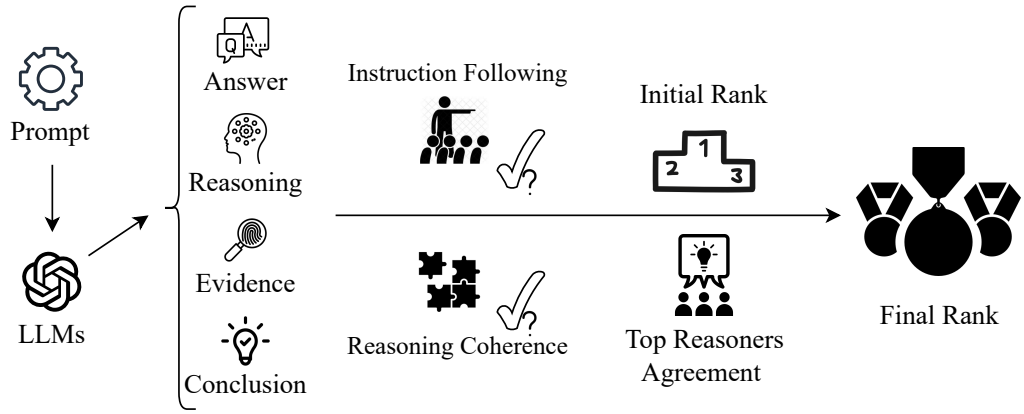

Figure 1: Overview of the ReasonerRank Framework. Given a prompt, each LLM generates a structured response consisting of an answer, multi-step reasoning, supporting evidence, and a conclusion. These components are then evaluated for instruction-following and reasoning coherence to identify a subset of top-performing reasoners. The pipeline then computes the agreement among the top-performing reasoners on each question to refine the rankings, resulting in a consensus-based final ranking of LLMs without relying on ground-truth labels.

led to a response. In contrast, human judgment in evaluating responses to questions inherently prioritizes logical consistency and semantic coherence. That is, people do not merely ask, "What is your answer?" but also, "Why? Why does this make sense?". It is our inherent human intuition that explanation, justification, and logical structure serve as key indicators of the credibility of a statement.

Extensive research in cognitive science already provides strong evidence supporting this intuition. Dual-process theories of reasoning (Evans and Stanovich, 2013) argue that people use fast, intuitive judgments and slow, analytical evaluations when assessing arguments. Notably, when individuals are motivated to evaluate or defend a position, such as in debates or collaborative problem solving, they tend to rely more on reasoning, because reasoning evolved primarily as a social tool for persuasion (Mercier and Sperber, 2011).

These insights suggest that the credibility and reliability of an answer depend not only on the correctness but also on the reasoning process that supports it. However, in many real-world scenarios, including our setting, there is no definitive ground truth against which responses can be evaluated. This motivates a fundamental transition in evaluation: **rather than relying on correctness relative to inaccessible ground truth, we prioritize assessing logical consistency and semantic coherence of reasoning as more reliable indicators of an LLM's ability to generate well-founded responses**. Building on these principles, we propose a ranking approach that evaluates LLMs based on their instruction following and reasoning quality. By measuring how well LLMs follow the instructions clear, logically structured, and semantically meaningful explanations, we obtain a more generalizable ranking of models—one that prioritizes reasoning over mere answer correctness. This approach reflects how humans naturally judge responses and provides a more robust framework for ranking LLMs when ground-truth labels are absent.

In the following section, we formally define this problem and present a framework to evaluate LLMs without having access to the ground truth.

3 Proposed Method

3.1 Ground-Truth-Free Ranking Protocol

To address the challenge of ranking a set of LLM candidates on a dataset without access to ground-truth labels, we propose a ground-truth-free ranking protocol called **ReasonerRank**. The overall workflow of our framework is illustrated in Figure 1. In the following section, we detail each component of the pipeline.

3.1.1 Instruction Following

To approximate the ideal ranking, we must first ensure that model responses are evaluated based on well-defined and structured criteria. Prior research has demonstrated a strong correlation between a model's general performance and its instruction-following ability. In *Large Language Model Instruction Following: A Survey of Progresses and Challenges*, the authors state:

> "However, the performance of instruction following highly relies on both model

and task scale: A larger LLM (or pre-training with more tokens) tuned on more diverse tasks can achieve significantly better few/zero-shot performances on the downstream tasks." (Lou et al., 2024)

This suggests that models with **better general capabilities**—whether due to more diverse training data or improved instruction tuning—tend to **exhibit stronger instruction-following abilities**. Consequently, models that perform well on general benchmarks also tend to follow instructions more reliably, reinforcing the idea that instruction-following ability is an integral aspect of overall model competence. Therefore, rather than assessing fluency or grammatical correctness, our first step is to judge **Instruction Following** at the language level, which verifies whether a model's response adheres to the prescribed structure given in the prompt. Each question is accompanied by a prompt template that instructs the LLM to generate a response following a designated format, which includes: (1) a direct answer, (2) a structured multi-step reasoning process, (3) a set of supporting evidence, and (4) a concise conclusion.

A model receives a score deduction of 1 for each missing element in the designated format. To be more specific, we verify whether the **Answer** and **Conclusion** fields are nonempty, ensure that all intermediate reasoning steps requested in the prompt are provided and non-empty, and check if we have enough supporting evidence for the reasoning steps. If fewer than three reasoning and evidence components are present, such a model is considered a bad instruction follower and will receive a penalty of 1 for each missing one. These penalties are aggregated into a component score that reflects how well the response follows the prescribed structure. Models with incomplete responses are strongly penalized by this score in the ranking. This strict structural check ensures that only models with properly formatted and complete responses are considered in the top rank and benefit from a deeper reasoning quality evaluation in our entire framework.

3.1.2 Reasoning Consistency

To evaluate the quality of reasoning in model-generated responses, we introduce a **reasoning consistency metric**, which captures the logical coherence between reasoning steps and the alignment between reasoning and supporting evidence.

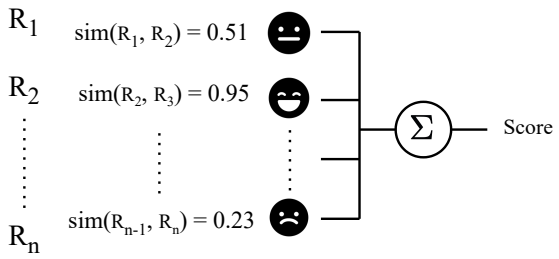

Figure 2: Internal Logical Coherence Computation. For a given response, the semantic similarity between each pair of consecutive reasoning steps is computed. These similarity scores are then aggregated to produce an overall reasoning coherence score, reflecting the logical consistency across the multi-step reasoning process.

