LFQA-E: Carefully Benchmarking Long-form QA Evaluation

Anonymous ACL submission

Abstract

Long-Form Question Answering (LFQA) refers to generating in-depth, paragraph-level responses to open-ended questions, thus posing a great challenge for evaluation, considering the free format. Previous benchmarks for LFQA evaluation lack references and are constrained by a medium size and limited topics, thus reducing their reliability. To address this gap, we propose a well-constructed, multilingual, and reference-based benchmark named LFQA-E, aiming to rigorously assess the performance of automatic evaluation metrics for LFQA. LFQA-E consists of 1625 questions and 7649 comparisons, covering 15 topics. It is derived from various sources, including online questions and examination questions, designed to test the comprehensive ability of the evaluation metrics. We evaluate 5 types of evaluation metrics, up to 15 specific metrics using LFQA-E. The results reveal that none of the current automatic evaluation metrics show comparable performance with humans, indicating that they cannot capture the dense information contained in long-form responses well. In addition, we provide a detailed analysis of the reasons why automatic evaluation metrics fail when evaluating LFQA and the generalization ability of these metrics.

1 Introduction

Long-form Question Answering (LFQA) (Fan et al., 2019) targets at generating in-depth, paragraph-level responses to open-ended questions. It requires models to have comprehensive domain-specific knowledge or use evidence from retrieved documents (Nakano et al., 2022; Akash et al., 2023) to provide accurate and relevant answers. Despite efforts to enhance the reasonableness and completeness of long-form answers, developing automatic, reliable, and human-aligned evaluation metrics for LFQA is still unexplored.

Evaluating long-form answers is particularly challenging, as they require evaluators to have a comprehensive understanding of the domain. Previous manual evaluations tend to rely on crowd-sourced workers for annotation, but the limited domain expertise inevitably causes low reliability. For automatic evaluation for LFQA, ROUGE (Lin, 2004) is always used. However, Krishna et al. (2021) criticizes ROUGE for its limited informativeness in long-form contexts. Due to the advancement of LLMs (OpenAI, 2023, 2024) and Test-time-scaled Large Reasoning Models (LRMs) (DeepSeek-AI et al., 2025a), many studies leverage them to develop evaluation metrics, through prompting (Wei et al., 2023), fine-tuning, test-time scaling, (Li et al., 2023; Jiang et al., 2024) or training LLMs to be Reward Models (RMs) (Liu et al., 2024a). Though many evaluation metrics exist for LFQA, which one is the most effective and human-aligned needs to be verified and benchmarked.

Previous benchmark for LFQA evaluation (Xu et al., 2023) samples records from reddit/ELI5, hiring experts to annotate the better one between two responses without references, and test alignment between automatic evaluation metrics and expert labels. However, their benchmark has several limitations: **1) Lack of authorized references** A reference answer provides a baseline for assessing whether a response covers key details and maintains factual accuracy. Without ground-truth references, the comparison between metrics may be unfair, and evaluations without clear criteria or rubrics are inherently unreliable. **2) Limited diversity** The benchmark consists of only 260 examples, all in English, constraining its linguistic and topical diversity. Moreover, it treats the comparison as an A/B task, but in real scenarios, a "tie" option is needed for more difficult selection.

To fill the gap, we introduce LFQA-E, towards evaluating the ability of different metrics. Especially, 1) To evaluate whether current automatic evaluation metrics can select a better one from two nuanced responses, we gather references that

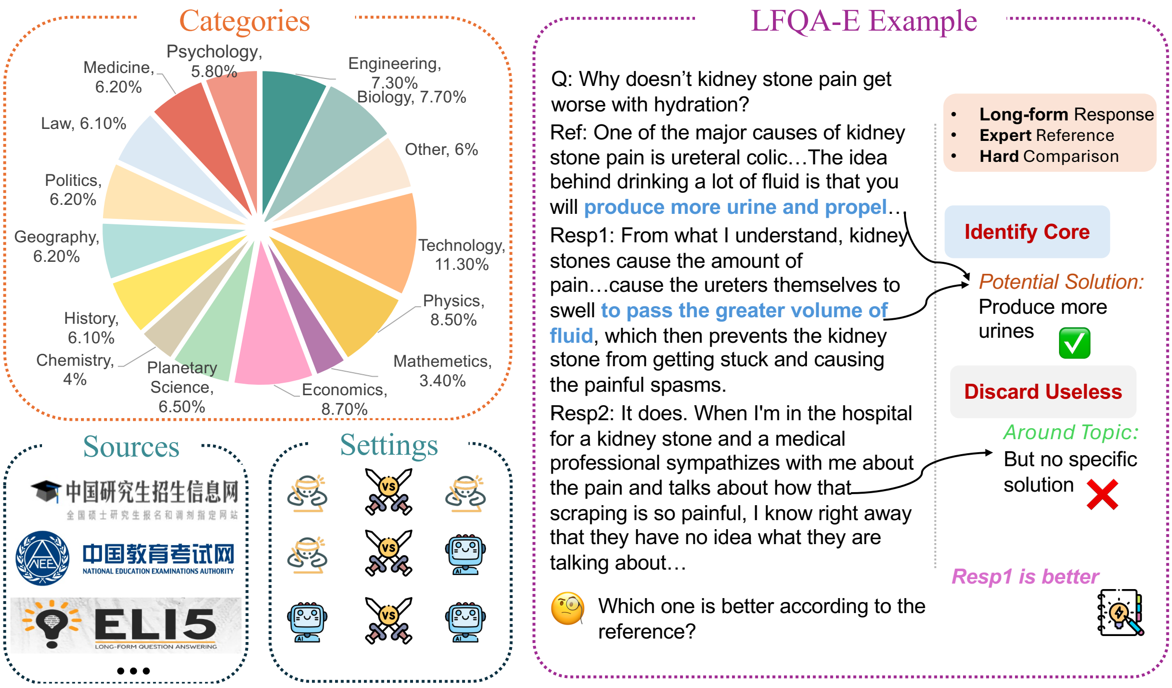

Figure 1: The figure shows the overview of LFQA-E. The left side displays the categories, sources, and three settings, showcasing its diversity. The right side illustrates an example of LFQA-E.

are examined by the experts, and judge based on them. 2) To analyze the bias among evaluation metrics, we rigorously assess the performance based on three settings, i.e, human vs human *h v. h*, human vs model *h v. m*, and model vs model *m v. m*. To ensure the difficulty, we choose human responses based on their upvotes or their scores, and model responses based on two models with comparable capabilities. Moreover, we collect multilingual responses, i.e, English and Chinese, and multiple domain-specific responses, e.g., Engineering, Law, Medicine. 3) To prevent data contamination, we collect data from offline examination, i.e., College Entrance Examination Simulation Questions (CEESQ) and Postgraduate Entrance Examination Questions (PEEQ) and online platform questions from the recent half-year to ensure the model doesn't see the data before. The overview of LFQA-E Benchmark is shown in Figure 1.

Using LFQA-E, we critically assess the efficacy of 15 evaluation metrics, including LLMs, LRMs, and RMs. The experimental results show that current leading evaluation metrics fail to capture core information as well as human beings from verbose responses, especially when differentiating the better one between two model-generated responses. Furthermore, we provide analysis on why automatic evaluation metrics fail in LFQA evaluation.

In conclusion, our contributions are as follows:

- We introduce LFQA-E BENCH, a challenging LFQA evaluation benchmark. It consists of multilingual, clear questions from various domains, authorized references, and nuanced responses to select.
- We test 15 evaluation metrics on LFQA-E BENCH and show the dilemma of current evaluation metrics on LFQA evaluation, even SOTA LRMs and LLMs.
- We provide further analysis on why these metrics fail when evaluating long-form responses. Also, we analyze the generalization of different metrics across topics, settings.

2 Related Work

Development of LFQA LFQA (Fan et al., 2019) requires models to generate paragraph-level responses to open-ended questions which is more complex compared to datasets like SQuAD (Rajpurkar et al., 2016), TriviaQA (Joshi et al., 2017), and NarrativeQA (Kočiský et al., 2017), where answers are primarily words or phrases extracted directly from documents. In LFQA, models must generate nuanced responses based on their knowledge or existing evidence documents. Several studies have analyzed the discourse structure of long-form answers (Xu et al., 2022) and have sought to enhance the performance on LFQA. (Chen et al., 2023; Akash et al., 2023).

