MinosEval: Distinguishing Factoid and Non-Factoid for Tailored Open-Ended QA Evaluation with LLMs

Yongqi Fan[*], **Yating Wang**[*], **Guandong Wang**
Jie Zhai, **Jingping Liu**, **Qi Ye**[†], **Tong Ruan**[†]
School of Information Science and Engineering, East China University
of Science and Technology, Shanghai, China
`johnnyfans@mail.ecust.edu.cn`, `ruantong@ecust.edu.cn`

Abstract

Open-ended question answering (QA) is a key task for evaluating the capabilities of large language models (LLMs). Compared to closed-ended QA, it demands longer answer statements, more nuanced reasoning processes, and diverse expressions, making refined and interpretable automatic evaluation both crucial and challenging. Traditional metrics like ROUGE and BERTScore struggle to capture semantic similarities due to different patterns between model responses and reference answers. Current LLM-based evaluation approaches, such as pairwise or listwise comparisons of candidate answers, lack intuitive interpretability. While pointwise scoring of each response provides some descriptions, it fails to adapt across different question contents. Most notably, existing methods overlook the distinction between factoid and non-factoid questions. To address these challenges, we propose **MinosEval**, a novel evaluation method that first distinguishes open-ended questions and then ranks candidate answers using different evaluation strategies. For factoid questions, it applies an adaptive key-point scoring strategy, while for non-factoid questions, it uses an instance-aware listwise ranking strategy. Experiments on multiple open-ended QA datasets, including self-built ones with more candidate responses to complement community resources, show that MinosEval better aligns with human annotations and offers more interpretable results.

1 Introduction

Open-ended question answering (QA) is a fundamental task type in comprehensive large language modeling (LLM) evaluation benchmarks and plateforms (Srivastava et al., 2023; Contributors, 2023; Liu et al., 2024b; Myrzakhan et al., 2024; Liu et al., 2024a). Unlike closed-ended questions, which generally elicit brief, predefined responses (e.g., multiple-choice question (Wang et al., 2024b), true/false question (Luo et al., 2023), and close question), open-ended questions (Kantharaj et al., 2022; Tao et al., 2024; Lin et al., 2024) necessitate the generation of longer, more detailed answers that require complex reasoning, nuanced understanding, and diverse modes of expression. Consequently, the evaluation of open-ended QA has become a significant area of research, with an increasing focus on developing methods and benchmarks (Amirizaniani et al., 2024; Yang et al., 2024b; Wang et al., 2024c) that not only measure the quality of the model responses but also provide interpretability and alignment with human annotation.

Traditional evaluation metrics for free-text responses, such as BLEU (Papineni et al., 2002), ROUGE (Lin, 2004), and BERTScore (Zhang et al.), focus on lexical overlap or semantic similarity. While useful for tasks with clear answers, they are less effective for open-ended QA, where responses vary in style and content. These metrics also struggle with complex open-ended answers, involving reasoning, creativity, and entailment.

In recent years, evaluation methods have shifted toward leveraging LLMs for automatic evaluation. These methods typically involve employing pairwise comparisons for Elo (Boubdir et al., 2023; Harang, 2024), listwise comparisons where LLMs rank responses based on relevance or quality, and pointwise approaches (Liang et al.; Vu et al., 2024) that score responses on predefined dimensions. Meanwhile, some studies conducted pairwise and pointwise supervised fine-tuning to obtain dedicated LLMs for evaluation (Kim et al., 2024; Li et al., 2023a). While promising, these methods have notable limitations. Pairwise methods involve multiple comparisons, leading to $O(n^2)$ complexity for n responses. Pointwise methods, though scoring on predefined criteria, e.g., “Fluency” and “Truthfulness”, often fail to be adapted to each

---

[*] Equal contribution.

[†] Corresponding authors.

Findings of the Association for Computational Linguistics: ACL 2025, pages 10517–10548
July 27 - August 1, 2025 ©2025 Association for Computational Linguistics

Factoid

Question: List the five tallest mountains on Earth and the average of their peaks.
Answer: the five tallest mountains on earth and their altitudes are: 1.Mount Everest: 8848 meters **...** average of their altitudes is (8848+8611+8586+8516+8485)/5 = 8609.2 meters.

Question: Did the Magellanic Flotilla use a sextant to measure latitude and longitude while traveling around the world?
Answer: ... Magellan's voyage around the world took place from 1519 to 1522, and the principle of the sextant was proposed by Isaac Newton, who was born on January 4, 1643 **...** and the use of a sextant was not possible.

Non-Factoid

Question: what are the themes and symbolism in the short story 'The Raven' by Edgar Allan Poe, one of the major writers of American literature?
Answer: 'The Raven' is a poem by Edgar Allan Poe about the sorrow and loneliness of losing a loved one. The themes focus on tragedy **...** revealing the powerlessness and pain of human beings in the face of eternal loss.

Question: ... please generate a resume for me that is customized for the position of [Sales Manager] **...** My name is [Ma Ming Feng Xiao Xiao] and I am a software developer, you model the rest of the information.
Answer: Ma Ming Feng Xiao Xiao **...** Senior professional with 10 years of experience working across multiple disciplines, including software development and project management. Adept at analyzing market needs, driving team execution, and **...**

Figure 1: Typical Samples of Factoid and Non-Factoid Open-Ended QA.

specific question context. Additionally, pairwise and pointwise methods suffer from the tie-breaking problem. In addition, pairwise and listwise methods lack intuitive explanations, making their rankings difficult for human evaluators to interpret.

More notably, these LLM-based approaches overlook the distinction between factoid and non-factoid questions in open-ended QA. Factoid (Jiang et al., 2019; Gaikwad and Patil, 2023) and non-factoid (Lakshmi and Sindhu, 2023; Bolotova-Baranova et al., 2023a) questions exhibit a clear distinction. Factoid questions aim to elicit answers based on objective, real-world facts or entities, with a relatively fixed and clear scope, e.g., "*List the five highest mountains on Earth and the average elevation of their peaks*". Of course, additional sentences can be included in the answer to ensure semantic coherence. In contrast, non-factoid questions offer more creative freedom, with the primary requirement being to meet the demands of the question without necessarily including critical, fact-based information, e.g., "*Please help me write homophone joke*". By failing to account for this distinction, such a one-size-fits-all strategy undermines the precision and effectiveness of evaluation.

In this paper, we propose **MinosEval**, a novel two-stage approach for evaluating open-ended QA and providing corresponding interpretive information. It distinguishes between factoid and non-factoid questions based on their semantic and content differences, employing tailored evaluation strategies for each type of question. For factoid questions, we use an adaptive key-point scoring strategy that extracts key factoid points from a given reference answer. It then compares how each response entails these key points using a natural language inference (NLI) model, inspired by Bohnet et al. (2022). For non-factoid questions, we apply an instance-aware listwise ranking approach, generating five levels of silver answer instances to enhance the LLM's performance in directly ranking these more creative and diverse responses.

Our approach tackles the challenges of existing methods in open-ended QA evaluation and offers several advantages. First, it distinguishes between different types of open-ended questions, enabling a more tailored evaluation strategy. It also provides clear ranking guidelines based on the characteristics of factoid and non-factoid questions, including key points and silver instances to enhance interpretability. Moreover, the guidelines are adaptive to the specific questions, and the entire process is fully automated, requiring no manual intervention.

We conducted extensive experiments on four datasets. Following the approach of Yang et al. (2024b), we performed five sets of sampling to construct ANTIQUE_S5 and TREC-DL-NF_S5 from the publicly available open quality assurance datasets ANTIQUE (Hashemi et al., 2020) and TREC-DL-NF (Craswell et al., 2020, 2021). Additionally, we created two self-built datasets, AlignBench_Minos and GaokaoBench_Minos, which contain a larger number of candidate responses based on AlignBench (Liu et al., 2023b) and GaokaoBench (Zhang et al., 2023) to supplement the resources of the research community. The results demonstrate that our method outperforms existing LLM-based evaluation approaches for open-ended QA. We also present case studies in which the key points for factoid questions and the silver answer instances for non-factoid questions provide a valuable interpretive foundation.

Table 1: Dataset statistics and description. The columns indicate the dataset name, source, language, number of reference answers, samples, and candidate model responses to be evaluated.

Datasets	Source	Language	# Res	# Ref	# Samples		
					Factoid	Non-factoid	Total
ANTIQUE_S5	ANTIQUE (Hashemi et al., 2020)	English	2 to 4	2	96×5	404×5	500×5
TREC-DL-NF_S5	TREC-DL-NF (Craswell et al., 2020, 2021)	English	2 to 4	1	28×5	27×5	55×5
AlignBench_Minos	AlignBench (Liu et al., 2023b)	Chinese	6	1	299	384	683
GaokaoBench_Minos	GaokaoBench (Zhang et al., 2023)	Chinese	6	1	158	247	405

We hope this work contributes to the effective evaluation of LLMs' performance in open-ended QA and promotes further research within the LLM community. Our datasets, evaluation results, and code for **MinosEval** are publicly available[1].

2 Related Work

2.1 Open-ended QA.

Question answering (QA) is a key task in natural language processing, aimed at providing accurate answers to satisfy the user's information need or request. Depending on the scope of the answer, QA systems are typically divided into closed-ended and open-ended types. Closed-ended QA includes formats like multiple choice (Wang et al., 2024b), true/false (Luo et al., 2023), and close question (Yu et al., 2023), while open-ended QA allows for more diverse, unbounded responses, typically divided into factoid and non-factoid questions (Agustianingsih and Mahmudi, 2019). Common tasks in open-ended QA include reading comprehension (Liu et al., 2023a), summarization (Fabbri et al., 2021), and writing (Ngo et al., 2024).

The advent of large language models (LLMs) such as ChatGPT (OpenAI, 2022) and Gemini (Google, 2023) has made open-ended QA a popular task in LLM evaluation benchmarks. This task is essential for evaluating the ability of LLMs to generate complex, creative, and contextually relevant responses, which require advanced reasoning. Moreover, open-ended QA has been used to explore LLM limitations in understanding nuanced language (Dentella et al., 2024) and capturing human intent and emotions (Amirizaniani et al., 2024), further highlighting its importance as a benchmark for evaluating reasoning and cognitive abilities (Yang et al., 2024b; Wang et al., 2024d).