Internal Logical Coherence. We define reasoning consistency as the degree to which consecutive reasoning steps are semantically aligned (see Figure 2). Given a sequence of reasoning steps $R = \{r_1, r_2, \ldots, r_n\}$, we compute the consistency score as:

$$C_{\text{reasoning}} = \frac{1}{n-1} \sum_{i=1}^{n-1} \text{sim}(r_i, r_{i+1}), \quad (1)$$

where $\text{sim}(\cdot, \cdot)$ represents an embedding-based similarity function. A higher score indicates a logically connected reasoning process, while a lower score suggests disjointed or inconsistent reasoning.

Reasoning-Evidence Consistency. To ensure that reasoning steps are well-supported by factual information, we assess the semantic alignment between each reasoning step and its most relevant supporting evidence (see Figure 3). Given a set of reasoning steps $R = \{r_1, r_2, ..., r_n\}$ and a set of supporting evidence $E = \{e_1, e_2, ..., e_m\}$, we define the reasoning-evidence consistency score as:

$$C_{\text{evidence}} = \frac{1}{n} \sum_{i=1}^{n} \max_j \text{sim}(r_i, e_j), \quad (2)$$

where each reasoning step r_i is matched to its most similar evidence e_j based on the similarity function $\text{sim}(\cdot, \cdot)$ (same as what we use in C_{evidence}, details see Section A.1.6). A higher score indicates that reasoning is well-grounded in evidence, while a lower score suggests weak justification.

In general, the penalty score for each component is numerically computed as discussed previously and then directly summed to obtain an overall quality score for each model output. We do not incorporate any weight of each quality measurement.

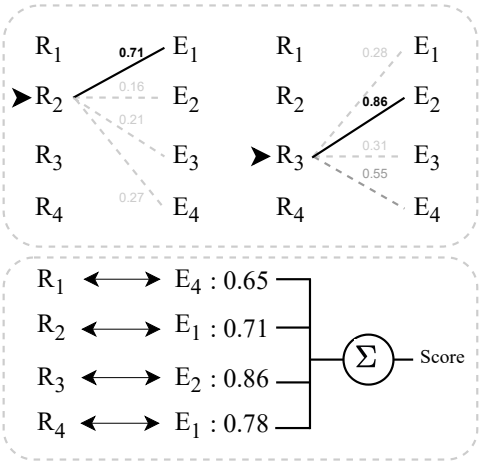

Figure 3: Reasoning-Evidence Consistency Computation. For each reasoning step, we compute its semantic similarity with all evidence and select the one with the highest similarity score. The score of all best-aligned reasoning-evidence pairs is then aggregated to yield a final consistency score, reflecting how well the reasoning steps are grounded in the supporting evidence.

3.2 TopK-ReRank: The Collective Wisdom of Top Reasoners Defines the Truth

While evaluating reasoning consistency ensures that logically coherent and well-supported responses are ranked higher, it does not fully eliminate uncertainty in the ranking process. In practice, model responses may still vary due to different reasoning styles, uncertainty in factual grounding, or even inconsistencies in evaluation metrics.

To further refine model rankings, we introduce **TopK-ReRank**, a method that **leverages the collective wisdom of the most reliable reasoners to define the truth**. Rather than relying on predefined correctness labels, TopK-ReRank constructs a **consensus answer** by identifying the most frequently agreed-upon response among the top-ranked models. This **adaptive reference answer** serves as a stable evaluation target, allowing us to measure how well each model aligns with the best reasoners.

3.2.1 Formal Definition of TopK-ReRank

Given an initial ranking π_0 obtained via reasoning consistency scores, we define the **TopK-ReRank** process as follows:

1. Select the top K models from π_0 based on their performance on instruction following and reasoning consistency.
2. Construct a **Consensus Answer** $A_{\text{consensus}}$ by majority voting over the final answers of the top K models.
3. Rank models based on their agreement with this consensus answer.

Mathematically, the consensus answer is determined as follows:

$$A_{\text{consensus}} = \arg\max_{A} \sum_{i=1}^{K} \mathbf{1}(A_i = A), \quad (3)$$

where $\mathbf{1}$ is an indicator function that equals 1 when its condition is true and 0 otherwise, and A_i is the final answer of model M_i, and the consensus answer is the most frequent one among the top K models. Each model is then ranked based on its agreement with the consensus answer.

By leveraging the most reliable reasoning outputs, **TopK-ReRank** enhances ranking stability and reduces noise introduced by model inconsistencies. Unlike traditional evaluation methods that rely on external correctness labels, this approach allows **truth to emerge dynamically from the best available reasoning patterns**. This ensures that models producing answers that align with the strongest reasoning outputs are consistently ranked higher, reinforcing the connection between reasoning quality and general model performance.

4 Experiments

4.1 Experiment Coverage

4.1.1 Datasets and Candidate LLMs

We conduct experiments on three benchmark datasets: MMLU (Massive Multitask Language Understanding) (Hendrycks et al., 2021), BBH (Big Bench Hard) (Srivastava et al., 2023), and GSM8K (Cobbe et al., 2021) (further details see Section A.1.1). Although our ranking method does not rely on ground-truth labels, we evaluate it on labeled datasets to quantitatively assess its effectiveness by comparing its output rankings against performance-based rankings derived from known answers. This allows us to benchmark our ground-truth-free approach against other ground-truth-free methods while using accuracy-aligned rankings as a proxy gold standard. We also select diverse sets of LLMs from leading research organizations, ensuring a comprehensive comparison between architectures, parameter scales, and training methodologies (Section A.1.2).

4.1.2 Evaluation Metrics

In this section, we define the metrics used to evaluate ranking alignment, focusing on overall agree-

ment and top-ranked items. We use Rank-Biased Overlap (RBO) and Set Precision at K (SP@K). For simplicity, we define rankings S and T to be the estimated ranking and target (ground-truth) ranking. Due to the space limit, here we provide an example of our benchmarked LLMs' performance and the resulting golden ranking we use to evaluate ground-truth-free ranking methods. More results will be released in our official repository.

Table 1: Golden Ranking of LLMs we use on MMLU Dataset (by accuracy).

Rank	Model	Accuracy (%)
1	Qwen2.5-72B-Instruct-Turbo	81.62
2	Llama-3.1-70B-Instruct-Turbo	80.31
3	Llama-3.3-70B-Instruct-Turbo	79.93
4	GPT-4o-Mini	75.59
5	Gemma-2-27B-IT	74.97
6	Mixtral-8x22B-Instruct-v0.1	73.58
7	Qwen2.5-7B-Instruct-Turbo	71.22
8	Mixtral-8x7B-Instruct-v0.1	67.67
9	Llama-3.1-8B-Instruct-Turbo	66.54
10	GPT-3.5-Turbo	64.63
11	Llama-3-8B-Instruct-Lite	61.61
12	Mistral-7B-Instruct-v0.2	58.97
13	Mistral-7B-Instruct-v0.3	58.94
14	Llama-3.2-3B-Instruct-Turbo	58.60
15	Llama-2-7B-Chat-HF	42.36
16	Gemma-2B-IT	39.59

Rank-Biased Overlap (RBO) RBO (Webber et al., 2010) Given rankings S and T, it is defined as:

$$RBO(S,T) = \frac{1}{|S|}\sum_{d=1}^{|S|} A_d \tag{4}$$

where A_d is the agreement at depth d (see section A.2.1 for more details).