Evaluation of LFQA The automatic evaluation of LFQA remains challenging and underexplored. Initially, ROUGE (Lin, 2004) was used as an automatic evaluation metric to calculate the similarity between a candidate and a reference. Later, Krishna et al. (2021) pointed out that ROUGE is not an adequately informative metric for LFQA evaluation. For human annotation, HURDLES (Krishna et al., 2021) and WEBGPT (Nakano et al., 2022) employed A/B testing, where crowdsourced annotators were instructed to choose the better of two candidate answers. Since annotation of LFQA requires high expertise, the results of crowdsourced workers may be unreliable. To address the gap, Xu et al. (2023) employed experts for annotation, and tested several evaluation metrics, such as ROUGE (Lin, 2004), BERTScore (Zhang et al., 2020), and BARTScore (Yuan et al., 2021), on an expert-annotated dataset. Their findings validated that no existing metrics fully align with human judgment. However, the dataset they used lacks expert-written references, sourced from Reddit/ELI5, and is limited in scale, comprising only about 260 samples. More recently, since the development of LLMs and LRMs, many work uses them for evaluation of free-form answers, using prompt (Wei et al., 2023), fine-tuning using specific data (Liu et al., 2023), and reinforcement-learning (Chen et al., 2025).

	LFQA-E-EN	LFQA-E-ZH
# Topics	9	6
# Questions	1026	599
# Comparisons	6156	1493
# Avg Que. Lens	13.4	24.6
# Avg Ref. Lens	299.1	187.2
# Avg Res. Lens	245.0	308.3
Annotate	Expert	Expert

Table 1: Detailed statistics of LFQA-E. **Avg Que. Lens**, **Avg Ref. Lens**, **Avg Res. Lens** corresponds to question lengths, reference lengths, and response lengths, respectively.

3 Methodology

3.1 Overview

To reasonably test the evaluation ability of different metrics for LFQA when having a reference, we introduce LFQA-E, a multilingual and comprehensive benchmark composed of different topics and questions. LFQA-E BENCH consists of the Chinese version LFQA-E-ZH and the English version LFQA-E-EN. Table 1 shows its overview. It includes 1625 questions and 7649 comparisons, consisting of 1493 comparisons in Chinese and 6139 comparisons in English. It spans 15 topics, ranging from history to engineering, ensuring its diversity. LFQA-E comprises expert-annotated references for fair comparison and nuanced responses. Therefore, it is naturally a hard yet reasonable benchmark for LFQA evaluation.

Reference-Based Evaluation For LFQA-E BENCH, references are sourced from academic examinations or widely discussed questions in Reddit/ELI5. After being reviewed by experts with relevant academic backgrounds, these references are ensured to cover all the key points needed to answer the question. This provides a baseline for evaluation metrics to look up and provide a more precise comparison.

Difficult Comparisons All the questions contained in LFQA-E have been carefully examined by domain experts to ensure it is answerable and clear to understand. We ensure that models have not seen the data by collecting data from recent examinations and forum questions. The responses are collected from human-written responses, with close scores or upvotes, and model responses generated by comparable LLMs. Therefore, it is hard to simply distinguish the better one.

Diverse Benchmark We meticulously collect 1493 questions and 7649 comparisons in 15 distinct domains, from natural science to social science, to guarantee a diverse and representative benchmark. Also, LFQA-E is multilingual, consisting of examples in both Chinese and English. Moreover, LFQA-E includes three kinds of comparisons, guaranteeing the comprehensibility of the benchmark.

3.2 Data Processing

The data processing pipeline can be divided into three phases: Data Collection, Human Response Collection, and Model Response Generation.

Data Collection For LFQA-E-ZH, we source our data from CEESQ and PEEQ, where questions and references are written by domain experts, e.g., teachers and professors from high school and colleges. For LFQA-E-EN, data is sourced from the Reddit/ELI5, where each question is explained without the use of specialized terminology or complex concepts, and we use the top-ranked answer as our reference. To prevent overlapping with potential training data, we avoid using data from the actual College Entrance Examination, and the questions we captured from ELI5 are all from the past 6 months. To ensure that all the questions are clear and answerable, we instruct GPT-4o to filter out questions whose description is unclear and whose answer is too broad to give a reference. After that, to ensure our references contain all the information needed to answer the questions, we first use GPT-4o as our checker. We prompt GPT-4o multiple times, and if at any one time, it identifies the reference as invalid, we will discard the example. Then we pass the remaining questions and references for expert annotation. After that, we get 1625 questions. The instructions we used are listed in Appendix C.1.

Human Response Collection For LFQA-E-ZH, we gather examination papers primarily in image format and employ Optical Character Recognition (OCR) systems to meticulously extract student responses. Specifically, we choose student answers with close scores to ensure the comparison difficulty. The OCR is conducted using the Volcano Engine API. For LFQA-E-EN, we collect responses from the forum section of the corresponding question. Also, we select answers within the many-voted yet close up-votes to make them hard to differentiate. However, the responses we collect for LFQA-E-ZH are mainly written during examination, it is concise and well structured, and the responses we collect for LFQA-E-EN include some special characters like URLs. These special patterns deteriorate our data quality. To handle it, we use GPT-4o to paraphrase and clean our human responses. The instruction we used is shown in Appendix C.1.

Model Response Generation When generating model responses, we focus on evaluating whether LLMs can understand the semantic meaning of texts well and properly select the better response. Therefore, we do not impose strict requirements on answer quality. Instead, we ensure the difficulty of LFQA-E BENCH by selecting models with similar ranking in the LMSYS Arena (Chiang et al., 2024; Zheng et al., 2023, 2024). Specifically, we leverage Llama-3-8B-Instruct (Dubey et al., 2024) and GPT-3.5-turbo (OpenAI, 2023) for response generation. For model-generated answers, we use "Generate reasonable answers to the following questions. Use references or examples if needed" to prompt LLMs. The generation temperature is set to 1.0 to encourage diverse and creative responses.

3.3 Human Annotation

The Human Annotation Process can be separated into the following steps: Annotator Decision, Annotation Setting, and Annotation Process.

Annotator Decision LFQA evaluation suffers from distinct challenges. Firstly, paragraph-level responses can overwhelm annotators, leading to a loss of focus. Secondly, annotators must have deep domain knowledge to accurately judge responses against references. Lastly, the syntactic and semantic complexities of long-form responses often intertwine correct and incorrect information within single sentences. To address these issues, we hire annotators from relevant aspects or who have taken relevant courses. Then we provide them with clear and detailed annotation recipes for better quality control. The annotation recipe is in Appendix D.

Annotation Setting Guided by Xu et al. (2023), our evaluation criteria mainly focus on factuality and completeness according to the reference, since almost all responses we collect are already very fluent. Unlike typical A/B testing, our method employs a triple-choice format, giving a tie option, to better capture the subtle differences between answers, as they often show comparable levels of information overlap with the reference. The additional information is useless or verbose according to the central topic.

Annotation Process The annotators assess two responses against a given reference and select the more informative and complete answer or declare a "tie" if both are of similar quality. The process includes identifying key Information, checking for Key Information in Responses, Handling Responses, and comparing overlapping information. During the process, we treat a piece of information as the basic unit. Initially, annotators extract the key information needed to answer the question from the provided reference and check whether the responses under evaluation contain similar statements. Then, they will select a better one based on the overlapped information. To minimize bias and subjectivity, each record is annotated by

Model	LFQA-E-EN		LFQA-E-ZH		Avg_{F1}	Avg_{Acc}
	F1	Accuracy	F1	Accuracy		
Static Evaluation Metric						
Length	26.0	42.8	33.5	52.6	30.8	47.7
ROUGE	37.5	55.5	34.0	49.7	35.8	52.6
BERTScore	35.9	54.1	36.6	52.4	36.3	53.3
LLMs-based Evaluation Metric						
Qwen2.5-32B-Instruct	45.8	63.5	41.8	56.7	43.8	60.1
Qwen2.5-72B-Instruct	43.1	61.2	39.0	53.0	41.1	57.1
Llama3.1-70B-Instruct	42.5	59.6	29.4	30.7	36.0	45.2
GPT-4o	**46.4**	61.7	42.6	53.2	**44.5**	57.5
DeepSeek-V3	39.3	57.9	41.1	53.8	40.2	55.9
RM-based Evaluation Metric						
Skywork-Reward-Llama	37.3	54.4	38.2	53.6	37.8	54.0
Skywork-Reward-Gemma	37.5	56.0	33.0	48.3	35.3	52.2
LRM-based Evaluation Metric						
o1-mini	45.9	62.9	**45.2**	**58.9**	45.6	**60.9**
Deepseek-R1	42.9	59.6	42.4	57.8	42.7	58.7
Trained Evaluation Metric						
Auto-J-6B-bilingual	46.0	**66.8**	35.4	51.9	40.7	59.4
Prometheus-7B-v2.0	41.8	64.2	34.1	50.1	38.0	57.2
M-Prometheus-14B	41.6	60.8	33.9	49.4	37.8	55.1

Table 2: Performance of different evaluation metrics on LFQA-E. The largest value is denoted in **bold**.

two independent reviewers. Each comparison takes around 7 minutes to annotate. For some hard-to-differentiate comparisons, detailed justification is saved to help understand. After annotation, we find the Fleiss' kappa correlation of inter-annotator agreement is approximately 64.8%, indicating a relatively high agreement rate.