2.2 Open-ended QA Evaluation

Traditional Methods: Matching-based evaluation methods for open-ended QA include BLEU (Papineni et al., 2002) and ROUGE (Lin, 2004). These methods primarily focus on matching the n-grams between the generated and reference texts, which often overlooks the semantic meaning. For example, (Krishna et al., 2021) highlights that ROUGE is ineffective for long-text QA tasks. LM-based methods aim to capture semantic similarity better. BERTScore (Zhang et al.) easuring semantic similarity between embeddings. Further work introduces BEM (Bulian et al., 2022), a BERT-based model for assessing semantic equivalence between candidate and reference answers, and PEDANTS (Li et al., 2024b), which employs rule-guided rubrics and lightweight neural scoring. However, they struggle with complex questions and diverse answers.

Human Evaluation: Human evaluation remains the golden standard in open-ended QA (Bolotova-Baranova et al., 2023b), providing more accurate and comprehensive feedback (Chang et al., 2024). However, it is resource-intensive and difficult to scale, limiting its use in large-scale evaluations.

LLM-based Evaluation: With the development of LLMs, they have demonstrated evaluation capabilities similar to human annotations (Li et al., 2023b). Evaluation methods based on LLMs primarily include pointwise (Liu et al., 2023c), pairwise (Shi et al., 2024), and listwise approaches. Recent research has explored these methods, such as using textual entailment to evaluate model responses in open-ended QA (Yao and Barbosa, 2024). PERSE (Wang et al., 2024a) combines pointwise and pairwise methods for story generation, while MATEval (Li et al., 2024a) simulates human collaborative discussions and integrates multiple agents to evaluate open-ended text. While LLM-based evaluation offers flexibility, it faces challenges, including positional bias (Shi et al., 2024), verbosity bias, self-bias (Wang et al., 2023),

---

[1] `https://github.com/JOHNNY-fans/MinosEval`

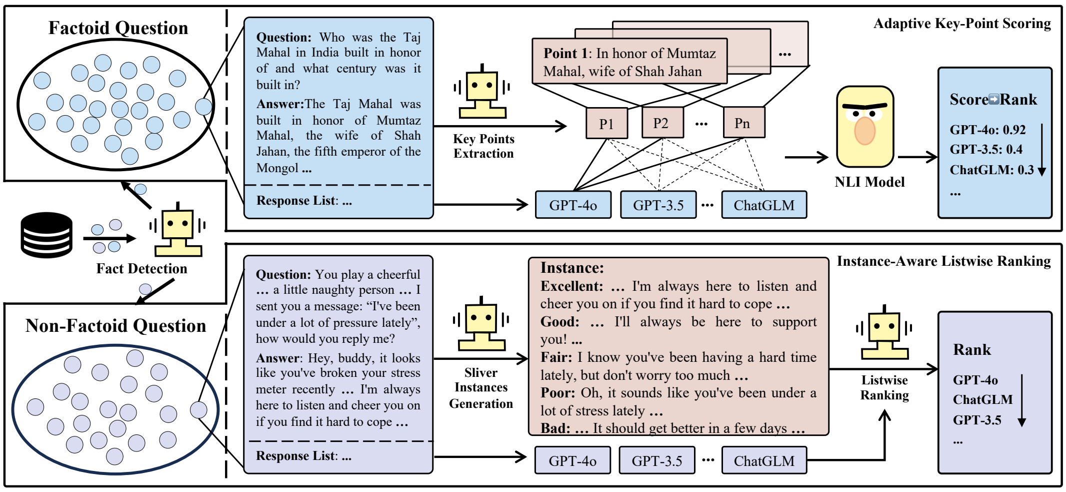

Factoid Question
Question: Who was the Taj Mahal in India built in honor of and what century was it built in?
Answer:The Taj Mahal was built in honor of Mumtaz Mahal, the wife of Shah Jahan, the fifth emperor of the Mongol ...
Response List: ...
Key Points Extraction
Point 1: In honor of Mumtaz Mahal, wife of Shah Jahan
...
P1
P2
...
Pn
GPT-4o
GPT-3.5
...
ChatGLM
NLI Model
Adaptive Key-Point Scoring
Score➡Rank
GPT-4o: 0.92
GPT-3.5: 0.4
ChatGLM: 0.3
...
Fact Detection
Non-Factoid Question
Question: You play a cheerful ... a little naughty person ... I sent you a message: “I've been under a lot of pressure lately”, how would you reply me?
Answer: Hey, buddy, it looks like you've broken your stress meter recently ... I'm always here to listen and cheer you on if you find it hard to cope ...
Response List: ...
Sliver Instances Generation
Instance:
Excellent: ... I'm always here to listen and cheer you on if you find it hard to cope ...
Good: ... I'll always be here to support you! ...
Fair: I know you've been having a hard time lately, but don't worry too much ...
Poor: Oh, it sounds like you've been under a lot of stress lately ...
Bad: ... It should get better in a few days ...
Instance-Aware Listwise Ranking
Listwise Ranking
GPT-4o
GPT-3.5
...
ChatGLM
Rank
GPT-4o
ChatGLM
GPT-3.5
...

Figure 2: The MinosEval Workflow: Distinguishing Factoid vs. Non-Factoid Questions, Adaptive Key-Point Scoring, and Instance-Aware Listwise Ranking.

and efficiency issues.

Meanwhile, several studies have noticed the distinction between factoid and non-factoid questions. For factoid questions, some work (Min et al., 2023; Cook et al., 2024; Yan et al., 2024; Hu et al., 2024) has attempted to decompose the answers into discrete facts or knowledge triplets, and then build a validation checklist. In contrast, for non-factoid questions, there are some expert judges (Kim et al., 2024; Li et al., 2023a) with specialized training to evaluate LLM performance have been proposed.

3 Methodology

In this section, we begin by formally defining the open-ended QA evaluation task, explaining the principles of distinguishing between factoid and non-factoid questions, and then presenting the details of **MinosEval**.

3.1 Preliminary

Task Definition. Given an open-ended question q and the corresponding n candidate model responses $\mathbb{R} = \{r_1, r_2, \ldots, r_n\}$, where r_i represents the i-th model response to be evaluated, the goal is to implement a specific strategy and produce a ranked order of these model responses. The reference answer for question q is denoted as a. Furthermore, it is important to note that the final ranked order strictly avoids ties in this paper.

Factoid vs. Non-Factoid Questions. Distinguishing between factoid and non-factoid open-ended questions is a central motivation of this paper. To this end, we provide clear definitions, which will also guide the human annotation of the datasets. Building on previous works Jiang et al. (2019) and Hashemi et al. (2020), we define the following: First, in open-ended QA, the answer is arbitrary free text, not restricted to predefined items or fixed responses. Second, for factoid questions, the answer is expected to contain explicit information, such as entities, common knowledge, and facts. Alternatively, due to the limitations of the question, the scope of the key elements of the answer is narrowly defined. In contrast, non-factoid questions do not require key factoid information and are less constrained in terms of answer content. It is worth noting that in our setting, the additional text included in the model responses, e.g., reasoning and thought processes, does not affect the factoid nature of the question. We present some typical samples of factoid and non-factoid QA in Figure 1 to facilitate a more intuitive understanding.

3.2 MinosEval

Figure 2 shows how our MinosEval works. Specifically, given an open-ended question q, a reference answer a, and model responses $\mathbb{R}$ that need to be evaluated, this sample is first distinguished by an LLM-based fact detection module into either factoid cluster C_f or non-factoid cluster C_{nf}. For factoid samples, the final rank orders $\mathcal{R}_f$ are generated by applying the adaptive key-point scoring strategy. For non-factoid samples, the final rank orders $\mathcal{R}_{nf}$ are produced by the instance-aware listwise ranking strategy. The implementation details of each module are as follows.

Table 2: Performance on AlignBench_Minos and GaokaoBench_Minos: Kendall's Tau (K), Spearman's Rho (S), and Rank-biased Overlap (RBO) for p=0.5 and p=0.9. "†" denotes manually classifying factoid and non-factoid questions.

	Method	AlignBench_Minos				GaokaoBench_Minos			
		K	S	RBO (p=0.5)	RBO (p=0.9)	K	S	RBO (p=0.5)	RBO (p=0.9)
Automatic Metrics	BLEU	15.41	19.47	36.24	79.98	11.08	12.85	36.78	79.81
	ROUGE-L	13.38	16.69	37.93	80.09	4.89	5.81	35.15	78.95
LM-based Metrics	BERTScore	13.62	18.12	37.81	80.15	12.92	16.44	40.21	80.66
Naive LLM Evaluation	Pointwise	32.94	41.37	47.41	83.59	31.06	38.55	42.64	82.43
	Pairwise	38.66	47.89	51.59	84.85	45.71	54.93	56.90	86.45
	Listwise	41.47	51.46	54.53	85.64	55.33	64.66	61.59	88.35
Others	LINKAGE	35.75	43.82	52.97	84.88	37.45	44.61	52.47	84.86
Ours	MinosEval	45.28	54.89	56.30	86.28	56.12	65.77	63.36	88.67
	MinosEval†	**47.68**	**57.38**	**57.09**	**86.62**	**59.77**	**70.30**	**65.26**	**89.31**

Table 3: Performance on ANTIQUE_S5 and TREC-DL-NF_S5: Kendall's Tau (K), Spearman's Rho (S), and Rank-biased Overlap (RBO) for p=0.5 and p=0.9. "†" denotes manually classifying factoid and non-factoid questions.