Set Precision at K (SP@K) SP@K evaluates the overlap between the top-K models from two rankings, focusing on model selection rather than exact order. Given rankings S, T and hyperparameter K, it is defined as:

$$SP@K(S,T) = \frac{|S_{[1:K]} \cap T_{[1:K]}|}{K}, \tag{5}$$

where $S_{[1:K]}$ and $T_{[1:K]}$ are the top-K elements in their respective rankings. This metric captures agreement in identifying top-performing models (see Section A.2.2 for more details). We set $K = 3$ in our experiments.

4.2 Baseline Ground-Truth-Free Ranking Methods

4.2.1 Majority Voting: MCA

In our implementation, we count the frequency of each answer in all participating models for a given input. The answer that receives the highest frequency is then designated as the ground-truth or consensus answer. Formally, given n candidate LLMs and their respective predictions for a specific question instance, the Majority Voting approach computes:

$$a_{\text{majority\_vote}} = \arg\max_{a \in A} \sum_{i=1}^{n} \mathbf{1}(a_i = a), \tag{6}$$

This method is robust in scenarios where individual model errors are uncorrelated, and thus the majority consensus is more likely to be accurate than any single model prediction. However, its effectiveness can be limited if only a small group of models are the "top reasoner". In our multiple-choice question setting in the table of our experiment, the majority vote strategy is referred to as the Most Common Answer (MCA, see Table 2).

4.2.2 Triplet Ranking: SelfRank

Greedy Triplet Ranking (GTR) Greedy Triplet Ranking begins with an arbitrary triplet of models. For each triplet of models, $\{A, B, C\}$ for example, one of the third model, like $\{C\}$, serves as a judge by comparing the outputs of the other two, like $\{A, B\}$, and selecting the preferred one. In each iteration, the model that loses the most pairwise comparisons within the triplet is eliminated and replaced by the next unranked model. Finally, one of the top two models is randomly selected to serve as the judge to evaluate the responses of the remaining models and produce the final ranking (Dhurandhar et al., 2024).

Full Triplet Ranking (FTR) Different from Greedy Triplet Ranking, FTR examines all possible combinations of three models, assigning each a reputation score determined by how frequently it outperforms the others in these triplet comparisons (Dhurandhar et al., 2024).

4.3 Main Results

We present abbreviated evaluation results for ranking 14 LLM candidates on MMLU, BBH, and GSM8K. The performance metrics are averaged across all included subsets within each dataset.

From Table 2, we conclude that out method consistently demonstrates SOTA performance across all three datasets, which span diverse domains and difficulty levels. Among the baselines, MCA outperforms both Greedy Triplet Ranking and Full Triplet Ranking on MMLU and BBH in terms of both RBO and SP@3, with a particularly large performance gap observed on the BBH dataset.

For GSM8K, Full Triplet Ranking achieves the best performance among the baselines; however, its RBO value is only 0.6078 (close to 0.5), suggesting that its ranking agreement is only marginally better than random guessing. This raises concerns about the overall reliability of the Full Triplet Ranking method, even when it appears stronger on this dataset (full results in Section A.3.2).

Why Does TopK-ReRank Hold Up Better? Table 2 shows that SelfRank experiences a substantial performance drop from MMLU to BBH and GSM8K comparing to other methods. This abrupt decline suggests that SelfRank's triplet-based ranking approach struggles when there is a wide performance disparities between models. While it performs reasonably well in MMLU, where models exhibit relatively comparable performance, its stability deteriorates in BBH, where some models fail entirely while others perform significantly better. This suggests that SelfRank is more vulnerable to extreme performance variation, and thus leads to inconsistent rankings in more challenging tasks.

Despite the challenges from GSM8K and BBH, ReasonerRank remains robust and consistently outperforms or matches baselines. Its resilience to performance divergence enables accurate ranking even with large gaps between strong and weak models. Unlike SelfRank, which relies on triplet-based pairwise comparisons and is prone to noise, ReasonerRank refines rankings through a structured re-ranking mechanism, and this mechanism makes our method adapt more effectively across different datasets of varying difficulty and domain.

4.4 Ablation Study

4.4.1 Different Prompt Instruction

To assess the impact of prompt instructions on ranking stability, we compare two reasoning step generation strategies: (1) a fixed-step approach, where models generate exactly three reasoning steps (our primary method), and (2) a dynamic-step approach, where step count varies based on question complexity (see Section A.1.3). As shown in Table 3, the fixed-step approach yields a higher RBO score (0.8787), suggesting that enforcing a uniform reasoning structure stabilizes rankings by reducing variation in model responses. In contrast, the dynamic-step approach achieves a perfect SP@3 score (1.0000) but introduces more variability, as models adapt reasoning length to task complexity, slightly altering ranking order.

These results highlight a trade-off: fixed-step reasoning enhances ranking consistency, while dynamic reasoning better aligns with natural model behaviors but increases variability.

Table 2: Abbreviated Results of Ranking 14 LLMs on MMLU, BBH Datasets, 10 LLMs on GSM8K Dataset, Averaged Across All Included Subsets.

Dataset	Method	RBO	SP@3
MMLU	MCA	0.7974	0.6082
	SelfRank-FTR	0.7817	0.5263
	SelfRank-GTR	0.7663	0.4737
	ReasonerRank (ours)	**0.8311**	**0.7018**
BBH	MCA	0.7208	0.5238
	SelfRank-FTR	0.5307	0.2857
	SelfRank-GTR	0.5847	0.4762
	ReasonerRank (ours)	**0.7672**	**0.5714**
GSM8K	MCA	0.5147	0.3333
	SelfRank-FTR	0.6078	0.3333
	SelfRank-GTR	0.5578	0.3333
	ReasonerRank (ours)	**0.8314**	**1.0000**

Table 3: Ablation study of using two different reasoning generation prompts on 5 subsets of MMLU dataset.

Prompt Type	RBO	SP@3
3-Steps Reasoning	**0.8787**	0.9370
Dynamic-Steps Reasoning	0.8538	**1.0000**

4.4.2 Impact of Re-Ranking

We evaluate the effect of TopK re-ranking by comparing rankings with and without it across different candidate LLM set sizes. As shown in Table 4, re-ranking consistently improves both RBO and SP@3 scores. Improvements are most pronounced in SP@3, which means that re-ranking significantly stabilizes the top-ranked models. While the initial ranking separates stronger and weaker models, re-ranking ensures precise alignment with the reference by filtering minor inconsistencies.

Unlike rigid ranking mechanisms, our approach treats the initial ranking as a flexible filtering mechanism rather than an absolute performance measure. This adaptability allows it to remain robust across

varying candidate sets and task difficulties, refining model distributions instead of relying on fixed performance thresholds.