4 Experiments

4.1 Models

We evaluate various evaluation metrics on LFQA-E-EN and LFQA-E-ZH respectively, including LLMs, Large Reasoning Models (LRMs), Reward Models (RMs), etc.

Static Metrics: We use Length-orientation, ROUGE-1 (Lin, 2004) and BERTScore (F1) (Zhang et al., 2020) since they are widely used as the evaluation metric for LFQA.

LLMs: We select Qwen2.5-32B-Instruct (Qwen et al., 2025), Qwen2.5-72B-Instruct, Llama-3.1-70B-Instruct (Dubey et al., 2024), Deepseek-V3 (DeepSeek-AI et al., 2025b), and GPT-4o (OpenAI et al., 2024).

LRMs: Considering the high time complexity and cost, we use o1-mini and Deepseek-R1 (DeepSeek-AI et al., 2025a).

RMs: Since RMs are capable of differentiating between nuanced responses, we test Skywork-Reward-Gemma-2-27B-v0.2 (Liu et al., 2024a) and Skywork-Reward-Llama-3.1-8B-v0.2 considering their leading position on Reward Bench (Lambert et al., 2024). We refer to them as Skywork-Reward-Gemma and Skywork-Reward-Llama.

Evaluation-Specific Models: There are some models trained to serve as evaluation models. Among these models, we select Auto-J-Bilingual (Li et al., 2023) and Prometheus-7B-v2.0 and M-

Prometheus-14B (Kim et al., 2024).

4.2 Implementation Details

We evaluate all the metrics in both LFQA-E-EN and LFQA-E-ZH. We use Jieba cut for ROUGE-zh. For BertScore, we use roberta-large for LFQA-E-ZH evaluation and bert-base-chinese for LFQA-E-EN. We set the temperature at 1.0 for all LLM-based evaluation metrics to encourage diverse responses. The prompts we used are shown in Appendix C.2. For models with specific training templates, we adopt them. We include references for models to look up in all our settings. We use Accuracy and macro-F1 as our indicators. The computations are as follows:

$$\text{Acc} = \frac{1}{N}\sum_{i=1}^{N}\mathbb{I}(\text{pred}_i = \text{label}_i) \tag{1}$$

$$\text{F1}_{\text{macro}} = \frac{1}{||\mathcal{C}||}\sum_{c\in\mathcal{C}}\left(2\cdot\frac{P_c R_c}{P_c + R_c}\right) \tag{2}$$

where $\mathcal{C} = \{\text{A}, \text{B}, \text{same}\}$

Model	LFQA-E-EN	LFQA-E-ZH
Deepseek-V3	1.8	10.2
Qwen2.5-32B-Instruct	7.2	7.5
Qwen2.5-72B-Instruct	2.6	3.6
Llama-3.1-70B-Instruct	5.0	7.7
o1-mini	7.1	14.1
Deepseek-R1	7.4	7.2
GPT-4o	**9.2**	**14.6**

Table 3: Performance of different evaluation metrics on comparisons that human labels as "tie". The largest value is denoted in **bold**.

4.3 Main Results

Table 2 lists our experimental results. The overall low accuracies and F1-scores of all evaluation metrics indicate the challenge LFQA-E poses to current models and methods.

Comparison Between Metrics Though none of the evaluation metrics achieves a high performance on LFQA-E, we observe that scaling model size doesn't definitely yield a better result. For example, Qwen2.5-32B-Instruct beats Qwen2.5-72B-Instruct by 3%. What's more, LRMs show a great performance compared with LLMs, thanks to their long CoT and extended thinking. Another interesting observation is that for trained evaluation metrics, they show comparable results with models of **100x** parameters, indicating their capability when differentiating nuanced responses. RM-based evaluation metrics don't show promising results when generalizing to LFQA evaluation, perhaps because they are trained to give a better one between two responses, renouncing the "tie" option. We will analyze further and give a fairer comparison in Section 5.3.

Comparison Between Subsets Almost all evaluation metrics show different degrees of performance degradation when evaluating on LFQA-E-ZH, indicating that they cannot generalize from one language to another well. Among all the metrics, LRMs show a relatively stable performance, mainly due to their test-time scaling ability, to have more time to reflect and rethink. Trained evaluation metrics suffer a lot when evaluating on LFQA-E-ZH. We attribute this to their smaller parameter sizes, which make them hard to generalize.

Comparison Between Indicators All evaluation metrics struggle to give a tie as good as human beings. Table 3 indicates that among the evaluation metrics we test, the best result is just 9.2% for LFQA-E-EN and 14.6% for LFQA-E-ZH. Observing the responses, we find that they are too conservative to say two responses are of equal quality. This explains why accuracy is always larger than Macro-F1. The low accuracy on tie comparison reflects the difficulty of LFQA-E again.

5 Analysis

5.1 Does Human Response or Model Response matter?

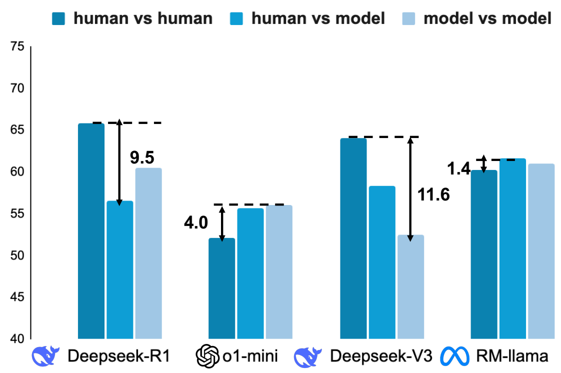

Figure 2: Performance of different models on our three settings on LFQA-E-EN.

To have a thorough understanding of whether the model evaluates human response or model response differently, we experiment on a different group of LFQA-E. We break it into three groups, i.e., *h v. h*,

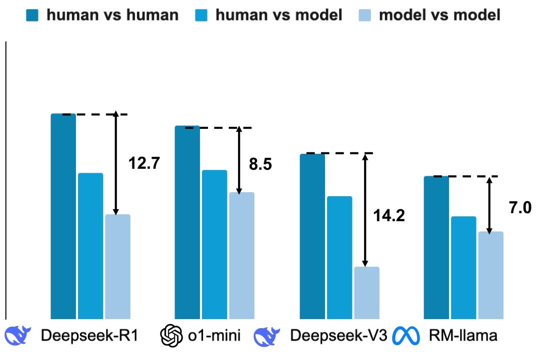

Figure 3: Performance of different models on our three settings on LFQA-E-ZH.