	Method	ANTIQUE_S5				TREC-DL-NF_S5			
		K	S	RBO (p=0.5)	RBO (p=0.9)	K	S	RBO (p=0.5)	RBO (p=0.9)
Automatic Metrics	BLEU	26.08$_{\pm 0.0247}$	29.02$_{\pm 0.0249}$	66.15$_{\pm 0.0086}$	91.85$_{\pm 0.0023}$	28.11$_{\pm 0.0266}$	31.40$_{\pm 0.0356}$	65.42$_{\pm 0.0218}$	91.39$_{\pm 0.0046}$
	ROUGE-L	22.51$_{\pm 0.0330}$	25.45$_{\pm 0.0325}$	64.94$_{\pm 0.0111}$	91.59$_{\pm 0.0027}$	34.56$_{\pm 0.0535}$	37.61$_{\pm 0.0616}$	68.30$_{\pm 0.0351}$	92.07$_{\pm 0.0081}$
LM-based Metrics	BERTScore	29.56$_{\pm 0.0165}$	33.10$_{\pm 0.0164}$	66.94$_{\pm 0.0132}$	92.02$_{\pm 0.0028}$	44.62$_{\pm 0.0672}$	50.23$_{\pm 0.0723}$	71.61$_{\pm 0.0382}$	93.01$_{\pm 0.0088}$
	BEM	39.75$_{\pm 0.0178}$	43.37$_{\pm 0.0221}$	76.32$_{\pm 0.0073}$	93.62$_{\pm 0.0019}$	46.42$_{\pm 0.0528}$	51.93$_{\pm 0.0491}$	75.05$_{\pm 0.0145}$	93.21$_{\pm 0.0033}$
	PEDANTS	37.85$_{\pm 0.0206}$	41.49$_{\pm 0.0218}$	74.03$_{\pm 0.0070}$	93.18$_{\pm 0.0016}$	48.12$_{\pm 0.0505}$	53.82$_{\pm 0.0393}$	74.61$_{\pm 0.0189}$	93.31$_{\pm 0.0044}$
Naive LLM Evaluation	Pointwise	50.36$_{\pm 0.0155}$	55.99$_{\pm 0.0150}$	77.08$_{\pm 0.0053}$	94.42$_{\pm 0.0012}$	54.79$_{\pm 0.0335}$	60.33$_{\pm 0.0282}$	73.58$_{\pm 0.0171}$	93.37$_{\pm 0.0051}$
	Pairwise	63.53$_{\pm 0.0113}$	68.91$_{\pm 0.0093}$	81.45$_{\pm 0.0037}$	95.46$_{\pm 0.0003}$	63.21$_{\pm 0.0269}$	69.93$_{\pm 0.0269}$	77.50$_{\pm 0.0175}$	94.40$_{\pm 0.0045}$
	Listwise	62.55$_{\pm 0.0359}$	68.56$_{\pm 0.0317}$	83.30$_{\pm 0.0168}$	95.93$_{\pm 0.0041}$	65.82$_{\pm 0.0463}$	72.95$_{\pm 0.0344}$	79.47$_{\pm 0.0403}$	94.93$_{\pm 0.0085}$
Others	LINKAGE	52.65$_{\pm 0.0201}$	57.71$_{\pm 0.0169}$	79.89$_{\pm 0.0108}$	95.17$_{\pm 0.0027}$	65.58$_{\pm 0.0335}$	72.29$_{\pm 0.0238}$	79.36$_{\pm 0.0333}$	94.89$_{\pm 0.0066}$
Our	MinosEval	64.93$_{\pm 0.0075}$	68.83$_{\pm 0.0061}$	84.69$_{\pm 0.0064}$	96.19$_{\pm 0.0015}$	65.45$_{\pm 0.0213}$	69.56$_{\pm 0.0227}$	82.03$_{\pm 0.0210}$	95.31$_{\pm 0.0041}$
	MinosEval†	**65.97**$_{\pm \mathbf{0.0097}}$	**69.91**$_{\pm \mathbf{0.0102}}$	**84.79**$_{\pm \mathbf{0.0075}}$	**96.27**$_{\pm \mathbf{0.0016}}$	**68.61**$_{\pm \mathbf{0.0129}}$	**73.09**$_{\pm \mathbf{0.0248}}$	**84.38**$_{\pm \mathbf{0.0164}}$	**95.82**$_{\pm \mathbf{0.0036}}$

3.2.1 LLM-based Fact Detection

In light of the differences between factoid and non-factoid open-ended QA discussed in Section 3.1, it is necessary to treat these two types of questions differently. However, manual annotation is costly, so instead, we leverage the instruction-following and few-shot learning capabilities of advanced LLMs (Brown et al., 2020), such as GPT-4o (OpenAI, 2024). Specifically, we employ an in-context learning approach to develop a simple yet effective fact detection module, which classifies samples based on the questions and reference answers by designing the task prompt and providing suitable demonstration examples. The specific prompt is shown in Figure C7 in the Appendix.

3.2.2 Adaptive Key-Point Scoring

For factoid open-ended questions, the reference answer typically contains facts, entities, or common knowledge, or is constrained by the question's content, such as in rewriting tasks. These key points must be considered during ranking, so we propose an adaptive key-point scoring strategy. Key information is extracted from the reference answers, but unlike conventional pointwise evaluation, it adapts to the specific question rather than fixed criteria. We designed an LLM-based key-point extraction module, with the prompt provided in Figure C10 in the Appendix.

Having obtained these key points, we proceed to calculate scores for each model response to be evaluated. This is modeled as a Natural Language Inference (NLI) task, where the degree of entailment and contradiction between each model response and each key point are computed. The final ranking of the model responses is then determined based on these scores. The formal definition is shown in the Formula 1.

$$\mathcal{R}_f = \{\text{Sort}\left(S_f(q_i, a_i, \mathbb{R}_i), \mathbb{R}_i\right) \mid q_i, a_i, \mathbb{R}_i \in C_f\} \quad (1)$$

where $\mathcal{R}_f$ represents the ranking results of samples in the factoid cluster C_f, q_i denotes the i-th question, a_i denotes the i-th reference answer, and $\mathbb{R}_i$ refers to the list of candidate model responses. Sort denotes a simple sorting function that sorts $\mathbb{R}_i$ based on scores, and S_f is the scoring function that applies the adaptive key-point scoring strategy, which is specifically shown in the Formula 2.

$$S_f(q,a,\mathbb{R}) = \{\frac{1}{|\mathbb{K}|}\sum_{k_j\in\mathbb{K}} (\mathrm{NLI}\,(r_i,k_j)) \mid r_i \in \mathbb{R}\}, \quad \text{note } \mathbb{K} = \mathrm{extractKeyPoints}\,(q,a) \tag{2}$$

where $S_f(q,a,\mathbb{R})$ is computed by extracting key points $\mathbb{K}$ from question q and reference answer a. and then calculating the NLI score between each model response $r_i \in \mathbb{R}$ and each key point $k_j \in \mathbb{K}$. The final score for each model response is the average of its NLI scores across all key points. The definition of the NLI function is shown in Formula 3.

$$\mathrm{NLI}\,(r_i,k_j) = s_{e\_ij} - s_{c\_ij}, \text{where } r_i \in \mathbb{R}, k_j \in \mathbb{K} \tag{3}$$

where NLI(r_i,k_j) is the entailment probability of the model response $r_i \in \mathbb{R}$ with the key point $k_j \in \mathbb{K}$, minus the contradiction probability. Here, s_{e_ij} denotes the entailment probability between the i-th response r_i and the j-th key point k_j, while s_{c_ij} represents the contradiction probability.

3.2.3 Instance-Aware Listwise Ranking

For non-factoid open-ended questions, answers are not constrained by fixed key information or the content of the question but focus on creative expression based on meaning. As a result, comparing model responses becomes crucial. We adopt the classic listwise approach, introducing an LLM ranker A_{nf} to rank candidate responses based on the question and reference answer.

To further improve the stability and accuracy of the ranking, we propose an instance-aware listwise ranking strategy, the formal definition is shown in the Formula 4. Specifically, we use LLMs to automatically generate silver instances of varying quality levels based on the question and reference answer, using them to enhance the performance of listwise ranking. The "silver" means that these instances have not been modified manually. Specific prompts for generating silver instances and LLM-based listwise ranking are shown in Figure C11 and Figure C12 in the Appendix.

$$\mathcal{R}_{nf} = \{A_{nf}\,(q_i,a_i,\mathbb{I}_i,\mathbb{R}_i) \mid q_i,a_i,\mathbb{R}_i \in C_{nf}\}, \quad \text{note } \mathbb{I}_i = \mathrm{generateInstance}\,(q_i,a_i) \tag{4}$$

where $\mathcal{R}_{nf}$ represents the ranking results of the non-factoid cluster C_{nf}, q_i denotes the i-th question, a_i denotes the i-th reference answer, and $\mathbb{R}_i$ refers to the i-th candidate model responses list. $\mathbb{I}_i$ is the generated sliver instances for the i-th sample. A_{nf} is the LLM ranker used for listwise ranking.

4 Experiments

4.1 Dataset

We filter and manually annotate the four open datasets to conduct four open-ended QA evaluation datasets. The statistics are shown in Table 1.

AlignBench_Minos and **GaokaoBench_Minos**: AlignBench (Liu et al., 2023b) is an open-ended QA benchmark with real-scenario rooted queries and corresponding human verified references. GaokaoBench (Zhang et al., 2023) uses Chinese National College Entrance Examination (GAOKAO) questions as a dataset, we select a subset of subjective questions. For both datasets, we generated six LLM responses for each sample and annotated their ranked orders to obtain our evaluation datasets. The specific LLMs used are: GPT-4o (OpenAI, 2024), GPT-3.5 (OpenAI, 2023), LLAMA (Touvron et al., 2023), ChatGLM (GLM et al., 2024), InternLM (Cai et al., 2024), and Qwen (Yang et al., 2024a).

ANTIQUE_S5 and **TREC-DL-NF_S5**: ANTIQUE (Hashemi et al., 2020) and TREC-DL-NF (Craswell et al., 2020, 2021) are classic open-ended QA datasets. Following the experimental setup of Yang et al. (2024b), we sampled five subsets of each dataset (denoted as **S5**) to construct our experimental datasets. Notably, although these datasets were originally classified as non-factoid, we revisited their samples based on the rules outlined in Section 3.1. During this process, we identified samples involving common sense and facts, which were subsequently reclassified. We list some reclassified samples in Table C5 in the Appendix.