Table 4: Ablation study of ReasonerRank w/wo Top K rerank.

Rank Setting	Re-Rank	RBO	SP@3
10 Models	No	0.4734	0.3041
	Yes	**0.8053**	**0.8538**
11 Models	No	0.4755	0.2982
	Yes	**0.7850**	**0.6667**
12 Models	No	0.4875	0.3509
	Yes	**0.7730**	**0.6082**
13 Models	No	0.4686	0.2515
	Yes	**0.8302**	**0.7310**
14 Models	No	0.4652	0.2515
	Yes	**0.8311**	**0.7018**
15 Models	No	0.4559	0.2398
	Yes	**0.8118**	**0.5673**
16 Models	No	0.4548	0.2398
	Yes	**0.8178**	**0.5673**

4.4.3 Hyperparameter K for TopK-ReRank

In our ranking algorithm, the parameter K determines how many of the estimated "top reasoners" are used to establish the consensus answer for producing the final ranking of LLMs. Although we fix the value of K in our main experiments, we conduct an ablation study to assess the sensitivity of TopK-ReRank mechanism of ReasonerRank to different choices of K by benchmarking its performance across varying values (see Table 5 for results, and see detailed discussion in Section A.1.7).

4.4.4 Effect on Different LLMs Set Size

We assess ranking stability by varying the number of candidate models from 10 to 16. The results for MMLU and BBH are shown in Tables 8 and 9, with trends visualized in Figures 4 and 5.

RBO remains stable across candidate set sizes, while SP@3 declines as the number of ranked models increases, particularly in BBH. This suggests that maintaining a stable top-ranked subset becomes more challenging with a larger candidate pool, aligning with our earlier observations in Section 4.3 that complex reasoning tasks introduce greater variance in model performance.

Despite these variations, ReasonerRank consistently achieves the highest RBO and SP@3, demonstrating that the initial ranking step provides a flexible foundation adaptable to different performance distributions. Instead of rigid ranking rules, our method enables **dynamic ranking refinement**, enhancing robustness across candidate sets.

Table 5: Ablation study results for $K \in [1, 15]$ on the MMLU dataset, where the **bold** one is the best result, and the **blue** one is our reported result (see discussion in Section A.1.7 for details).

ReasonerRank's K	RBO	SP@3
K = 1	0.7831	0.4386
K = 2	0.7906	0.5322
K = 3	0.7378	0.2573
K = 4	0.7109	0.1813
K = 5	0.7458	0.3626
K = 6	**0.8178**	**0.5673**
K = 7	**0.8230**	**0.6433**
K = 8	0.8067	0.5906
K = 9	0.7918	0.5439
K = 10	0.7635	0.4971
K = 11	0.7739	0.4854
K = 12	0.7878	0.5556
K = 13	0.7979	0.5673
K = 14	0.8048	0.5789
K = 15	0.7949	0.5614

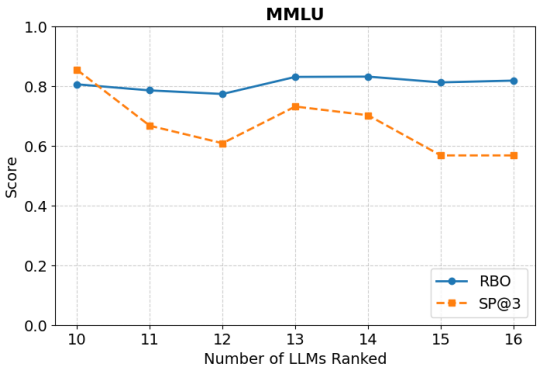

Figure 4: Performance curve of ReasonerRank as the number of LLMs to rank increases from 10 to 16 on the MMLU dataset.

5 Related Works

Evaluating LLM reasoning remains a key challenge. Prior works explore structured evaluation frameworks, ranking-based methods, and consensus-driven approaches. Several studies propose frameworks for assessing LLM reasoning, focusing on detecting inconsistencies (Liu et al., 2024), step-by-

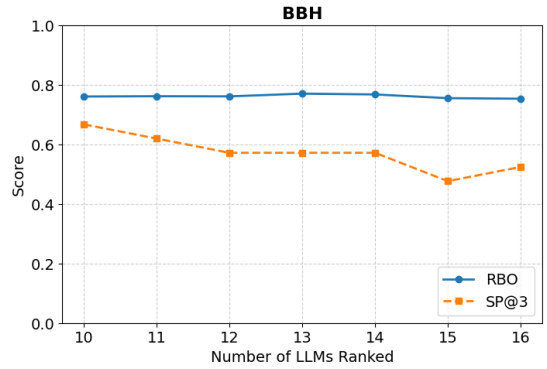

Figure 5: Performance curve of ReasonerRank as the number of LLMs to rank increases from 10 to 16 on the BBH dataset.

step reasoning evaluation (Hao et al., 2024), and correctness assessment (Prasad et al., 2023). Multi-agent debate has also been explored as a way to improve factuality and reasoning (Du et al., 2023). Challenges in LLM-based NLG evaluation have been extensively discussed (Gao et al., 2024; Xu et al., 2024b; Li et al., 2024a). What's more, the efficiency of LLM reasoning is becoming an important topic recently (Sui et al., 2025; Zhang et al., 2025b; Liu et al., 2025; Zhang et al., 2025a).

Ranking-based approaches assess models without ground truth labels, using triplet ranking (Dhurandhar et al., 2024), reputation-based consensus (Ning et al., 2025), peer discussion ranking (Li et al., 2023), and multi-agent debate frameworks (Chan et al., 2024). Reference-free scoring mechanisms have also been proposed (Zheng et al., 2024a; Kenton et al., 2025), along with studies analyzing NLG evaluation metrics (Xiao et al., 2023). Consensus-based methods aggregate multiple model outputs for evaluation, addressing biases in ranking and improving stability (Li et al., 2024b; Wan et al., 2024; Xu et al., 2024a). Unlike prior work, our method integrates instruction adherence, reasoning coherence, and reasoning-evidence consistency into a unified ranking framework, ensuring stability across tasks and model distributions.

6 Conclusions

Our paper highlights the challenges of ranking large language models in increasingly complex tasks. As dataset difficulty increases, traditional triplet-based ranking methods like SelfRank become less reliable, particularly when the performance gap between models widens. In contrast, our proposed ReasonerRank method remains stable under these conditions.

Beyond ranking accuracy, our findings emphasize the need for alternative evaluation criteria when assessing foundational language models, especially in scenarios where ground truth is unavailable or incomplete. Even in cases where ground truth exists, these aspects may still be extremely critical in evaluating a model's ability to generalize and maintain consistency across diverse tasks.

Future work can explore integrating such qualitative evaluation factors into ranking frameworks, moving toward a more holistic assessment of language models. By expanding the scope of evaluation beyond purely performance-driven measures, we can develop more comprehensive frameworks for assessing the effectiveness of large language models in real-world applications.