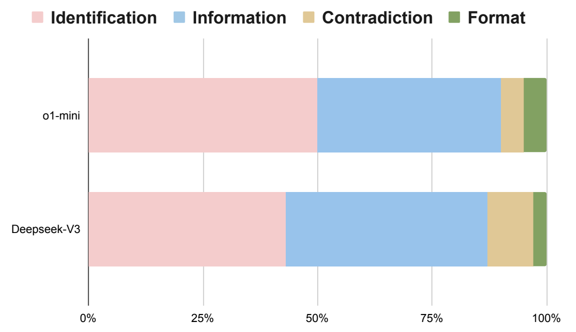

Figure 4: The probability of each error occurring for LMs when evaluating on LFQA-E.

h v. m, and *m v. m*, and see the accuracy changes. The results are listed in Figure 2 for LFQA-E-EN and Figure 3 for LFQA-E-ZH. We can observe that for many evaluation metrics, there exists a huge difference between different comparison settings. In LFQA-E-EN, the RMs show steady ability while others exhibit degradation when model responses are introduced. In LFQA-E-ZH, all the metrics show a drastic accuracy decline under *m v. m*, with a maximum drop of 14.2% from Deepseek-V3. This further validates our assumption, current automatic evaluation metrics cannot handle two responses with similar quality.

5.2 Why Evaluation Metrics fail when Evaluating Long-form Responses?

Since all of the metrics cannot evaluate long-form responses well, we analyze them from the error perspective.

For LM-based Evaluation Metric We observe the outputs of several LLMs and find that almost all errors arise from the following aspects.

- *Keypoints Identification Error*: The model fails to correctly identify and separate bullet keypoints or enumerated lists in responses, leading to poorly structured answers.
- *Irrelevant/Incorrect Information Error*: The model does not penalize or filter out irrelevant or factually incorrect details in its responses, reducing accuracy.
- *Contradiction Error*: During reasoning, the model generates inconsistent or contradictory statements due to factual hallucinations.
- *Formatting Error*: The model produces responses with improper formatting that can't be parsed.

We show the probability of each error occurring in Figure 4. We choose Deepseek-V3 and o1-mini for representation. *Point Identification Error* and *Irrelevant/Incorrect Information Error* happen most time, indicating the relatively low inherent ability for LMs when evaluating long-form answers.

RM-based Evaluation Metrics As reported in Liu et al. (2024b), current RMs struggle to find a better one when giving two nuanced responses. Because the RMs we use give two scalars as responses, we are impossible to figure out why they make such a mistake. We suppose that the error arises from out-of-distribution data in LFQA-E and the challenge of long context.

Static Evaluation Metrics These methods simply leverage word-level or embedding-level similarities, which scratch on the surface when evaluating. As described in Fan et al. (2024), considering evaluating two long responses around a topic, there may be many words overlapping. The overly long responses also dilute semantics, making the originally important key information trivial. Therefore, these metrics fail to consider informativeness, but only focus on similarity.

5.3 How do the Performance Changes in A/B Comparison?

Considering that giving a tie option is difficult for both humans and models, we drop out the records that are labeled as a tie and conduct the experiments again. We show the results in Table 4. After discarding the tied comparison, all the evaluation metrics show nontrivial performance boosts. GPT-4o even gets a 6.4% bonus. This increase matches what we find when comparing indicators. Similar to what we observe above, LRMs remain leading on the fairer comparison, and RMs still struggle to generalize to long-form response evaluation. Static

Model	LFQA-E-en	LFQA-E-zh	Avg
	Static Evaluation Metric		
Length	42.7	56.9	49.8 ↑ 2.1%
ROUGE	57.5	53.8	55.7 ↑ 3.1%
BERTScore	56.0	56.6	56.3 ↑ 3.0%
	LLM-based Evaluation Metric		
Qwen2.5-32B-Instruct	66.8	62.8	64.8 ↑ 4.7%
Qwen2.5-72B-Instruct	63.4	57.6	60.5 ↑ 3.4%
Llama-3.1-70B-Instruct	66.1	34.0	50.1 ↑ 4.9%
GPT-4o	66.3	61.4	63.9 ↑ **6.4%**
Deepseek-V3	60.0	60.0	60.0 ↑ 4.1%
	RM-based Evaluation Metric		
Skywork-Reward-Llama	56.7	58.4	57.6 ↑ 3.6%
Skywork-Reward-Gemma	58.2	52.5	55.4 ↑ 3.2%
	LRM-based Evaluation Metric		
o1-mini	67.3	**64.2**	**65.8** ↑ 4.9%
Deepseek-R1	61.6	63.1	62.4 ↑ 3.7%
	Trained Evaluation Metric		
Auto-J-6B-bilingual	**70.0**	57.3	63.7 ↑ 4.3%
Prometheus-7B-v2.0	66.5	54.1	60.3 ↑ 3.1%
M-Prometheus-14B	63.7	53.3	58.5 ↑ 3.4%

Table 4: Performance of different evaluation metrics on LFQA-E. The examples whose labels are tie are discarded for fairer comparison. The largest value is denoted using **bold**.

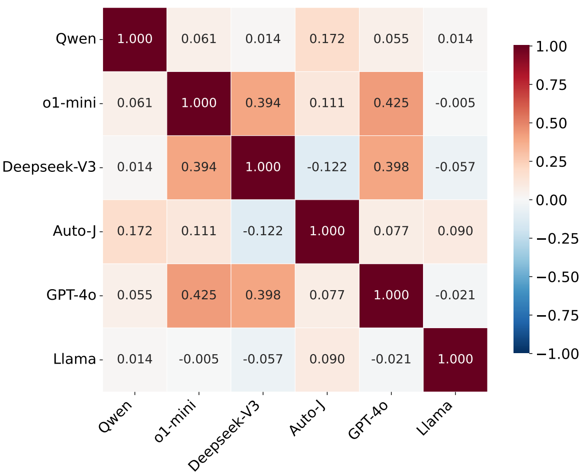

Figure 5: The Cohen's Kappa Correlation Matrix in LFQA-E-EN.

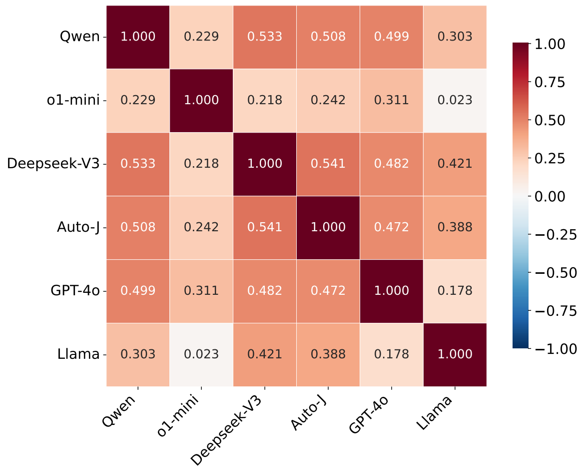

Figure 6: The Cohen's Kappa Correlation Matrix in LFQA-E-ZH.

evaluation metrics, however, show the least improvement. The experimental results demonstrate the potential of test-time scaling, while reflects the generalization problem of RMs. What's more, specific evaluation models show their great potential once again, ranking first on LFQA-E-EN, displaying its future for LFQA evaluation.

5.4 Do Evaluation Metrics Agree with Each Other?

To find whether there is a correlation between different evaluation metrics, we observe the detailed evaluation results. Specifically, we select ROUGE, Qwen2.5-32B-Instruct (simplified as Qwen), GPT-4o, Skywork-Reward-Llama (simplified as Llama), o1-mini, and Auto-J-6B-bilingual (simplified as Auto-J), considering their relatively better performance on LFQA-E. Figure 5 and Figure 6 show the results. We observe that neither of the two metrics achieves a high correlation, indicating two metrics may contradict each other to a large degree. There are even some negative correlations between the two metrics under LFQA-E-EN. This phenomenon further illustrates that there is no stable evaluation metric that aligns well with human preferences.

6 Conclusion

We introduce LFQA-E, a multilingual benchmark for LFQA evaluation. It consists of 1625 questions and 7649 comparisons, spanning 15 topics, from natural science to social science, consisting of 3 settings, i.e., *h v. h*, *h v. m*, and *m v. m*. Each records include a clear question, an authorized reference, and two hard-to-differentiate responses, ensuring its difficulty. We conduct experiments on 15 automatic evaluation metrics. The results show that none of the metrics can evaluate long-form responses as well as human beings. We further analyze the generalization of different metrics across languages and across settings. The results further indicate that all models struggle to generalize well to all comparisons. We find that LRMs and specifically trained evaluation models lead on LFQA-E. The test-time-scaled evaluation model may be used to enhance the performance of LFQA evaluation.