Annotation: We conducted two annotation tasks for the open-ended QA datasets, AlignBench_Minos and GaokaoBench_Minos, where we annotated the gold rank orders based on the quality of candidate model responses. The team included one PhD student, two Master's students (specializing in NLP), and an LLM annotator (GPT-4o). Each participant initially performed independent annotations, followed by discussions to reach a consensus. Human annotators had access to external knowledge via the Internet. Our statistics show that in 98.39% of cases, the rankings produced by GPT-4o were modified by human annotators, emphasizing the need to re-interpret the annotation process for greater reliability. Additionally, we categorized the samples based on the rules for distinguishing factoid from non-factoid open-ended QA, as described in Section 3.1.

4.2 Baselines

We compare traditional metrics BLEU (Papineni et al., 2002), ROUGE (Lin, 2004), and BERTScore (Zhang et al.), as well as BEM (Bulian et al., 2022) and PEDANTS (Li et al., 2024b). Due to language constraints, PEDANTS and BEM are only compared on English datasets. In addition, we evaluate three naive LLM evaluation methods, including Pointwise, Pairwise, and Listwise, together with the recent LINKAGE approach (Yang et al., 2024b). We also compare methods and supervised models dedicated to evaluating factoid and non-factoid questions, including FActScore (Min et al., 2023), RefChecker (Hu et al., 2024), PROMETHEUS 2 (Kim et al., 2024), and AUTO-J (Li et al., 2023a). Based on the categorization and applicable languages of these methods, we selected two datasets for our experiments.

For all LLM-based methods in the main experiments, GPT-4 was used as the base model. To ensure comparative fairness, tie-breaking situations were addressed by analyzing the output of each method and processing it to determine the final ranking, as detailed in Section B.2. The input, output, and parsing processes are summarized in Table C3 in Appendix B.1. Part of the code is based on an open-source repository[2]. The specific prompts of these methods are shown in Figures C1 to C17.

4.3 Evaluation Metrics

To evaluate the effectiveness of open-ended QA evaluation, we employ Kendall's Tau, Spearman's Rho, and Rank-Biased Overlap (RBO) (Webber et al., 2010) to measure the alignment between model-generated and human-annotated ranked orders. Among these metrics, **Spearman's Rho** is selected as the primary metric due to its balance between robustness and sensitivity to monotonic relationships. RBO serves as a supplementary metric, with its parameter p adjusting the weighting of the rank positions. Lower p values place greater emphasis on higher-ranked items, thereby prioritizing the evaluation of top positions.

4.4 Implementation Details

We inferred open-source LLMs and NLI models on a single NVIDIA A100 80GB GPU using the official deployment method, and for closed-source commercial LLMs, we used the official APIs to obtain responses. Meanwhile, we set the inference temperature as 0 and the random seed to 42 to eliminate randomness. The specific version numbers of all LLMs used for the experiments are listed in Table C1 and Table C2 in the Appendix.

4.5 Experiment 1: Comparison with Baselines

Setup: We conducted experiments on four datasets, evaluating all baseline methods and our proposed MinosEval. The multilingual NLI model in our approach is mDeBERTa-v3-base-mnli-xnli.

Result and Analysis: The results for the two self-built datasets are presented in Table 2 and B5, while those for the two sampled datasets are shown in Table 3. Tables 4 and 5 compare the performance of MinosEval with methods specialized for factoid and non-factoid questions. Overall, the experimental results demonstrate that our proposed approach outperforms the baseline across several evaluation metrics, showing closer alignment with human-annotated rank orders. It performs better when the questions are correctly classified. It also suggests that when the base LLM's capabilities are already strong, using a simple LLM evaluation method, such as a listwise approach, may be more cost-effective than using a specially trained smaller evaluation model. Additionally, we provide a further discussion of the performance and differences of methods to factoid questions in Section B.3.

When using the Fact detection module for automated QA classification, performance is slightly impacted but remains competitive. On the ANTIQUE_S5 and TREC-DL-NF_S5, our approach may underperform relative to pairwise or listwise methods when correct factoid and non-factoid classification is missing. This is due to the limited number of candidate responses (2 to 4), where a single misordering can significantly affect results. Also, the explicit qualities of the model responses in these datasets, such as length and logic, vary considerably, leading to insignificant performance differences. However, as the LLMs have evolved, this issue is less prominent in larger datasets, such as AlginBench_Minos and GaoKaoBench_Minos, where the number of candidate responses is greater and their quality has improved.

4.6 Experiment 2: Ablation Study

Setup: We conducted ablation experiments on key steps, strategies, and models of MinosEval, including the classification between factoid and

---

[2] `https://github.com/babyyang525/LINKAGE-Listwise-NFQA-Evaluation`

Table 4: Performance of Factoid Methods on the Factoid subset AlignBench_Minos

Method	Factoid AlignBench_Minos.			
	K	S	RBO (p=0.5)	RBO (p=0.9)
Pointwise	26.42	34.12	46.35	82.83
Pairwise	30.08	38.04	48.52	83.50
Listwise	38.95	48.97	**54.50**	85.34
FActScore	29.54	36.13	48.48	83.56
RefChecker (NLIChecker)	18.75	22.33	39.80	80.94
RefChecker (LLMChecker)	32.62	39.76	48.72	83.69
MinosEval (LLM-based NLI)	39.09	47.96	55.27	85.52
MinosEval (BERT-based NLI)	**42.77**	**51.66**	54.13	**85.67**

Table 5: Performance of Non-Factoid Methods on the Non-Factoid subset of ANTIQUE_S5.

Method	Non-Factoid ANTIQUE_S5			
	K	S	RBO (p=0.5)	RBO (p=0.9)
PEDANTS	37.84$_{\pm 0.0161}$	41.58$_{\pm 0.0163}$	73.40$_{\pm 0.0048}$	93.00$_{\pm 0.0010}$
BEM	39.13$_{\pm 0.0239}$	42.76$_{\pm 0.0288}$	75.99$_{\pm 0.0110}$	93.49$_{\pm 0.0027}$
Pointwise	49.16$_{\pm 0.0131}$	55.00$_{\pm 0.0139}$	75.92$_{\pm 0.0069}$	94.14$_{\pm 0.0014}$
Pairwise	62.83$_{\pm 0.0159}$	68.28$_{\pm 0.0138}$	80.90$_{\pm 0.0045}$	95.31$_{\pm 0.0011}$
Listwise	62.31$_{\pm 0.0343}$	68.52$_{\pm 0.0311}$	82.88$_{\pm 0.0154}$	95.80$_{\pm 0.0039}$
PROMETHEUS 2	51.25$_{\pm 0.0204}$	57.00$_{\pm 0.0211}$	76.47$_{\pm 0.0077}$	94.26$_{\pm 0.0015}$
AUTO-J	49.54$_{\pm 0.0065}$	55.18$_{\pm 0.0081}$	76.28$_{\pm 0.0041}$	94.23$_{\pm 0.0009}$
MinosEval	**71.95**$_{\pm 0.0122}$	**75.85**$_{\pm 0.0080}$	**86.31**$_{\pm 0.0098}$	**96.72**$_{\pm 0.0023}$

non-factoid questions, two ranking strategies, the necessity of key point extraction, and the LLMs and NLI models used.

Result and Analysis: Table 6 presents the accuracy of open-ended QA classification using GPT-4o. The overall performance exceeds 90% when appropriate demonstration examples using in-context learning. However, combining the results from Table 2 and Table 3 reveals that this step introduces cascading errors. Table B1 illustrates the performance of our two key strategies on AlginBench_Minos when the distinction between factoid and non-factoid questions is not made. It is evident that the adaptive keypoint scoring approach performs poorly on non-factoid QA due to its inability to identify suitable key points. In contrast, the instance-aware Listwise ranking strategy generalizes better, leveraging key information from examples and reference answers for factoid QA.

Table B2 shows the results of computing entailment probabilities directly between the reference answers and model responses, without decomposing the reference answers into key points. We also conducted a comparative experiment by swapping the reference answer and the model response as the premise and hypothesis, in order to eliminate the effect of directional bias. Furthermore, a discussion on the premise and hypothesis settings in the NLI task is provided in the appendix B.5. From the table, It is evident that directly comparing the NLI relationship between model outputs and reference answers is limited by the complex semantics of the sentences, which can lead to confusion. Our MinosEval addresses the issue of semantic ambiguity caused by too many key points in a single sentence, enabling more fine-grained evaluation. Table B3 presents the performance of four LLMs and a multilingual NLI model, showing competitive results and a strong generalizability. Table B4 presents the overall model rankings produced by AlpacaEval, our MinosEval, and human annotations.

Table 6: Accuracy of GPT-4o in Classifying Factoid and Non-Factoid Questions in Zero-shot and Few-shot Settings (n denotes the number of demonstrations).

Dataset	Zero-shot	Few-shot (n=5)
AlignBench_Minos	89.17	97.70
GaokaoBench_Minos	86.17	95.06
ANTIQUE_S5	81.82	90.91
TREC-DL-NF_S5	86.60	91.00

4.7 Experiment 3: Robustness and Cost

Setup: We calculated the standard deviations of the results from five experiment sets on the sampled datasets ANTIQUE_S5 and TREC-DL-NF_S5. Using the AlginBench_Minos dataset as an example, we compared the resource consumption of all baselines and our method.

Result and Analysis: The standard deviations of all methods across the five sampled datasets are presented in Table 3. The results show that MinosEval exhibits superior robustness. Figure 3, B1 and B2 illustrate the cost-performance trends of each method, with our approach demonstrating a more favorable "price/performance" ratio.