7 Limitations

While our study provides valuable insights into ranking large language models, several limitations warrant further exploration. While our method relies on structured output to measure instruction-following ability as a proxy for general LLM capability, we do not explore cases where a model exhibits strong reasoning ability but fails to adhere to the prescribed output format. Additionally, our approach relies on text embeddings as semantic representations to approximate LLMs' reasoning quality, so more effective metrics or evaluation frameworks could be explored in future work to provide a more accurate assessment. Despite these limitations, our work lays a foundation for advancing LLM ranking research and encourages further developments in evaluation methodologies.

Acknowledgments

This research was partially supported by NSF Awards ITE-2429680, IIS-2310260. The views and conclusions in this paper are those of the authors and do not represent the views of any funding or supporting agencies.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Boxi Cao, Mengjie Ren, Hongyu Lin, Xianpei Han, Feng Zhang, Junfeng Zhan, and Le Sun. 2024. Structeval: Deepen and broaden large language model assessment via structured evaluation. *Preprint*, arXiv:2408.03281.

Chi-Min Chan, Weize Chen, Yusheng Su, Jianxuan Yu, Wei Xue, Shanghang Zhang, Jie Fu, and Zhiyuan Liu. 2024. Chateval: Towards better LLM-based evaluators through multi-agent debate. In *The Twelfth International Conference on Learning Representations*.

Yupeng Chang, Xu Wang, Jindong Wang, Yuan Wu, Linyi Yang, Kaijie Zhu, Hao Chen, Xiaoyuan Yi, Cunxiang Wang, Yidong Wang, et al. 2024. A survey on evaluation of large language models. *ACM Transactions on Intelligent Systems and Technology*, 15(3):1–45.

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

Amit Dhurandhar, Rahul Nair, Moninder Singh, Elizabeth Daly, and Karthikeyan Natesan Ramamurthy. 2024. Ranking large language models without ground truth. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 2431–2452, Bangkok, Thailand. Association for Computational Linguistics.

Yilun Du, Shuang Li, Antonio Torralba, Joshua B. Tenenbaum, and Igor Mordatch. 2023. Improving factuality and reasoning in language models through multiagent debate. *Preprint*, arXiv:2305.14325.

Jonathan St BT Evans and Keith E Stanovich. 2013. Dual-process theories of higher cognition: Advancing the debate. *Perspectives on psychological science*, 8(3):223–241.

Mingqi Gao, Xinyu Hu, Jie Ruan, Xiao Pu, and Xiaojun Wan. 2024. Llm-based nlg evaluation: Current status and challenges. *arXiv preprint arXiv:2402.01383*.

Saibo Geng, Hudson Cooper, Michał Moskal, Samuel Jenkins, Julian Berman, Nathan Ranchin, Robert West, Eric Horvitz, and Harsha Nori. 2025. Jsonschemabench: A rigorous benchmark of structured outputs for language models. *Preprint*, arXiv:2501.10868.

Shibo Hao, Yi Gu, Haotian Luo, Tianyang Liu, Xiyan Shao, Xinyuan Wang, Shuhua Xie, Haodi Ma, Adithya Samavedhi, Qiyue Gao, et al. 2024. Llm reasoners: New evaluation, library, and analysis of step-by-step reasoning with large language models. *arXiv preprint arXiv:2404.05221*.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding. *Preprint*, arXiv:2009.03300.

Zachary Kenton, Noah Siegel, János Kramár, Jonah Brown-Cohen, Samuel Albanie, Jannis Bulian, Rishabh Agarwal, David Lindner, Yunhao Tang, Noah Goodman, et al. 2025. On scalable oversight with weak llms judging strong llms. *Advances in Neural Information Processing Systems*, 37:75229–75276.

Ruosen Li, Teerth Patel, and Xinya Du. 2023. Prd: Peer rank and discussion improve large language model based evaluations. *arXiv preprint arXiv:2307.02762*.

Zhen Li, Xiaohan Xu, Tao Shen, Can Xu, Jia-Chen Gu, Yuxuan Lai, Chongyang Tao, and Shuai Ma. 2024a. Leveraging large language models for NLG evaluation: Advances and challenges. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 16028–16045, Miami, Florida, USA. Association for Computational Linguistics.

Zongjie Li, Chaozheng Wang, Pingchuan Ma, Daoyuan Wu, Shuai Wang, Cuiyun Gao, and Yang Liu. 2024b. Split and merge: Aligning position biases in llm-based evaluators. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 11084–11108.

Ruikang Liu, Yuxuan Sun, Manyi Zhang, Haoli Bai, Xianzhi Yu, Tiezheng Yu, Chun Yuan, and Lu Hou. 2025. Quantization hurts reasoning? an empirical study on quantized reasoning models. *Preprint*, arXiv:2504.04823.

Ziyi Liu, Soumya Sanyal, Isabelle Lee, Yongkang Du, Rahul Gupta, Yang Liu, and Jieyu Zhao. 2024. Self-contradictory reasoning evaluation and detection. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 3725–3742, Miami, Florida, USA. Association for Computational Linguistics.

Renze Lou, Kai Zhang, and Wenpeng Yin. 2024. Large language model instruction following: A survey of progresses and challenges. *Computational Linguistics*, 50(3):1053–1095.

Hugo Mercier and Dan Sperber. 2011. Why do humans reason? arguments for an argumentative theory. *Behavioral and brain sciences*, 34(2):57–74.

Ibomoiye Domor Mienye and Yanxia Sun. 2022. A survey of ensemble learning: Concepts, algorithms, applications, and prospects. *IEEE Access*, 10:99129–99149.

Kun-Peng Ning, Shuo Yang, Yuyang Liu, Jia-Yu Yao, Zhenhui Liu, Yonghong Tian, Yibing Song, and Li Yuan. 2025. PiCO: Peer review in LLMs based on consistency optimization. In *The Thirteenth International Conference on Learning Representations*.

Archiki Prasad, Swarnadeep Saha, Xiang Zhou, and Mohit Bansal. 2023. ReCEval: Evaluating reasoning chains via correctness and informativeness. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 10066–10086, Singapore. Association for Computational Linguistics.

Connor Shorten, Charles Pierse, Thomas Benjamin Smith, Erika Cardenas, Akanksha Sharma, John Trengrove, and Bob van Luijt. 2024. Structuredrag: Json response formatting with large language models. *arXiv preprint arXiv:2408.11061*.

Aarohi Srivastava, Abhinav Rastogi, Abhishek Rao, et al. 2023. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models. *Preprint*, arXiv:2206.04615.

Yang Sui, Yu-Neng Chuang, Guanchu Wang, Jiamu Zhang, Tianyi Zhang, Jiayi Yuan, Hongyi Liu, Andrew Wen, Shaochen Zhong, Hanjie Chen, and Xia Hu. 2025. Stop overthinking: A survey on efficient reasoning for large language models. *Preprint*, arXiv:2503.16419.