References

Pritom Saha Akash, Kashob Kumar Roy, Lucian Popa, and Kevin Chen-Chuan Chang. 2023. Long-form question answering: An iterative planning-retrieval-generation approach. *Preprint*, arXiv:2311.09383.

Hung-Ting Chen, Fangyuan Xu, Shane Arora, and Eunsol Choi. 2023. Understanding retrieval augmentation for long-form question answering. *Preprint*, arXiv:2310.12150.

Xiusi Chen, Gaotang Li, Ziqi Wang, Bowen Jin, Cheng Qian, Yu Wang, Hongru Wang, Yu Zhang, Denghui Zhang, Tong Zhang, Hanghang Tong, and Heng Ji. 2025. Rm-r1: Reward modeling as reasoning. *Preprint*, arXiv:2505.02387.

Wei-Lin Chiang, Lianmin Zheng, Ying Sheng, Anastasios Nikolas Angelopoulos, Tianle Li, Dacheng Li, Hao Zhang, Banghua Zhu, Michael Jordan, Joseph E. Gonzalez, and Ion Stoica. 2024. Chatbot arena: An open platform for evaluating llms by human preference. *Preprint*, arXiv:2403.04132.

DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, and 181 others. 2025a. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *Preprint*, arXiv:2501.12948.

DeepSeek-AI, Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, Damai Dai, Daya Guo, Dejian Yang, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, and 181 others. 2025b. Deepseek-v3 technical report. *Preprint*, arXiv:2412.19437.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, and 514 others. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Angela Fan, Yacine Jernite, Ethan Perez, David Grangier, Jason Weston, and Michael Auli. 2019. Eli5: Long form question answering. *Preprint*, arXiv:1907.09190.

Yuchen Fan, Xin Zhong, Yazhe Wan, Chengsi Wang, Haonan Cheng, Gaoche Wu, Ning Ding, and Bowen Zhou. 2024. Eva-score: Evaluating abstractive long-form summarization on informativeness through extraction and validation. *Preprint*, arXiv:2407.04969.

Dongfu Jiang, Yishan Li, Ge Zhang, Wenhao Huang, Bill Yuchen Lin, and Wenhu Chen. 2024. Tigerscore: Towards building explainable metric for all text generation tasks. *Preprint*, arXiv:2310.00752.

Mandar Joshi, Eunsol Choi, Daniel S. Weld, and Luke Zettlemoyer. 2017. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. *Preprint*, arXiv:1705.03551.

Seungone Kim, Juyoung Suk, Shayne Longpre, Bill Yuchen Lin, Jamin Shin, Sean Welleck, Graham Neubig, Moontae Lee, Kyungjae Lee, and Minjoon Seo. 2024. Prometheus 2: An open source language model specialized in evaluating other language models. *Preprint*, arXiv:2405.01535.

Tomáš Kočiský, Jonathan Schwarz, Phil Blunsom, Chris Dyer, Karl Moritz Hermann, Gábor Melis, and Edward Grefenstette. 2017. The narrativeqa reading comprehension challenge. *Preprint*, arXiv:1712.07040.

Kalpesh Krishna, Aurko Roy, and Mohit Iyyer. 2021. Hurdles to progress in long-form question answering. *Preprint*, arXiv:2103.06332.

Nathan Lambert, Valentina Pyatkin, Jacob Morrison, LJ Miranda, Bill Yuchen Lin, Khyathi Chandu, Nouha Dziri, Sachin Kumar, Tom Zick, Yejin Choi, Noah A. Smith, and Hannaneh Hajishirzi. 2024. Rewardbench: Evaluating reward models for language modeling. `https://huggingface.co/spaces/allenai/reward-bench`.

Junlong Li, Shichao Sun, Weizhe Yuan, Run-Ze Fan, Hai Zhao, and Pengfei Liu. 2023. Generative judge for evaluating alignment. *Preprint*, arXiv:2310.05470.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Chris Yuhao Liu, Liang Zeng, Jiacai Liu, Rui Yan, Jujie He, Chaojie Wang, Shuicheng Yan, Yang Liu, and Yahui Zhou. 2024a. Skywork-reward: Bag of tricks for reward modeling in llms. *arXiv preprint arXiv:2410.18451*.

Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. 2023. G-eval: Nlg evaluation using gpt-4 with better human alignment. *Preprint*, arXiv:2303.16634.

Yantao Liu, Zijun Yao, Rui Min, Yixin Cao, Lei Hou, and Juanzi Li. 2024b. Rm-bench: Benchmarking reward models of language models with subtlety and style. *Preprint*, arXiv:2410.16184.

Reiichiro Nakano, Jacob Hilton, Suchir Balaji, Jeff Wu, Long Ouyang, Christina Kim, Christopher Hesse, Shantanu Jain, Vineet Kosaraju, William Saunders, Xu Jiang, Karl Cobbe, Tyna Eloundou, Gretchen Krueger, Kevin Button, Matthew Knight, Benjamin Chess, and John Schulman. 2022. Webgpt: Browser-assisted question-answering with human feedback. *Preprint*, arXiv:2112.09332.

OpenAI, :, Aaron Hurst, Adam Lerer, Adam P. Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, Aleksander Mądry, Alex Baker-Whitcomb, Alex Beutel, Alex Borzunov, Alex Carney, Alex Chow, Alex Kirillov, and 401 others. 2024. Gpt-4o system card. *Preprint*, arXiv:2410.21276.

OpenAI. 2023. Chatgpt: Chat generative pre-trained transformer. `https://chat.openai.com/`. Accessed: 2024-08-05.

OpenAI. 2024. Hello gpt-4o. `https://openai.com/index/hello-gpt-4o/`. Accessed: 2024-08-05.

Qwen, :, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, and 25 others. 2025. Qwen2.5 technical report. *Preprint*, arXiv:2412.15115.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text. *Preprint*, arXiv:1606.05250.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. 2023. Chain-of-thought prompting elicits reasoning in large language models. *Preprint*, arXiv:2201.11903.

Fangyuan Xu, Junyi Jessy Li, and Eunsol Choi. 2022. How do we answer complex questions: Discourse structure of long-form answers. *Preprint*, arXiv:2203.11048.

Fangyuan Xu, Yixiao Song, Mohit Iyyer, and Eunsol Choi. 2023. A critical evaluation of evaluations for long-form question answering. *Preprint*, arXiv:2305.18201.

Weizhe Yuan, Graham Neubig, and Pengfei Liu. 2021. Bartscore: Evaluating generated text as text generation. *Preprint*, arXiv:2106.11520.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert. *Preprint*, arXiv:1904.09675.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Tianle Li, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zhuohan Li, Zi Lin, Eric Xing, Joseph E. Gonzalez, Ion Stoica, and Hao Zhang. 2024. Lmsys-chat-1m: A large-scale real-world llm conversation dataset. In *The Twelfth International Conference on Learning Representations*.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

A Limitation

Coverage of Topics and Languages Considering the time and cost, we are unable to obtain more topics and languages. We agree that constructing a benchmark with more topics and languages is significant. We will let this as our future work.

Scalability of LFQA-E LFQA-E cannot be scaled easily, since it requires expert annotation, which costs a great deal of time and money. A more precise evaluation metric should be proposed to ease the tension of dense annotation.

B Case Study

We list several cases below.

B.1 Failed Case

We show error cases in Table 5, Table 6, and Table 7. Table 5 shows the *Incorrect Information Error*. Table 6 shows the *Point Identification Error*. Table 7 shows the *Format Error*.

B.2 Success Case

We show successful cases in Table 8 and Table 9.

C Instructions

The following are instructions we used.

C.1 Data Collection Instructions

The instruction for the question filter is in Table 10. The instruction for the reference checking is in Table 11. The instruction for the paraphrase is in Table 12.

QUESTION:
How does law enforcement track down a criminal during a manhunt?

REFERENCE:
1) Do we know where he/she is likely to "go to ground?" We try to find people he/she relies on from phone taps, social media, prior crimes etc. You'd be amazed how many guys are found at their aunt's/ex's/former cellmate's place.
2) Put the word out to local TV news. In my experience they plaster the manhunt news and mugshot on their website landing page right away.
3) License plate scanner hits on patrol cars and traffic cameras, if available. This is surprisingly effective.
4) Social media phishing. Is he/she into certain drugs? Certain types of prostitutes? Is he trying to find his wife or kids who are working with victim services and in hiding (very common in bad domestic abuse cases)?