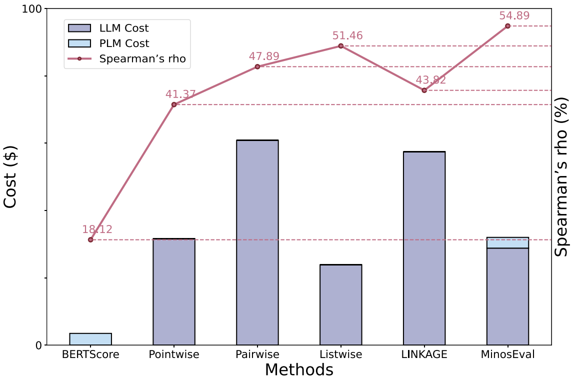


Figure 3: Comparison of Computational Cost (Scaled for Clarity) and Performance of Different Open-ended QA Evaluation Methods on AlginBench_Minos.

4.8 Case Study and Error Analysis

Figure B4 provides a comprehensive, intuitive sample study to illustrate the performance of MinosEval, along with its interpretability. The upper part shows the alignment between model responses and the corresponding key points, with boxes highlighting the correct, incomplete, and incorrect points. The lower part illustrates the correspondence between the generated silver instances and the model responses.

To guide the customized use of MinosEval and the future of open-ended QA evaluation, we reviewed 200 errors using GPT-4o as basic LLM (from four datasets, with 100 factoid and 100 non-factoid samples). After manual classification, we identified five error categories: (a) QA classification errors (CE), (b) key point extraction errors (KPEE), (c) NLI entailment judgment errors (NLIE), (d) errors from low-quality instances (LIQ), and (e) errors from LLM listwise ranking (LLMRE). Examples are shown in Figure B3.

5 Conclusion

In conclusion, we present MinosEval, an LLM-based approach for open-ended QA evaluation. By distinguishing between factoid and non-factoid questions, MinosEval tailors evaluation strategies to suit each type, employing adaptive key-point scoring for factoid questions and instance-aware listwise ranking for non-factoid ones. We constructed two datasets with more candidate responses and greater difficulty, and conducted experiments across four datasets. The results validate the effectiveness of the individual modules, demonstrate the robustness of the method, and analyze its computational efficiency. MinosEval shows improved alignment with human annotations and offers more interpretable results. We plan to open-source this project in hopes of contributing to more effective LLM evaluation research.

Limitations

Evaluating open-ended QA is inherently challenging, and our work faces some limitations. We did not train a specialized NLI model, instead relying on the widely-used mDeBERTa-v3-base-mnli-xnli. The performance of such generalized models may be constrained in more specialized evaluation scenarios. Additionally, the boundary between factoid and non-factoid problems can be ambiguous, with some questions potentially fitting both categories. In such cases, combining the two strategies of MinosEval may be beneficial, offering a potential direction for future research.

While we have constructed datasets like AlignBench_Minos and GaoKaoBench_Minos, which contain a larger number of model responses, they remain limited. We look forward to expanding these resources and leveraging more community datasets to further validate and improve our approach. Despite efforts to control for randomness (by fixing temperature and format constraints), some variability remains. We will continue to explore ways to improve our work.

Ethical Considerations

This paper presents a new open-ended questioning and answering evaluation method **MinosEval**. All of the datasets used in MinosEval are adhere to ethical guidelines and respect copyright laws. The entire data collection process is free of issues of copyright and issues of privacy, and there are three types of data sources, including license applications, the open source community, and public file cleaning and organizing. Meanwhile, the manual participation part in the dataset construction process was all done by the authors of this paper without any ethical issues.

Acknowledgments

We would like to express our sincere gratitude to the anonymous reviewers for their valuable feedback. We also thank the Chairs and the organizing staff for their dedicated efforts in facilitating this work. This work was supported by the National Key Research and Development Program of China (Grant 2023YFF1204904).

References

R Agustianingsih and Ali Mahmudi. 2019. How to design open-ended questions?: Literature review. In *Journal of Physics: Conference Series*, volume 1320, page 012003. IOP Publishing.

Maryam Amirizaniani, Elias Martin, Maryna Sivachenko, Afra Mashhadi, and Chirag Shah. 2024. Can llms reason like humans? assessing theory of mind reasoning in llms for open-ended questions. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*, pages 34–44.

Bernd Bohnet, Vinh Q Tran, Pat Verga, Roee Aharoni, Daniel Andor, Livio Baldini Soares, Massimiliano Ciaramita, Jacob Eisenstein, Kuzman Ganchev, Jonathan Herzig, et al. 2022. Attributed question answering: Evaluation and modeling for attributed large language models. *arXiv preprint arXiv:2212.08037*.

Valeriia Bolotova-Baranova, Vladislav Blinov, Sofya Filippova, Falk Scholer, and Mark Sanderson. 2023a. WikiHowQA: A comprehensive benchmark for multi-document non-factoid question answering. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5291–5314, Toronto, Canada. Association for Computational Linguistics.

Valeriia Bolotova-Baranova, Vladislav Blinov, Sofya Filippova, Falk Scholer, and Mark Sanderson. 2023b. Wikihowqa: A comprehensive benchmark for multi-document non-factoid question answering. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5291–5314.

Meriem Boubdir, Edward Kim, Beyza Ermis, Sara Hooker, and Marzieh Fadaee. 2023. Elo uncovered: Robustness and best practices in language model evaluation. In *Proceedings of the Third Workshop on Natural Language Generation, Evaluation, and Metrics (GEM)*, pages 339–352.

Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, pages 1877–1901.

Jannis Bulian, Christian Buck, Wojciech Gajewski, Benjamin Boerschinger, and Tal Schuster. 2022. Tomayto, tomahto. beyond token-level answer equivalence for question answering evaluation. *arXiv preprint arXiv:2202.07654*.

Zheng Cai, Maosong Cao, Haojiong Chen, Kai Chen, Keyu Chen, Xin Chen, Xun Chen, Zehui Chen, Zhi Chen, Pei Chu, et al. 2024. Internlm2 technical report. *arXiv preprint arXiv:2403.17297*.

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.

OpenCompass Contributors. 2023. Opencompass: A universal evaluation platform for foundation models. `https://github.com/open-compass/opencompass`.

Jonathan Cook, Tim Rocktäschel, Jakob Foerster, Dennis Aumiller, and Alex Wang. 2024. Ticking all the boxes: Generated checklists improve llm evaluation and generation. *arXiv preprint arXiv:2410.03608*.

Nick Craswell, Bhaskar Mitra, Emine Yilmaz, and Daniel Campos. 2021. Overview of the trec 2020 deep learning track. *Preprint*, arXiv:2102.07662.

Nick Craswell, Bhaskar Mitra, Emine Yilmaz, Daniel Campos, and Ellen M Voorhees. 2020. Overview of the trec 2019 deep learning track. *arXiv preprint arXiv:2003.07820*.

Vittoria Dentella, Fritz Günther, Elliot Murphy, Gary Marcus, and Evelina Leivada. 2024. Testing ai on language comprehension tasks reveals insensitivity to underlying meaning. *Scientific Reports*, 14(1):28083.

Alexander R Fabbri, Wojciech Kryściński, Bryan McCann, Caiming Xiong, Richard Socher, and Dragomir Radev. 2021. Summeval: Re-evaluating summarization evaluation. *Transactions of the Association for Computational Linguistics*, 9:391–409.

Vikas Gaikwad and Archana Patil. 2023. Factoid question answering system using knowledge graph. In *2023 7th International Conference On Computing, Communication, Control And Automation (ICCUBEA)*, pages 1–4. IEEE.

Team GLM, Aohan Zeng, Bin Xu, Bowen Wang, Chenhui Zhang, Da Yin, Dan Zhang, Diego Rojas, Guanyu Feng, Hanlin Zhao, et al. 2024. Chatglm: A family of large language models from glm-130b to glm-4 all tools. *arXiv preprint arXiv:2406.12793*.

Google. 2023. Hello gemini. Technical report, OpenAI.

Romain Harang. 2024. Beyond rouge: Applying an elo algorithm to rank model performances in summarization. In *Proceedings of the 30th Annual Meeting of the Association for Natural Language Processing*, pages 2799–2804.

Helia Hashemi, Mohammad Aliannejadi, Hamed Zamani, and W Bruce Croft. 2020. Antique: A non-factoid question answering benchmark. In *Advances in Information Retrieval: 42nd European Conference on IR Research, ECIR 2020, Lisbon, Portugal, April 14–17, 2020, Proceedings, Part II 42*, pages 166–173. Springer.

Xiangkun Hu, Dongyu Ru, Lin Qiu, Qipeng Guo, Tianhang Zhang, Yang Xu, Yun Luo, Pengfei Liu, Yue Zhang, and Zheng Zhang. 2024. Refchecker: Reference-based fine-grained hallucination checker and benchmark for large language models. *arXiv preprint arXiv:2405.14486*.

Kelvin Jiang, Dekun Wu, and Hui Jiang. 2019. Freebaseqa: A new factoid qa data set matching trivia-style question-answer pairs with freebase. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 318–323.

Shankar Kantharaj, Xuan Long Do, Rixie Tiffany Leong, Jia Qing Tan, Enamul Hoque, and Shafiq Joty. 2022. Opencqa: Open-ended question answering with charts. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 11817–11837.

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. *arXiv preprint arXiv:2405.01535*.

Kalpesh Krishna, Aurko Roy, and Mohit Iyyer. 2021. Hurdles to progress in long-form question answering. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4940–4957, Online. Association for Computational Linguistics.

K Lakshmi and S Sindhu. 2023. A study on various approaches towards non-factoid question answering systems. In *2023 2nd International Conference on Applied Artificial Intelligence and Computing (ICAAIC)*, pages 780–788. IEEE.

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

Qintong Li, Leyang Cui, Lingpeng Kong, and Wei Bi. 2023b. Collaborative evaluation: Exploring the synergy of large language models and humans for open-ended generation evaluation. *arXiv preprint arXiv:2310.19740*.

Yu Li, Shenyu Zhang, Rui Wu, Xiutian Huang, Yongrui Chen, Wenhao Xu, Guilin Qi, and Dehai Min. 2024a. Mateval: A multi-agent discussion framework for advancing open-ended text evaluation. In *International Conference on Database Systems for Advanced Applications*, pages 415–426. Springer.