Yuxuan Wan, Wenxuan Wang, Yiliu Yang, Youliang Yuan, Jen-tse Huang, Pinjia He, Wenxiang Jiao, and Michael Lyu. 2024. LogicAsker: Evaluating and improving the logical reasoning ability of large language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 2124–2155, Miami, Florida, USA. Association for Computational Linguistics.

Yicheng Wang, Jiayi Yuan, Yu-Neng Chuang, Zhuoer Wang, Yingchi Liu, Mark Cusick, Param Kulkarni, Zhengping Ji, Yasser Ibrahim, and Xia Hu. 2024. Dhp benchmark: Are llms good nlg evaluators? *arXiv preprint arXiv:2408.13704*.

William Webber, Alistair Moffat, and Justin Zobel. 2010. A similarity measure for indefinite rankings. *ACM Transactions on Information Systems (TOIS)*, 28(4):1–38.

Ziang Xiao, Susu Zhang, Vivian Lai, and Q. Vera Liao. 2023. Evaluating evaluation metrics: A framework for analyzing NLG evaluation metrics using measurement theory. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 10967–10982, Singapore. Association for Computational Linguistics.

Fangzhi Xu, Qika Lin, Jiawei Han, Tianzhe Zhao, Jun Liu, and Erik Cambria. 2024a. Are large language models really good logical reasoners? a comprehensive evaluation and beyond. *Preprint*, arXiv:2306.09841.

Shuying Xu, Junjie Hu, and Ming Jiang. 2024b. Large language models are active critics in nlg evaluation. *arXiv preprint arXiv:2410.10724*.

Wangsong Yin, Mengwei Xu, Yuanchun Li, and Xuanzhe Liu. 2024. Llm as a system service on mobile devices. *arXiv preprint arXiv:2403.11805*.

Jiayi Yuan, Ruixiang Tang, Xiaoqian Jiang, and Xia Hu. 2024. Large language models for healthcare data augmentation: An example on patient-trial matching. In *AMIA Annual Symposium Proceedings*, volume 2023, page 1324.

Jiayi Yuan, Jiamu Zhang, Andrew Wen, and Xia Hu. 2025. The science of evaluating foundation models. *arXiv preprint arXiv:2502.09670*.

Nan Zhang, Yusen Zhang, Prasenjit Mitra, and Rui Zhang. 2025a. When reasoning meets compression: Benchmarking compressed large reasoning models on complex reasoning tasks. *Preprint*, arXiv:2504.02010.

Wenyuan Zhang, Shuaiyi Nie, Xinghua Zhang, Zefeng Zhang, and Tingwen Liu. 2025b. S1-bench: A simple benchmark for evaluating system 1 thinking capability of large reasoning models. *Preprint*, arXiv:2504.10368.

Xin Zhang, Yanzhao Zhang, Dingkun Long, Wen Xie, Ziqi Dai, Jialong Tang, Huan Lin, Baosong Yang, Pengjun Xie, Fei Huang, Meishan Zhang, Wenjie Li, and Min Zhang. 2024. mGTE: Generalized long-context text representation and reranking models for multilingual text retrieval. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pages 1393–1412, Miami, Florida, US. Association for Computational Linguistics.

Danna Zheng, Danyang Liu, Mirella Lapata, and Jeff Z Pan. 2024a. Trustscore: Reference-free evaluation of llm response trustworthiness. *arXiv preprint arXiv:2402.12545*.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. 2024b. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in Neural Information Processing Systems*, 36.

A Appendix

A.1 Experiment Detail

A.1.1 Dataset Selection

We selected three benchmark datasets: MMLU, GSM8K, and a subset of BBH. These datasets were chosen to comprehensively evaluate models on both broad knowledge and complex reasoning tasks.

MMLU consists of a diverse collection of subjects spanning multiple disciplines, covering both general knowledge and specialized domains. We used all subjects from MMLU to ensure a comprehensive assessment of a model's general understanding and factual recall capabilities.

GSM8K consists of high-quality math problems specifically designed to require multi-step reasoning, and this makes it well-suited for our evaluation game.

For BBH, we curated a subset of tasks that focus on logical reasoning and error detection. Specifically, we include the following categories:

- **Boolean Expressions**: Evaluates the model's ability to understand and manipulate boolean logic.
- **Formal Fallacies**: Assesses the model's capacity to recognize logical errors in reasoning.
- **Logical Deduction** (Three, Five, and Seven Objects): Tests the model's ability to infer logical conclusions from given premises under different levels of complexity.
- **Temporal Sequences**: Measures the model's understanding of ordered events and their relationships.
- **Salient Translation Error Detection**: Examines the model's ability to identify errors in translations that significantly alter meaning.

This selection ensures that our evaluation framework balances general knowledge comprehension and reasoning-intensive tasks, and this helps provide a robust comparison of model performance across different cognitive and reasoning challenges.

A.1.2 Model Access and Selection

To conduct a comprehensive evaluation, we access model inference through APIs provided by TogetherAI and OpenAI. These APIs enabled us to interact with a wide range of models, covering both open-source and proprietary variants. Using API-based access ensured consistency across models, as we could evaluate them under a unified framework without requiring individual deployments.

Our model selection is designed to capture a broad spectrum of model characteristics, ensuring a diverse set of candidate models for evaluation. One key factor was size coverage, particularly for open-source models, ranging from smaller-scale models such as Gemma-2B-It to larger models such as Qwen2.5-72B-Instruct-Turbo. In addition to size, we consider architectural and implementation diversity, selecting models from different research labs and organizations, including Meta's LLaMA, Google's Gemma, Mistral's Mixtral, and Alibaba's Qwen. Beyond individual architectures, we included both single-expert models and mixture-of-experts (MoE) architectures to explore their respective performance characteristics. Finally, we considered model accessibility, which make sure that our selection encompasses both open-source and closed-source models. Open models, such as LLaMA, Gemma, and Qwen, were evaluated alongside proprietary models like GPT-3.5-Turbo and GPT-4o-Mini.

By selecting models based on these factors–size, architecture, implementation diversity, and accessibility–we creat a diverse candidate set that allows us to rigorously test the effectiveness of different label-free ranking methods in correctly ranking models across a wide range of capabilities. This selection ensures that our evaluation method generalizes well as it can capture meaningful distinctions in model reasoning performance.

A.1.3 Prompt Templates for Reasoning Generation

We use two different prompts to generate structured reasoning steps in the ablation study: (1) a fixed-step reasoning prompt that limits the reasoning process to three steps, and (2) a dynamic-step reasoning prompt that allows models to generate reasoning steps adaptively based on question complexity.