RESPONSE 1:
There are a lot of ways that law enforcement can track someone during a manhunt. First and foremost, we live in the age of technology, and technology is a really powerful tool for tracking someone. A cell phone is pretty much always transmitting your approximate location, and law enforcement can gain access to the satellite that keeps track of that location.
There are also a few tried and true methods: the public. Law enforcement will almost always provide a physical description of a suspect, complete with a photograph or artist rendition, and urge people to report anyone who fits the bill close enough to be the person. This is a powerful tool when to get anything modern, you need to interact with other humans or come into close contact with human society.
There are also recordings, photographs, and footage captured by CCTV cameras everywhere. Police may also deploy tracking hounds if they know who you are and have anything that has your smell on it and escaping on foot from persistent, trained hunting dogs is unbelievably difficult."

RESPONSE 2:
Ohhhhh so many ways. Really depends on what resources are on hand.
1. Suspect probably has friends and family. Check their houses, interview them, see what they know about the suspects hobbies and motives. Do they have a goal they're trying to achieve? Is that goal local or would it send them running elsewhere?
2. Flowing from above, roadblocks. If you generally know where your suspect is, where they might try to go, and what means of transportation are available to them, you can monitor those points of access until they show up.
3. Technology. Where did the suspect use their credit card. Oh, they pulled out cash to avoid being traced? We can see where they made the withdrawal and go from there. Is their phone still online? Can we call and trace them? Were they recently active on social media? How about their friends? A lot of those posts have geotags in the metadata

MODEL JUDGE:
- Response 1 aligns well with the standard answer by emphasizing technology (cell phone tracking), public involvement (providing descriptions and photos), and surveillance methods (CCTV cameras and tracking dogs). It touches upon finding the suspect's connections, similar to the standard answer's first point about identifying likely places to "go to ground."
- Response 2 also covers relevant points such as investigating friends and family, setting up roadblocks, and using technology like phone tracing and social media analysis. However, it delves into areas like credit card tracking, which the standard answer specifically mentions as ineffective.
Overall, **Response 1** more closely mirrors the strategies outlined in the standard answer. (*The Judgement is **incorrect** by incorrectly estimating the weights of irrelevant or unimportant keypoints.*)

Table 5: Case Study from LFQA-E from GPT-4o. This results from an incorrect information error.

QUESTION:
Why do swear words exist? What is it about swear words that make us feel better? Why don't other normal words make us feel the same way swear words do?

REFERENCE:
Swear words fall broadly into two categories: profanity and obscenity.Profanity originates with the Ten Commandments, specifically "Do not take the Lord's name in vain" in Abrahamic religions. Generally the idea though is "don't disparage or devalue the sacred". This is where the term "curse words" comes from, as you would curse someone by saying "God damn you". It's also where the term "swear words" comes from, i.e. "I swear to God". Obscenity, by contrast, is primarily words that relate to sex or using the toilet. These are activities during which you would be vulnerable to being attacked, and also that produce excrement that could be harmful for others. There is sometimes a need to discuss these things in polite company, so terms were divided into "polite" and "rude". In English, this distinction largely comes from the language of origin. English has linguistic roots in both Romance languages (ones based on Latin like French) and Germanic languages. At the time when the language was developing, higher society had more background in Romance languages while the lower classes were more Germanic. So the former words were considered refined and the latter crude and crass. Most English obscenities have close analogs in German. In modern usage, there's actually official definitions of profanity and obscenity by the FCC. Profanity is using these words in an exclamatory sense, while obscenity is using these words to describe the actions. The former is more allowable than the latter. As for why it feels good to swear, it's mostly because breaking a taboo gives a feeling of exhilaration, and if nobody actually suffers as a result, you also get to feel like you're "getting away with it". Basically, if we didn't put so much emphasis on NOT saying the words, it wouldn't feel so good when we do say them.

RESPONSE 1:
Language is a tool of communication. Expletives communicate something that you basically cannot express using polite language. There's certainly an element to them where the social transgression is part of what makes them function as they do. What is or isn't considered a swear word largely depends on cultural values. In the Middle Ages it's pretty apparent that words related to bodily functions or sexuality were not considered particularly offensive, while blasphemies were generally considered a lot more offensive than they are in most modern societies. Medieval folks appear to have been simultaneously less prudish but more fearful of the almighty.

RESPONSE 2:
Language has been used to separate classes for a very long time. Do you know why a pig is a pig in the pen but pork on your plate? Or why a cow becomes beef when you serve it up? Because one word was used by the lower class who raised the animals and the other word was used by the upper class who could afford to consume it (in this case, Normans). Words used by the lower classes frequently, but not by upper classes, will be found to be taboo in the upper classes, leading them to be essentially banned in polite society. They have power because they are taboo, and for no other reason. The word 'merde' has no power to you unless you speak French, and the word 'crap' has no power to you unless you're 6. You've been conditioned to believe these words are strong, so you feel strong when saying them. It really is that simple. It's nothing more than the placebo effect.

MODEL JUDGE:
Upon evaluating both responses against the reference answer:
- Response 1 aligns closely with the standard answer by discussing the role of swear words in communication, the significance of social transgression, and cultural variations in what is considered offensive. It also touches on historical perspectives, similar to how the reference answer differentiates profanity and obscenity based on origins and societal values.
- Response 2 introduces the concept of language separating classes and the power of taboo stemming from class distinctions. While it addresses the conditioning and perceived strength of swear words, it deviates more from the primary focus of the standard answer, which centers on cultural and linguistic origins rather than class-based language differentiation.

Therefore, **Response 1** is better.(*The Judgement is **incorrect** by observing the number of information coexisting.*)

Table 6: Case Study from LFQA-E from GPT-4o. This results from an identification error.

QUESTION:

Can someone explain Benford's Law. I get that certain numbers show up more often in large data sets, but why?

REFERENCE:

Benford's Law is typically stated about data that span multiple orders of magnitude; so we have values that are many times larger than others instead of them being close together. Next is that we have at least a bit of what is called scale-invariance: unlike many simpler probability problems, we don't assume that all numbers are equally likely, but instead that larger ones are rarer. More precisely, we want roughly the same chance to land between x and 2x as for landing between y and 2y. If those two are satisfied, Benford's Law states that the leading digit 1 is more likely than an other, and the larger digits are the less common ones. There can be no perfect examples of scale-invariance in probability. But between those lie dozens of orders of magnitude, easily more than enough within reason...

RESPONSE 1:

Benford's law says if you gather up a bunch of random numbers that appear "in the wild" and you tally up the first digits of each of these numbers then you will tend to find lots of 1's and not many 9's. This tends to happen when you have a range of numbers which is spread out over multiple orders of magnitude. The reason is when numbers are spread out over many orders of magnitude, it tends to be roughly uniform when measured on a "log scale". This means that the number of data points between x and 2x will be about the same for different values of x. In other words, there should be as many values between 100 and 200 as there are between 200 and 400.

RESPONSE 2:

Benford's law says the first digit of a number picked from a large range of numbers tends to be a 1. Think of it like this between 1 and 20, 11 of the 20 numbers start with a 1. Between 1 and 99, 11 of the 99 numbers start with a 1. This repeats for 0-200 and 0-999 etc always holding the max of "just over half" and the min of "about 11 percent" so if you average that for all ranges, you get that about 30 percent of numbers in an unknown range start with 1. Between 1 and 89, 1 number starts with 9, basically 1/89 percent. Going up to 99 brings us back to 11/99, but now 11 is the max and "almost 0" is the min, so again average it over all ranges, and you get more like 4 percent of lead numbers. You can then apply this to some fraud cases. If the numbers span multiple orders of magnitude AND should be roughly random, AND there are a lot of them, you should expect them to match Benford's law pretty well. If they don't, one of the 3 requirements is probably false. If you know the first and last are true, you can say "these probably arent actually random".

MODEL JUDGE:

- Strengths of Response 1:

Correctly identifies that Benford's Law applies to numbers spanning large ranges.

Explains the higher probability of 1 as a leading digit by comparing intervals which aligns with the scale-invariance idea.

Mentions the decreasing probability for higher digits.

Correctly notes the conditions for applying Benford's Law and its use in fraud detection.