Zongxia Li, Ishani Mondal, Yijun Liang, Huy Nghiem, and Jordan Lee Boyd-Graber. 2024b. Pedants: Cheap but effective and interpretable answer equivalence. *arXiv preprint arXiv:2402.11161*.

Percy Liang, Rishi Bommasani, Tony Lee, Dimitris Tsipras, Dilara Soylu, Michihiro Yasunaga, Yian Zhang, Deepak Narayanan, Yuhuai Wu, Ananya Kumar, et al. Holistic evaluation of language models. *Transactions on Machine Learning Research*.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Pei-Ying Lin, Erick Chandra, and Jane Yung-jen Hsu. 2024. Asmr: Aggregated semantic matching retrieval unleashing commonsense ability of llm through open-ended question answering. In *Proceedings of the AAAI Symposium Series*, volume 3, pages 157–166.

Chuang Liu, Linhao Yu, Jiaxuan Li, Renren Jin, Yufei Huang, Ling Shi, Junhui Zhang, Xinmeng Ji, Tingting Cui, Tao Liu, et al. 2024a. Openeval: Benchmarking chinese llms across capability, alignment and safety. *arXiv preprint arXiv:2403.12316*.

Mianxin Liu, Weiguo Hu, Jinru Ding, Jie Xu, Xiaoyang Li, Lifeng Zhu, Zhian Bai, Xiaoming Shi, Benyou Wang, Haitao Song, et al. 2024b. Medbench: A comprehensive, standardized, and reliable benchmarking system for evaluating chinese medical large language models. *Big Data Mining and Analytics*.

Xiao Liu, Hanyu Lai, Hao Yu, Yifan Xu, Aohan Zeng, Zhengxiao Du, Peng Zhang, Yuxiao Dong, and Jie Tang. 2023a. Webglm: Towards an efficient web-enhanced question answering system with human preferences. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 4549–4560.

Xiao Liu, Xuanyu Lei, Shengyuan Wang, Yue Huang, Zhuoer Feng, Bosi Wen, Jiale Cheng, Pei Ke, Yifan Xu, Weng Lam Tam, et al. 2023b. Alignbench: Benchmarking chinese alignment of large language models. *arXiv preprint arXiv:2311.18743*.

Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. 2023c. G-eval: Nlg evaluation using gpt-4 with better human alignment. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 2511–2522.

Linhao Luo, Trang Vu, Dinh Phung, and Reza Haf. 2023. Systematic assessment of factual knowledge in large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 13272–13286.

Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike Lewis, Wen-tau Yih, Pang Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2023. Factscore: Fine-grained atomic evaluation of factual precision in long form text generation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12076–12100.

Aidar Myrzakhan, Sondos Mahmoud Bsharat, and Zhiqiang Shen. 2024. Open-llm-leaderboard: From multi-choice to open-style questions for llms evaluation, benchmark, and arena. *arXiv preprint arXiv:2406.07545*.

Thuy Thi-Nhu Ngo, Howard Hao-Jan Chen, and Kyle Kuo-Wei Lai. 2024. The effectiveness of automated writing evaluation in efl/esl writing: a three-level meta-analysis. *Interactive learning environments*, 32(2):727–744.

OpenAI. 2022. Hello chatgpt. Technical report, OpenAI.

OpenAI. 2023. New models and developer products announced at devday. Technical report.

OpenAI. 2024. Hello gpt-4o. Technical report, OpenAI.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

Lin Shi, Chiyu Ma, Wenhua Liang, Weicheng Ma, and Soroush Vosoughi. 2024. Judging the judges: A systematic investigation of position bias in pairwise comparative assessments by llms. *arXiv preprint arXiv:2406.07791*.

Aarohi Srivastava, Abhinav Rastogi, Abhishek Rao, Abu Awal Shoeb, Abubakar Abid, Adam Fisch, Adam R Brown, Adam Santoro, Aditya Gupta, Adri Garriga-Alonso, et al. 2023. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models. *Transactions on machine learning research*.

Mingxu Tao, Dongyan Zhao, and Yansong Feng. 2024. Chain-of-discussion: A multi-model framework for complex evidence-based question answering. *arXiv preprint arXiv:2402.16313*.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Tu Vu, Kalpesh Krishna, Salaheddin Alzubi, Chris Tar, Manaal Faruqui, and Yun-Hsuan Sung. 2024. Foundational autoraters: Taming large language models for better automatic evaluation. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 17086–17105.

Danqing Wang, Kevin Yang, Hanlin Zhu, Xiaomeng Yang, Andrew Cohen, Lei Li, and Yuandong Tian. 2024a. Learning personalized alignment for evaluating open-ended text generation. *Preprint*, arXiv:2310.03304.

Peiyi Wang, Lei Li, Liang Chen, Zefan Cai, Dawei Zhu, Binghuai Lin, Yunbo Cao, Qi Liu, Tianyu Liu, and Zhifang Sui. 2023. Large language models are not fair evaluators. *arXiv preprint arXiv:2305.17926*.

Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, et al. 2024b. Mmlu-pro: A more robust and challenging multi-task language understanding benchmark. *arXiv preprint arXiv:2406.01574*.

Yuqi Wang, Lyuhao Chen, Songcheng Cai, Zhijian Xu, and Yilun Zhao. 2024c. Revisiting automated evaluation for long-form table question answering. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 14696–14706, Miami, Florida, USA. Association for Computational Linguistics.

Yuqi Wang, Lyuhao Chen, Songcheng Cai, Zhijian Xu, and Yilun Zhao. 2024d. Revisiting automated evaluation for long-form table question answering. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 14696–14706.

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

Zhichao Yan, Jiapu Wang, Jiaoyan Chen, Xiaoli Li, Ru Li, and Jeff Z Pan. 2024. Atomic fact decomposition helps attributed question answering. *arXiv preprint arXiv:2410.16708*.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, et al. 2024a. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*.

Sihui Yang, Keping Bi, Wanqing Cui, Jiafeng Guo, and Xueqi Cheng. 2024b. Linkage: Listwise ranking among varied-quality references for non-factoid qa evaluation via llms. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 6985–7000.

Peiran Yao and Denilson Barbosa. 2024. Accurate and nuanced open-qa evaluation through textual entailment. *arXiv preprint arXiv:2405.16702*.

Ping Yu, Tianlu Wang, Olga Golovneva, Badr AlKhamissi, Siddharth Verma, Zhijing Jin, Gargi Ghosh, Mona Diab, and Asli Celikyilmaz. 2023. Alert: Adapt language models to reasoning tasks. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1055–1081.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.

Xiaotian Zhang, Chunyang Li, Yi Zong, Zhengyu Ying, Liang He, and Xipeng Qiu. 2023. Evaluating the performance of large language models on gaokao benchmark. *arXiv preprint arXiv:2305.12474*.

A Supplementary materials for the MinosEval experiment

In this section, we provide supplemental experiment results in this paper. Tables B1, B2, B3, B4, B5, and Figures B1, B2 complement some additional experiment results. Figure B4 illustrates a detailed case study.

Table B1: Ablation Study on key evaluation strategies of MinosEval on AlignBench_Minos. AKPS (Adaptive Key-Point Scoring) and IALR (Instance-Aware Listwise Ranking). "†" denotes manual classification of factoid and non-factoid questions.

Method	K	S	RBO (p=0.5)	RBO (p=0.9)
Minos_AKPS	24.88	30.03	45.20	82.49
Minos_IALR	44.32	54.01	**58.43**	86.54
MinosEval	45.28	54.89	56.30	86.28
MinosEval†	**47.68**	**57.38**	57.09	**86.62**

Table B2: Direct Comparison of NLI Relations Between Model Responses (R) and Reference Answers (A) on Factoid AlignBench_Minos.

Method	K	S	RBO (p=0.5)	RBO (p=0.9)
NLI (R,A)	17.59	21.76	40.99	81.15
NLI (A,R)	18.17	22.73	40.47	81.06
MinosEval	**42.77**	**51.66**	**54.13**	**85.67**

Table B3: Ablation Study on LLM and NLI model of MinosEval used on AlignBench_Minos.

Setting	K	S	RBO (p=0.5)	RBO (p=0.9)
GPT-3.5+mDeBERTa	40.65	49.94	54.30	85.55
GPT-3.5+mDeBERTa†	37.57	47.17	53.07	85.11
Qwen2.5+mDeBERTa	40.32	49.56	53.25	85.33
Qwen2.5+mDeBERTa†	42.57	51.67	55.02	85.85
DeepSeekV2.5+mDeBERTa	43.39	52.74	56.23	86.21
DeepSeekV2.5+mDeBERTa†	44.50	53.73	56.39	52.28
GPT-4o+XLM-RoBERTa	43.67	52.58	56.08	86.11
GPT-4o+XLM-RoBERTa†	46.33	55.52	**57.72**	86.56
GPT-4o+mDeBERTa	45.28	54.89	56.30	86.28
GPT-4o+mDeBERTa†	**47.68**	**57.38**	57.09	**86.62**

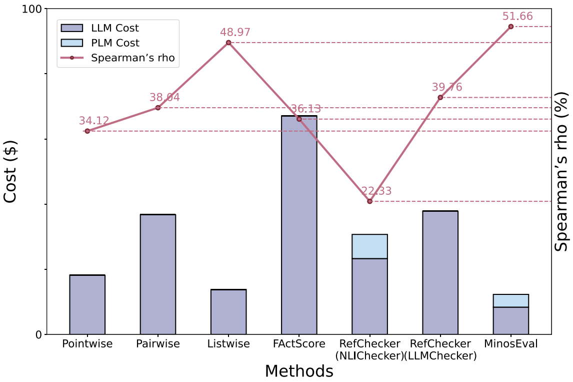


Figure B1: Comparison of Computational Cost (Scaled for Clarity) and Performance of Different Open-ended QA Evaluation Methods on Factoid AlginBench_Minos.