Fixed 3-Step Reasoning Prompt

```
You are tasked with answering a question and explaining your reasoning in a clear and logical
  format.

Use the following structure to organize your response:

1. Start with the direct answer.
2. Provide a step-by-step reasoning process that leads to your answer,
    and please limit your reasoning in 3 steps.
3. Support your reasoning with evidence, examples, or observations.

Here is the template you should follow:

Answer: [Provide the direct answer.]

Reasoning:
Step 1: [Explain the initial step or assumption.]
Step 2: [Show how this step connects to the next.]
Step 3: [Continue until the reasoning is complete.]

Supporting Evidence:
- [List relevant evidence, facts, or examples.]

Conclusion:
- [One sentence summary of your conclusion.]

Please respond to this question in the specified format: "<question>".

Response:
```

Dynamic-Step Reasoning Prompt

```
You are tasked with answering a question and explaining your reasoning in a clear and logical
  format.

Use the following structure to organize your response:

1. Start with the direct answer.
2. Provide a step-by-step reasoning process that leads to your answer.
3. Support your reasoning with evidence, examples, or observations.

Here is the template you should follow:

Answer: [Provide the direct answer.]

Reasoning:
- [List reasoning steps.]

Supporting Evidence:
- [List relevant evidence, facts, or examples.]

Conclusion:
- [One sentence summary of your conclusion.]

Please respond to this question in the specified format: "<question>".

Response:
```

A.1.4 Design of Structured Response: A Brief Discussion

Requiring a specific structured format not only falls well within the capabilities of modern LLMs but also provides a clear measure of their instruction-following ability, which is an essential factor in ground-truth-free tasks. Empirically, those models that constantly and easily adhere to the required structure and provide reasoning steps have good performance. Only a small minority fail to follow the instructions, leading to poorer performance on the main task.

Although in the real-world scenario, it is possible that a well-performing model may not fully comply with the "Structured Response" format, leading to a potentially suboptimal ranking, our goal here is not to perfectly measure each model's absolute problem-solving ability. Instead, the structured response serves as a practical mechanism to identify a group of strong reasoners whose outputs can be compared and aggregated to define an estimated ground truth. Such a "structured response" design emphasizes collaborative alignment among top-performing models rather than rigidly enforcing formatting as a proxy for LLMs' capability of solving problems.

Moreover, the ability of LLMs to generate structured outputs has emerged as a key indicator of model quality in recent research. Structured output formats, such as JSON and templated reasoning steps (our case), are now widely used across domains, including code generation, data extraction, and tool use. Several benchmarks have been proposed to systematically evaluate LLMs under such constraints (Shorten et al., 2024; Geng et al., 2025; Cao et al., 2024).

A.1.5 Efficiency Analysis

We acknowledge the importance of evaluating the computational efficiency of our framework. While our method may incur slightly higher evaluation costs due to generating more structured responses and computing text embeddings compared to merely generating a single answer, these overheads are negligible compared to the extensive human labor required to manually craft large-scale ground-truth labels.

Though compared with methods like MCA (Mienye and Sun, 2022) and Triplet Ranking (Dhurandhar et al., 2024), our method needs 100 more tokens generated by LLM and 11 overhead of generating embeddings for each question. Empirically, the entire evaluation pipeline will cost 2 to 4 hours (depending on the size of the dataset), which is acceptable and can be considered as an "online evaluation".

In addition, other baseline methods often suffer performance degradation and ranking biases. Such inaccuracies can lead to greater long-term costs and effort for enterprises, as they may need to rerank models or retrain systems to correct these errors. In this context, "slow is fast" - investing in a more rigorous evaluation upfront can save substantial time and resources in the long run by ensuring better initial rankings.

A.1.6 Semantic Similarity

To evaluate reasoning consistency and reasoning-evidence alignment, we compute the semantic similarity between sentences using an embedding-based approach.

Embedding Our approach utilizes a sentence embedding model trained to capture semantic relationships between texts, specifically employing `Alibaba-NLP/gte-large-en-v1.5` (Zhang et al., 2024). While embedding models may vary in performance, our method does not rely on a specific one – stronger models may enhance reasoning consistency evaluation, but even smaller, accessible models can still provide meaningful rankings.

Similarity Computation Given two sentences s_1 and s_2, we first encode them into vector representations:

$$v_1 = \text{Enc}(s_1), \quad v_2 = \text{Enc}(s_2) \tag{7}$$

where $\text{Enc}(\cdot)$ denotes the embedding model. We then compute cosine similarity:

$$\text{sim}(s_1, s_2) = \frac{v_1 \cdot v_2}{\|v_1\|\|v_2\|} \tag{8}$$

where a higher score indicates stronger semantic alignment.

Effect of Embedding Choice While different embedding models may yield slight variations in similarity scores, the overall ranking framework remains robust. Our selected model is neither the most powerful nor the weakest but offers a strong balance between performance and accessibility.

A.1.7 Selection of K in TopK-ReRank

We report results with fixed $K = 4$ in the GSM8K dataset and $K = 6$ in the MMLU and BBH

datasets. These choices balance two objectives: (1) ensuring a stable and meaningful consensus without overfitting to any specific ranking configuration, and (2) maintaining comparability across different model set sizes. Beyond this, we conduct an ablation study on evaluating the performance of ReasonerRank under different K input, and the result shows that **the effect of varying K within the examined range is somewhat limited when comparing with other baseline methods**, and this further underscore the reliability of our method in diverse scenarios. Table 6 below summarizes the ablation results.

Table 6: Ablation study results for $K \in [1, 15]$ (with $K = 16$ corresponding to Majority Vote) on the MMLU dataset, where 16 LLMs are evaluated - the most challenging setting in our experimental setup. The **bold** one is the best result, and the **blue** one is our reported result. The overall closeness to stable performance across different K values demonstrates the robustness of our evaluation method.

TopK-ReRank	RBO	SP@3
K = 1	0.7831	0.4386
K = 2	0.7906	0.5322
K = 3	0.7378	0.2573
K = 4	0.7109	0.1813
K = 5	0.7458	0.3626
K = 6	**0.8178**	**0.5673**
K = 7	**0.8230**	**0.6433**
K = 8	0.8067	0.5906
K = 9	0.7918	0.5439
K = 10	0.7635	0.4971
K = 11	0.7739	0.4854
K = 12	0.7878	0.5556
K = 13	0.7979	0.5673
K = 14	0.8048	0.5789
K = 15	0.7949	0.5614

A.2 Extended Evaluation Metrics

A.2.1 Rank-Biased Overlap (RBO)

Given that S and T (estimated rank and target rank) are two lists of equal size ($|S| = |T|$), and let $d \in [1, |S|]$ denote the depth of both lists, the size of the intersection of lists S and T up to depth d is defined as:

$$X_{S,T,d} = |I_{S,T,d}| = |S_{:d} \cap T_{:d}| \tag{9}$$

The agreement of S and T at depth d is the proportion of overlap, given by:

$$A_{S,T,d} = \frac{|I_{S,T,d}|}{d} \to A_d \tag{10}$$

To control the contribution of each item at depth d to the RBO value, a weight for A_d is introduced:

$$RBO(S, T, w) = \sum_d w_d \cdot A_d \tag{11}$$

Let $w_d = (1 - p) \cdot p^{d-1}$, where $\sum_d w_d = 1$, the rank-biased overlap becomes:

$$RBO(S, T, p) = (1 - p) \sum_d p^{d-1} \cdot A_d \tag{12}$$

In our setting, we treat each depth equally, so the RBO function returns the average overlap across all depths:

$$RBO(S, T) = \frac{1}{|S|} \sum_{d=1}^{|S|} A_d \tag{13}$$

A.2.2 Set Precision at K (SP@K)

Set Precision at K (SP@K) is a ranking evaluation metric that measures the extent of agreement between two ranked lists by focusing specifically on the top-K elements from each list. Unlike traditional ranking metrics that emphasize exact ordering or pairwise positional comparisons, SP@K evaluates the consistency in identifying the most relevant or highest-performing subset of items. In the context of model rankings, SP@K is particularly useful when exact ordering is less critical than accurately selecting the set of best models.