- Weaknesses of Response 1:

Doesn't explicitly mention scale-invariance, though the intuition is somewhat captured.

The explanation for why 1 is more likely is a bit simplistic.

- Strengths of Response 2: Introduces Benford's Law in a simple, relatable context. Correctly notes that 1 and 2 are more common as leading digits.

- Weaknesses of Response 2:

Misrepresents the core reasons for Benford's Law. The explanations given are incorrect or misleading. Benford's Law is not about human behavior but about inherent properties of scale-invariant data.

Doesn't mention orders of magnitude or scale-invariance at all, which are central to the reference answer.

The references at the end are irrelevant to the actual content of the response.

No preference provided!

Table 7: Case Study For LFQA-E from GPT-4o. This results from a format error.

QUESTION:
how do metals put together with electricity cause technology and chips and computers and data etc?

REFERENCE:
Silicon along with a little bit of some other elements combined form what is called a semiconductor, of which there are 2 types, the n type and the p type. When you then sandwich these semiconductors in a specific way, it has the property that applying a current to these sandwiches causes it to become conductive, or if you invert the sandwich it has the property that it's conductive when you don't apply a current and then not conductive when you do. So essentially, what you have created is an electrical switch, which is controlled by electricity, and we call these transistors.
If you're looking at how these transistors then form a computer, you can arrange transistors to perform logical functions. A simple one would be the NOT gate, which just uses the second switch I mentioned that turns off when there is an input. An AND gate would be if you took two switches, placed them one after the other, and each switch is then controlled by an input. Both inputs need to be on for there to be an output.

RESPONSE 1:
For that, you need to look at boolian values and at turing machines. A boolian value is just a yes no answer. Yes vs no, on vs off, one vs zero. Technically you use boolian values all the time in speech. "Is Tommy in the bathroom?" You could answer "Yes", but you could also say "one", and anyone who knows Boolian will understand.
A Turing machine is just a name for any machine that uses Boolian values. It has rules, for example "if you see seven zeros in a row followed by a one, then do". It seems pretty abstract. But math people have actually proved that you can do any calculation, think any thought, perform any task, if you have the right rules designed for your turing machine.
So once we figured that out, it just became a matter of inventing the right rules for the machine, and inventing the best form for a machine to store boolian values. A transistor is always either a One or a Zero, and they can be made very small, so it makes sense to use them for your Turing Machine.

RESPONSE 2:
When metals are put together with electricity in the context of technology, it typically refers to the process of creating integrated circuits or microchips. These chips are made up of layers of different metals, such as silicon, aluminum, and copper, that are carefully arranged on a silicon wafer.
Electricity is used to power the various components on the chip, allowing them to perform their specific functions. The flow of electricity through these metal layers enables the chip to process and store information, ultimately allowing computers and other electronic devices to function.
Additionally, metals such as gold and copper are used in the construction of circuit boards and connectors within electronic devices. These metals have excellent conductivity properties, which allow for the efficient flow of electricity throughout the device.

MODEL JUDGE:
Response 1 delves into the concept of Boolean values and Turing machines, which extends the topic to how computers "think" and process data. It links the use of transistors to Turing machines and emphasizes the binary nature of data processing, which aligns well with the reference answer's explanation of logical functions and binary systems accessible through transistors.
Response 2 focuses on the physical aspects of creating chips, such as the use of various metals and their conductivity. While it provides a general overview of chip construction and functionality, it lacks depth on the conceptual workings, like the function of transistors and binary systems, which is a significant aspect of the reference answer.
The reference answer provides an explanation centered around the role of transistors as switches and their organization to perform logical operations. It links this to the representation of numbers and data processing in computers through binary systems, which Response 1 addresses more comprehensively compared to Response 2. Therefore, **Response 1** is better. *(The evaluation is correct!)*

Table 8: Case Study For LFQA-E from Deepseek-V3.

QUESTION:
How did human calculate time?

REFERENCE:
For a long time we didn't and no one cared. However, once you start farming, you need to know what season it is, what hour of the day it is more or less, so you know what you can and should do on the fields for a bountiful yield. The easiest way to do is to watch the celestial bodies, namely the two major ones, the Sun and the Moon (the Stars also helped, but eventually it was more of a tool for navigation). Over time people figured out that the Sun shifts, days are not evenly distributed over the year, but luckily it goes in cycles, so it is predictable. Once you make a chart of these shifts throughout the year, you got yourself more or less a calendar. Civilizations with accurate-enough calendars survived and thrived. Calendars got more and more accurate over time as our math and science got better and here we are, keeping time down to split seconds. By the way, it does not fit perfectly. We accumulate quite the error over time, but the Gregorian Calendar is designed as a way to correct for these accumulative errors. The Persian Solar Calendar is the most accurate one, because the new year starts after a celestial event. Gregorian calendar has a bigger focus on religious events rather than any celestial event, so it needs corrections. We have a very accurate time measurement, but that is NOT what is used on the daily, because we don't need it.

RESPONSE 1:
Absolutely nothing about the way humans decided to calculate time "fits neatly" into Earth's behaviours. Years are actually 365 days plus 6 hours and 9 minutes, days are actually 23 hours and 56 minutes long.Ot her time systems have been attempted, with the Romans dividing day and night into 12 hours, which varied in actual length as the days and nights shifted through the year. They also had the "Civil Day" system: formal names given to time according to certain markers (such as separate named times for "rooster is crowing" and "rooster stops crowing", or "time we light candles" and "bed time"). Humans have been struggling with measuring it forever.

RESPONSE 2:
Time has been calculated and measured in various ways throughout history. One of the earliest methods used by ancient civilizations was observing the movement of celestial bodies such as the sun, moon, and stars. By tracking the positions of these objects in the sky, early astronomers were able to develop calendars and timekeeping systems. The modern concept of time measurement is based on the International System of Units (SI). In terms of dividing the day into hours, minutes, and seconds, this system was developed by the ancient Babylonians and Egyptians. The Babylonians used a base-60 system for counting, which led to the division of the hour into 60 minutes and the minute into 60 seconds. The Egyptians also used a similar system of timekeeping. Overall, the calculation of time has evolved over centuries through a combination of astronomical observations, scientific advancements, and cultural practices. The precise measurement of time today is achieved through the use of atomic clocks and coordinated universal time (UTC) to ensure accurate timekeeping worldwide.

MODEL JUDGE:
The reference answer emphasizes the historical development of time measurement, focusing on the transition from no precise timekeeping to the necessity of using celestial bodies for farming.
Response 1 discusses the inconsistency and challenges of time measurements historically, noting the Roman system and difficulties faced by humans in tracking time over time. While it touches on early attempts to measure time, it doesn't delve specifically into celestial observations or the evolution of calendars as much.
Response 2 covers the historical aspect of using celestial bodies and tracks the progression to modern timekeeping with atomic clocks. It also discusses the Babylonian and Egyptian contributions to dividing time. This response aligns more closely with the standard answer by tracing the transition from ancient methods to modern precision, even mentioning cultural influences on timekeeping.
Overall, while both responses provide historical context, Response 2 is closer to the reference answer as it better follows the progression of timekeeping from ancient observations leading to the precise systems we have today. Therefore, **Response 2** is better. *(The evaluation is correct!)*

Table 9: Case Study For LFQA-E from Deepseek-V3.

PROMPT FOR LLM FILTER.

```
# Question Filtering Instructions

## Objective
Filter out questions that are either unclear in
↪  their description or too broad to provide a
↪  meaningful reference.

## Filtering Criteria

### 1. Unclear Questions
Reject questions that exhibit:
* Ambiguous wording or phrasing
* Multiple possible interpretations
* Missing critical context or parameters
* Vague or undefined terms
* Grammatical issues that obscure meaning
* Incomplete or fragmented thoughts

### 2. Overly Broad Questions
Reject questions that:
* Request information on topics with no
↪  reasonable boundaries
* Would require encyclopedic or book-length
↪  answers
* Ask for opinions on vast, multi-faceted
↪  subjects
* Lack specific focus or scope constraints
* Would yield references too general to be useful
* Cover multiple unrelated topics simultaneously

## Process
1. Read the question carefully and completely
2. Evaluate against both clarity and breadth
↪  criteria
3. Make a filtering decision:
   * **PASS**: Question is clear and
   ↪  appropriately scoped
   * **REJECT - UNCLEAR**: Question lacks clarity
   ↪  (provide specific reason)
   * **REJECT - TOO BROAD**: Question is overly
   ↪  broad (provide specific reason)

## Examples of Questions to Reject
* "What about technology?" (unclear)
* "Explain everything about human history" (too
↪  broad)
* "How does stuff work in general?" (both unclear
↪  and too broad)
* "What are all the factors affecting everything
↪  in the world?" (too broad)

## Examples of Questions to Pass
* "What is the boiling point of water at sea
↪  level?"
* "How does photosynthesis work in green plants?"
* "What were the main causes of World War I?"
```

Table 10: Prompt for LLM Filter.