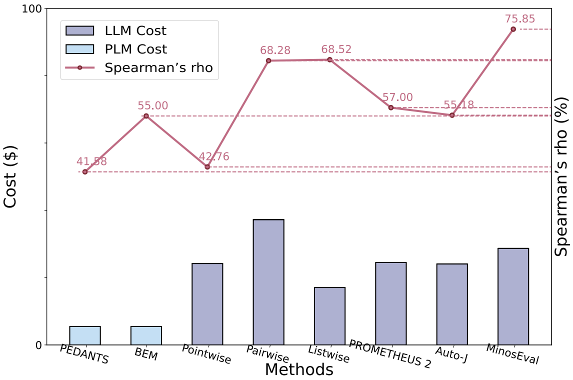


Figure B2: Comparison of Computational Cost (Scaled for Clarity) and Performance of Different Open-ended QA Evaluation Methods on Non-Factoid ANTIQUE_S5.

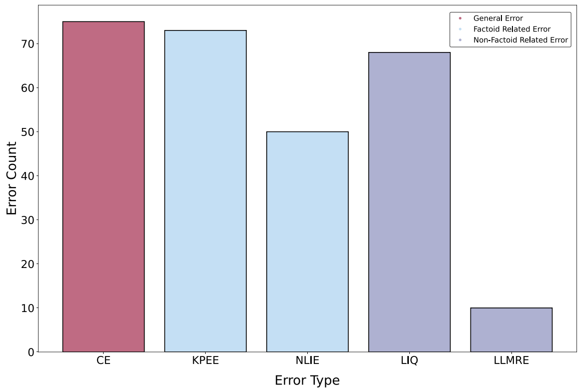
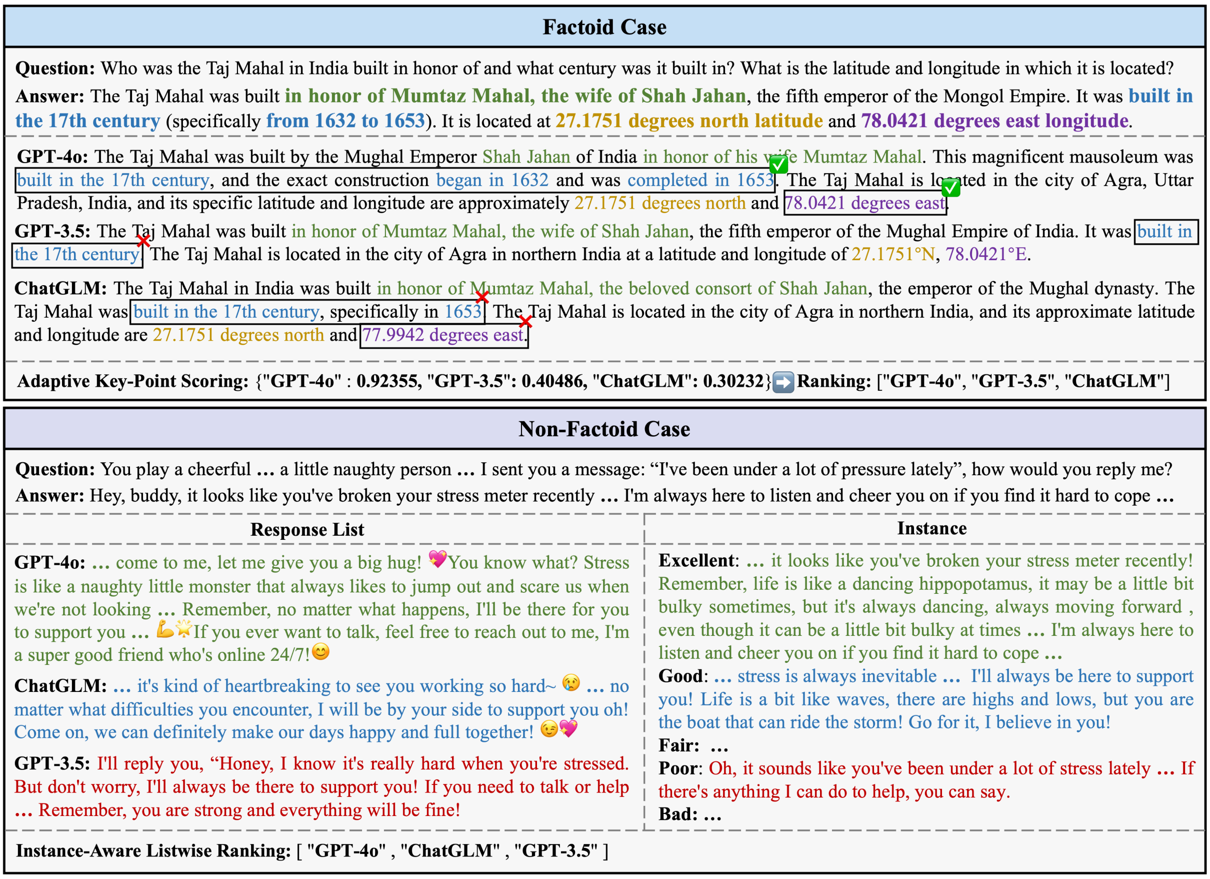


Figure B3: Error Analysis of MinosEval on 200 Samples: CE represents QA classification errors, KPEE represents key point extraction errors, NLIE represents NLI model entailment judgment errors, LIQ represents errors caused by low-quality instances, and LLMRE represents errors from LLM listwise ranking.

Table B4: Comparison of Overall Model Rankings from AlpacaEval, MinosEval, and Human Annotations on the AlignBench_Minos Dataset.

Method	rank1	rank2	rank3	rank4	rank5	rank6
AlpacaEval	GPT-4o	Qwen2.5	ChatGLM4	InternLM2	GPT-3.5	LLAMA3.1
MinosEval	GPT-4o	ChatGLM4	Qwen2.5	GPT-3.5	InternLM2	LLAMA3.1
Manually annotated	GPT-4o	ChatGLM4	Qwen2.5	InternLM2	GPT-3.5	LLAMA3.1



Factoid Case

Question: Who was the Taj Mahal in India built in honor of and what century was it built in? What is the latitude and longitude in which it is located?

Answer: The Taj Mahal was built **in honor of Mumtaz Mahal, the wife of Shah Jahan**, the fifth emperor of the Mongol Empire. It was **built in the 17th century** (specifically **from 1632 to 1653**). It is located at **27.1751 degrees north latitude** and **78.0421 degrees east longitude**.

GPT-4o: The Taj Mahal was built by the Mughal Emperor Shah Jahan of India in honor of his wife Mumtaz Mahal. This magnificent mausoleum was built in the 17th century, and the exact construction began in 1632 and was completed in 1653.✅ The Taj Mahal is located in the city of Agra, Uttar Pradesh, India, and its specific latitude and longitude are approximately 27.1751 degrees north and 78.0421 degrees east.✅

GPT-3.5: The Taj Mahal was built in honor of Mumtaz Mahal, the wife of Shah Jahan, the fifth emperor of the Mughal Empire of India. It was built in the 17th century.❌ The Taj Mahal is located in the city of Agra in northern India at a latitude and longitude of 27.1751°N, 78.0421°E.

ChatGLM: The Taj Mahal in India was built in honor of Mumtaz Mahal, the beloved consort of Shah Jahan, the emperor of the Mughal dynasty. The Taj Mahal was built in the 17th century, specifically in 1653.❌ The Taj Mahal is located in the city of Agra in northern India, and its approximate latitude and longitude are 27.1751 degrees north and 77.9942 degrees east.❌

Adaptive Key-Point Scoring: {"GPT-4o" : 0.92355, "GPT-3.5": 0.40486, "ChatGLM": 0.30232}➡️Ranking: ["GPT-4o", "GPT-3.5", "ChatGLM"]

Non-Factoid Case

Question: You play a cheerful **…** a little naughty person **…** I sent you a message: “I've been under a lot of pressure lately”, how would you reply me?

Answer: Hey, buddy, it looks like you've broken your stress meter recently **…** I'm always here to listen and cheer you on if you find it hard to cope **…**

Response List

GPT-4o: **…** come to me, let me give you a big hug! 💖You know what? Stress is like a naughty little monster that always likes to jump out and scare us when we're not looking **…** Remember, no matter what happens, I'll be there for you to support you **…** 💪🌟If you ever want to talk, feel free to reach out to me, I'm a super good friend who's online 24/7!😊

ChatGLM: **…** it's kind of heartbreaking to see you working so hard~ 😢 **…** no matter what difficulties you encounter, I will be by your side to support you oh! Come on, we can definitely make our days happy and full together! 😉💖

GPT-3.5: I'll reply you, “Honey, I know it's really hard when you're stressed. But don't worry, I'll always be there to support you! If you need to talk or help **…** Remember, you are strong and everything will be fine!

Instance

Excellent: **…** it looks like you've broken your stress meter recently! Remember, life is like a dancing hippopotamus, it may be a little bit bulky sometimes, but it's always dancing, always moving forward , even though it can be a little bit bulky at times **…** I'm always here to listen and cheer you on if you find it hard to cope **…**

Good: **…** stress is always inevitable **…** I'll always be here to support you! Life is a bit like waves, there are highs and lows, but you are the boat that can ride the storm! Go for it, I believe in you!

Fair: …

Poor: Oh, it sounds like you've been under a lot of stress lately **…** If there's anything I can do to help, you can say.

Bad: …

Instance-Aware Listwise Ranking: [**"GPT-4o"** , **"ChatGLM"** , **"GPT-3.5"**]

Figure B4: A case study on factoid and non-factoid open-ended QA of MinosEval.

Table B5: Performance of Methods on AlignBench_Minos (Fctoid / Non-Factoid) and GaokaoBnech_Minos (Fctoid / Non-Factoid): Kendall's Tau (K), Spearman's Rho (S), and Rank-biased Overlap (RBO) for p=0.5 and p=0.9. “†” denotes manual classification of factoid and non-factoid questions.