Formally, given two rankings S and T, SP@K for a specific hyperparameter K is calculated as:

$$SP@K(S, T) = \frac{|S_{[1:K]} \cap T_{[1:K]}|}{K}, \tag{14}$$

where $S_{[1:K]}$ and $T_{[1:K]}$ represent the sets of the top-K elements from rankings S and T, respectively. The numerator, $|S_{[1:K]} \cap T_{[1:K]}|$, counts the number of elements shared between these top-K subsets. The division by K normalizes the score so that the value of SP@K ranges from 0 to 1. A score

of **1** indicates perfect agreement, that both rankings agree completely on the top-K elements, irrespective of their exact order. A score of **0** indicates complete disagreement, with no overlap between the top-K elements identified by each ranking.

Such a characteristic of SP@K is especially valuable in scenarios where prioritizing a group of top candidates is more important than precisely determining their ranking order. Additionally, SP@K can be extended by varying the hyperparameter K to explore how agreement changes as we consider larger or smaller subsets of the rankings, and this flexibility allows researchers to identify stable subsets of highly performing models and to analyze the robustness of the ranking methodologies employed.

A.3 Full Experiment Table

A.3.1 Benchmark Performance of LLMs on MMLU

To provide additional context for model evaluation, we benchmark the performance of all candidate LLMs on the MMLU dataset. Each model is evaluated using an identical prompt under a zero-shot setting, ensuring a fair comparison without task-specific fine-tuning.

Table 7: Benchmark performance of LLMs on MMLU (averaged over all subsets). The **bolded** model and values indicate the best-performing model within each family.

Model Family	Model	Accuracy (%)
Gemma	Gemma-2B-IT	39.59
	Gemma-2-27B-IT	**74.97**
Llama	Llama-2-7B-Chat-HF	42.36
	Llama-3.2-3B-Instruct-Turbo	58.60
	Llama-3-8B-Instruct-Lite	61.61
	Llama-3.1-8B-Instruct-Turbo	66.54
	Llama-3.3-70B-Instruct-Turbo	79.93
	Llama-3.1-70B-Instruct-Turbo	**80.31**
Mistral	Mistral-7B-Instruct-v0.3	58.94
	Mistral-7B-Instruct-v0.2	58.97
	Mixtral-8x7B-Instruct-v0.1	67.67
	Mixtral-8x22B-Instruct-v0.1	**73.58**
Qwen	Qwen2.5-7B-Instruct-Turbo	71.22
	Qwen2.5-72B-Instruct-Turbo	**81.62**
GPT	GPT-3.5-Turbo	64.63
	GPT-4o-Mini	**75.59**

A.3.2 Full Ranking Experiments

Table 8: Results for Ranking Models (10 to 16 Candidates) on the MMLU Dataset (Averaged Across All Subjects)

Rank Setting	Method	RBO	SP@3
10 Models	MCA	0.7998	0.8187
	SelfRank-FTR	0.7746	0.6725
	SelfRank-GTR	0.7722	0.5965
	ReasonerRank	**0.8053**	**0.8480**
11 Models	MCA	**0.8165**	**0.7778**
	SelfRank-FTR	0.7907	0.6374
	SelfRank-GTR	0.7890	0.6199
	ReasonerRank	0.7850	0.6667
12 Models	MCA	**0.7965**	**0.6725**
	SelfRank-FTR	0.7848	0.6082
	SelfRank-GTR	0.7646	0.5205
	ReasonerRank	0.7730	0.6082
13 Models	MCA	0.7986	0.6549
	SelfRank-FTR	0.7866	0.5497
	SelfRank-GTR	0.7828	0.5439
	ReasonerRank	**0.8302**	**0.7310**
14 Models	MCA	0.7974	0.6082
	SelfRank-FTR	0.7817	0.5263
	SelfRank-GTR	0.7663	0.4737
	ReasonerRank	**0.8311**	**0.7018**
15 Models	MCA	0.7979	0.5497
	SelfRank-FTR	0.7812	0.4561
	SelfRank-GTR	0.7812	0.4561
	ReasonerRank	**0.8118**	**0.5673**
16 Models	MCA	0.8026	0.5614
	SelfRank-FTR	0.7748	0.4444
	SelfRank-GTR	0.7748	0.4444
	ReasonerRank	**0.8178**	**0.5673**

Table 9: Results for Ranking Models (10 to 16 Candidates) on the BBH Dataset (Averaged Across All Included Subsets)

Rank Setting	Method	RBO	SP@3
10 Models	MCA	0.6909	0.5714
	SelfRank-FTR	0.5208	0.2857
	SelfRank-GTR	0.5768	0.4762
	ReasonerRank	**0.7603**	**0.6667**
11 Models	MCA	0.7112	0.5238
	SelfRank-FTR	0.5036	0.2857
	SelfRank-GTR	0.5946	0.3333
	ReasonerRank	**0.7613**	**0.6190**
12 Models	MCA	0.7202	0.5238
	SelfRank-FTR	0.5015	0.2857
	SelfRank-GTR	0.5829	0.2857
	ReasonerRank	**0.7608**	**0.5714**
13 Models	MCA	0.7390	**0.5714**
	SelfRank-FTR	0.5221	0.2857
	SelfRank-GTR	0.5616	0.3810
	ReasonerRank	**0.7699**	**0.5714**
14 Models	MCA	0.7208	0.5238
	SelfRank-FTR	0.5307	0.2857
	SelfRank-GTR	0.5847	0.4762
	ReasonerRank	**0.7672**	**0.5714**
15 Models	MCA	0.7269	0.4286
	SelfRank-FTR	0.5212	0.2857
	SelfRank-GTR	0.6095	0.2857
	ReasonerRank	**0.7548**	**0.4762**
16 Models	MCA	0.7412	0.4762
	SelfRank-FTR	0.4922	0.0476
	SelfRank-GTR	0.5939	0.3333
	ReasonerRank	**0.7529**	**0.5238**