PROMPT FOR LLM CHECK.

```
## Objective
Evaluate whether the provided reference contains
↪  all necessary information to fully answer
↪  the given question.

## Process
1. Analyze the question to identify:
   * The main topic being asked about
   * Specific information points required for a
   ↪  complete answer
   * Any implied sub-questions or requirements

2. Thoroughly examine the reference material for:
   * Direct answers to the question's
   ↪  requirements
   * Necessary context and background information
   * Supporting details and evidence

3. Perform a gap analysis:
   * Match each question requirement against
   ↪  information in the reference
   * Identify any missing information points
   * Note ambiguities or incomplete explanations

4. Make a determination:
   * If all required information is present: Mark
   ↪  as "SUFFICIENT"
   * If partial information is present: Mark as
   ↪  "PARTIALLY SUFFICIENT" and list missing
   ↪  elements
   * If critical information is missing: Mark as
   ↪  "INSUFFICIENT" and explain what's missing

5. Provide a brief explanation supporting your
↪  assessment

## Important Considerations
* Focus only on information completeness, not
↪  quality or accuracy
* Consider both explicit and implicit
↪  information in the reference
* Do not supplement missing information from
↪  external knowledge
* Be specific about any information gaps
↪  identified
```

Table 11: Prompt for LLM Check.

PROMPT FOR LLM PARAPHRASE.

```
## Objective
Transform the provided response into a more
↪  verbose version while strictly preserving
↪  the original meaning and information.

## Requirements
- Expand the original text by adding descriptive
↪  language, elaborations, and explanatory
↪  phrases
- Maintain complete fidelity to the original
↪  information—do not introduce any new facts,
↪  claims, or insights
- Preserve the tone and intent of the original
↪  message
- Use stylistic techniques such as:
  * Adding clarifying phrases and parenthetical
  ↪  explanations
  * Employing more elaborate sentence structures
  * Incorporating synonyms and varied vocabulary
  * Adding transitional phrases between ideas
  * Expanding brief points into full explanations
- Ensure the final text feels natural and not
↪  artificially inflated

## Process
1. Thoroughly analyze the original response to
↪  understand its complete meaning
2. Identify core points and supporting details
3. Expand each point methodically while
↪  maintaining the original structure
4. Review to confirm no new information has been
↪  introduced
5. Polish the text for readability and flow
```

Table 12: Prompt for LLM Paraphrase.

C.2 English LLM Evaluation Instruction

The instruction for all LLMs and LRMs is in Table 13. The instruction for Prometheus series is in Table 14. The instruction for Auto-J is in Table 15.

PROMPT FOR ENGLISH LLM EVALUATION.

```
## Objective
Transform the provided response into a more
↪  verbose version while strictly preserving
↪  the original meaning and information.

## Requirements
- Expand the original text by adding descriptive
↪  language, elaborations, and explanatory
↪  phrases
- Maintain complete fidelity to the original
↪  information—do not introduce any new facts,
↪  claims, or insights
- Preserve the tone and intent of the original
↪  message
- Use stylistic techniques such as:
  * Adding clarifying phrases and parenthetical
  ↪  explanations
  * Employing more elaborate sentence structures
  * Incorporating synonyms and varied vocabulary
  * Adding transitional phrases between ideas
  * Expanding brief points into full explanations
- Ensure the final text feels natural and not
↪  artificially inflated

## Process
1. Thoroughly analyze the original response to
↪  understand its complete meaning
2. Identify core points and supporting details
3. Expand each point methodically while
↪  maintaining the original structure
4. Review to confirm no new information has been
↪  introduced
5. Polish the text for readability and flow
```

Table 13: Prompt for English LLM Evaluation.

PROMPT FOR PROMETHEUS EVALUATION.

```
###Task Description:
An instruction (might include an Input inside
↪  it), a response to evaluate, and a score
↪  rubric representing a evaluation criteria
↪  are given.
1. Write a detailed feedback that assess the
↪  quality of two responses strictly based on
↪  the given score rubric, not evaluating in
↪  general.
2. After writing a feedback, choose a better
↪  response between Response A and Response B.
↪  You should refer to the score rubric.
3. The output format should look as follows:
↪  "Feedback: (write a feedback for criteria)
↪  [RESULT] (A or B)"
4. Please do not generate any other opening,
↪  closing, and explanations.

###Instruction:
{orig_instruction}

###Response A:
{response_A}

###Response B:
{response_B}

###Score Rubric:
{score_rubric}

###Feedback:
```

Table 14: Prompt for Prometheus Evaluation.

PROMPT FOR AUTO-J EVALUATION.

```
You are a helpful and precise assistant for
↪  checking the quality of the feedback.
Two pieces of feedback have been provided for the
↪  same response to a particular query. Which
↪  one is better with regard to their
↪  correctness, comprehensiveness, and
↪  specificity to the query?

[BEGIN DATA]
***
[Query]: {prompt}
***
[Response]: {response}
***
[Feedback 1]: {feedback1}
***
[Feedback 2]: {feedback2}
***
[END DATA]

Please choose from the following options, and
↪  give out your reason in the next line.
A: Feedback 1 is significantly better.
B: Feedback 2 is significantly better.
C: Neither is significantly better."""
```

Table 15: Prompt for Auto-J Evaluation.

D Annotation Recipe

We show the annotation recipe below.

PROMPT FOR LLM CHECK.

```
## Overview
This guide helps annotators evaluate and compare long-form responses against a reference to determine
↪  which response is more informative and complete. The process uses a triple-choice format
↪  (Response A Better, Response B Better, or Tie).

## Key Principles
- Focus on **factuality** and **completeness** according to the reference
- Fluency is not a primary evaluation criterion (all responses are expected to be fluent)
- Use information units as the basic evaluation unit
- Minimize bias through systematic comparison

## Prerequisites
- Domain knowledge relevant to the question topic
- Understanding of the subject matter through academic coursework or professional experience
- Ability to maintain focus during paragraph-level analysis

## Evaluation Process

### Step 1: Extract Key Information from Reference
1. Read the question carefully to understand what information is being requested
2. Read the reference thoroughly
3. Identify and list all key information units that:
   - Directly answer the question
   - Provide necessary context or background
   - Support the main answer with evidence or examples
4. Organize key information into logical categories or themes

### Step 2: Check for Key Information in Responses
For each response (A and B):
1. Read the response completely
2. Map each key information unit from the reference to the response
3. Mark which key information units are:
   - Present and accurate
   - Present but inaccurate
   - Missing entirely
4. Note any additional information not in the reference

### Step 3: Handle Response Content
1. Evaluate additional information:
   - Is it relevant to the central topic?
   - Does it enhance understanding or is it verbose/unnecessary?
2. Identify intertwined information:
   - For sentences containing both correct and incorrect information, separate the components
   - Assess the impact of any inaccuracies on the overall response quality

### Step 4: Compare Overlapping Information
1. Compare how well each response covers the key information units
2. Consider:
   - Completeness: Which response includes more key information?
   - Accuracy: Which response presents information more correctly?
   - Relevance: Which response stays more focused on the question?
3. Compare the quality of overlapping information presentation

### Step 5: Make Final Decision
Select one of three options:
- **Response A is Better**: A contains more key information and/or presents it more accurately
- **Response B is Better**: B contains more key information and/or presents it more accurately
- **Tie**: Both responses are comparable in information coverage and accuracy

## Common Pitfalls to Avoid
1. Losing focus due to long paragraphs - use the systematic approach
2. Allowing domain bias to influence decisions - stick to the reference
3. Confusing eloquence with accuracy
4. Missing subtle differences between comparable responses
```

Table 16: Annotation recipe of LFQA-E.