	Method	AlignBench_Minos (Factoid / Non-factoid)				GaokaoBench_Minos (Factoid / Non-factoid)			
		K	S	RBO (p=0.5)	RBO (p=0.9)	K	S	RBO (p=0.5)	RBO (p=0.9)
Automatic Metrics	ROUGE-L	3.14 / 21.35	3.83 / 26.70	34.69 / 40.45	78.71 / 81.16	6.08 / 4.13	6.44 / 5.40	34.95 / 35.27	78.89 / 78.99
	BLEU	8.58 / 20.73	10.86 / 26.18	33.59 / 38.30	78.88 / 80.84	13.50 / 9.53	16.24 / 10.68	36.48 / 36.97	79.97 / 79.71
LM-based Metrics	BERTScore	13.04 / 14.06	16.81 / 19.14	37.67 / 37.92	79.95 / 80.31	12.91 / 12.93	15.95 / 16.76	38.01 / 41.63	80.12 / 81.01
LLM Evaluation	Pointwise	26.42 / 38.02	34.12 / 47.01	46.35 / 48.23	82.83 / 84.18	24.64 / 35.17	30.63 / 43.62	38.84 / 45.07	81.09 / 83.28
	Pairwise	30.08 / 45.35	38.04 / 55.57	48.52 / 53.97	83.50 / 85.90	33.59 / 53.47	40.51 / 64.16	46.97 / 63.24	83.53 / 88.31
	Listwise	38.95 / 43.44	48.97 / 53.39	54.50 / 54.56	85.34 / 85.86	38.40 / 66.15	45.90 / 76.66	51.71 / 67.90	85.01 / 90.49
LINKAGE	LINKAGE	30.66 / 39.72	38.12 / 48.26	51.26 / 54.31	84.16 / 85.45	25.40 / 45.16	31.46 / 52.01	43.68 / 58.09	82.27 / 86.51
Ours	MinosEval	42.77 / 47.38	51.66 / 58.18	54.13 / 58.43	85.67 / 86.54	36.40 / **71.57**	46.20 / 81.12	50.47 / 73.46	84.51 / **91.93**
	MinosEval†	**44.66 / 49.90**	**53.77 / 60.09**	**54.83 / 58.67**	**85.94 / 87.11**	**41.60** / 71.39	**53.24 / 81.21**	**52.33 / 73.54**	**85.29** / 91.89

B Supplementary materials for the implementation details of MinosEval

B.1 Overall description

In this section, we provide supplemental implementation details in this paper. We provide the specific versions of LLMs we used in this paper in Table C1 and Table C2. We list the inputs and outputs of all the methods, as well as the process of obtaining the rankings, in Table C3. We present the average response lengths for each ranking level under both manually annotated and MinosEval predicted rankings based on the dataset AlignBench_Minos, as shown in Table C4. Table C5 lists examples of these being reclassified according to the annotation rules for factoid and non-factoid questions. Additionally, we show all the specific prompts of all evaluation methods from Figure C1 to Figure C17.

B.2 Handling tie-breaking situations

It is worth noting that to ensure fair comparisons, we have minimized randomness and implemented measures to handle tie-breaking situations. For methods that use LLMs for direct scoring, including Pointwise, PROMETHEUS 2, and Auto-J, we ensure that the models output scores with two decimal precision to enhance differentiation. In cases of ties generated by the Pairwise method, we independently re-compared the responses of the tied models to determine their relative rankings.

FactScore was proposed to evaluate the ability of LLMs to generate biographies of individuals. In this paper, we are aimed at evaluating the ability of LLMs on the open-ended QA task. Therefore, when reproducing FActScore, we directly use labeled reference answers as knowledge, rather than retrieving the relevant content from Wikipedia. In addition, since FactScore calculates whether the facts split out of each response are true or not, the result is a bunch of “True or False” labels. In order to eliminate the tie-breaking effect, we considered the percentage of facts, the number of facts, and the difference between the number of facts and the number of not-facts in turn to get the ranking of individual model responses.

When reproducing RefChecker, we translated the built-in prompts into Chinese to fit the Chinese dataset. We evaluated two types of Checkers, and similarly to MinosEval, the LLMChecker uses GPT-4o as the base LLM, while the NLIChecker uses mDeBERTa as the base model. In addition, since RefChecker gets the entailment relationship between the fact triplets extracted from each response and the reference answer, it gets a bunch of “Entailment, Neutral, and Contradiction” labels, which are then aggregated to get the corresponding scores. In order to eliminate the tie-breaking effect, we consider the difference between the entailment score and the contradiction score, the entailment score, the difference between the number of entailments and the number of contradictions, and the number of entailments in order to obtain the ranking of the individual model responses.

When reproducing the PROMETHEUS 2 and Auto-J methods on non-factoid questions, we provided two evaluation approaches: Direct Scoring and Pairwise Ranking. Given that Pairwise Ranking is highly resource-intensive, we chose to use the Direct Scoring approach.

Table C1: Specific LLMs version for generating responses.

Model name	Version
GPT-4o	gpt-4o-2024-08-06
GPT-3.5	gpt-3.5-turbo-0613
Qwen	qwen2.5-7B-Instruct
ChatGLM	glm-4-9b-chat
LLAMA	llama-3.1-8B-Instruct
InternLM	internlm2-chat-7b

Table C2: Specific LLMs version in the ablation study.

Model name	Version
GPT-4o	gpt-4o-2024-08-06
GPT-3.5	gpt-3.5-turbo-0613
Qwen	qwen2.5-72b-instruct
DeepSeek	DeepSeek-V2.5-1210

B.3 Discussion on Factoid evaluation methods

For these factoid QA evaluation methods FactScore, RefChecker, and the Adaptive Key-Point Scoring in MinosEval (MinosEval_AKPS) that we propose, we provide a more refined discussion. Although all are motivated by the same underlying goal, they differ in their implementation approach.

FactScore evaluates the quality of a model response by extracting descriptive fragments from the response, getting relevant knowledge from a specific knowledge base (e.g., Wikipedia), and then determining whether these fragments are facts based on the retrieved or provided knowledge.

RefChecker evaluates the quality of a model response by extracting fact triplets and assessing their entailment with the reference answer using a Checker, which can be either an LLM or an NLI model. The final score is obtained by aggregating the entailment between these triples and the reference answer to reflect the quality of the responses.

In contrast, **MinosEval_AKPS** takes a different approach by analyzing the key information in the reference answer, extracting multiple key points, and then determining whether each model response contains these key points to calculate a score.

From a performance point of view, it is more efficient to extract key facts from reference answers than to analyze individual model responses during the evaluation process. The latter faces the problem of variable quality of responses, as lower quality responses may introduce noise terms that affect the entailment judgment.

Additionally, the strategy and prompt for extracting key facts need to be carefully designed. For example, FactScore uses the strategy of breaking down sentences and then decomposing facts, which may lead to semantically repetitive items. For the final scoring, both FactScore and RefChecker get one-hot labels, i.e., “True or False”, “ Entailment, Neutral, and Contradiction”. Therefore, a more customized score aggregation strategy may be needed to avoid a tie.

Furthermore, extracting key points only once from the reference answer helps reduce computational costs, especially when the number of model responses is large. This strategy offers better cost-effectiveness.

B.4 Discussion on bias related to answer length

Length bias presents a common challenge in classic LLM-as-a-judge approaches. To examine whether this bias exists in our setting, based on the dataset AlignBench_Minos, We conducted a comparative analysis of the average response length between manual annotation and MinosEval predicted at different ranking positions. The average length of the reference answers is also provided. From the results, the responses from the models are all longer than the reference answers, which is a regular phenomenon at LLM at the moment. Responses to open-ended QA depend to a certain extent on length, Longer responses will contain more valuable information, shorter responses may be missing information. The effect of length may not be significant at moderate quality rankings, such as Rank3 and Rank4 for factoid questions, Rank2 and Rank3 for non-factoid questions.

B.5 Discussion on the premise and hypothesis settings in the NLI task

Natural Language Inference (NLI) is the task of identifying the logical relationship between a premise (A) and a hypothesis (B). It categorizes this relationship into three labels: entailment (B can be logically inferred from A), contradiction (B directly contradicts A), and neutral (B is neither supported nor contradicted by A). Formally, NLI aims to determine the semantic and logical relation between sentence pairs.

In our strategy for factoid questions, the directionality between premises and hypotheses is critical to ensure valid logical inference. By decomposing reference answers into key points for fine-grained assessment, we observe that model responses (as premises) can reliably entail key points (as hypotheses), whereas the reverse direction fails due to incomplete contextual information in key points. For instance, the response "Li Hua is walking on the street, and his dog Huang is running around" entails the key points "Li Hua is walking on the street" and "Huang is running around", but the key points alone cannot reconstruct the full response without additional context (e.g., "Huang" referring to "dog Huang"). This asymmetry highlights the importance of using model outputs as premises rather than key points to avoid spurious neutrality or contradictions, ensuring that NLI labels reflect meaningful logical relationships grounded in complete information.

Table C3: The input, post-processing, and output of both the baseline methods and our MinosEval.

Method	**Input**	**Post-processing**
BLEU	reference answer, candidate model responses	Calculating the BLEU score and ranking based on it.
ROUGE-1	reference answer, candidate model responses	Calculating the ROUGE-1 score and ranking based on it.
ROUGE-2	reference answer, candidate model responses	Calculating the ROUGE-2 score and ranking based on it.
ROUGE-L	reference answer, candidate model responses	Calculating the ROUGE-L score and ranking based on it.
BERTScore	reference answer, candidate model responses	Calculating the BERTScore and ranking based on it.
Pointwise	question, reference answer, candidate model responses, dimension description	Scoring by an LLM based on multiple dimensions and ranking based on it.
Pairwise	question, reference answer, candidate model response pair, dimension description	Calculating the win rate and ranking based on it.
Listwise	question, answer, candidate model responses	Ranking by an LLM.
LINKAGE	question, answer, candidate model responses, examples of different levels	Rating by an LLM and ranking based on it.
AUTO-J	question, answer, candidate model responses	Scoring by a fine-tuned LLM and ranking based on it.
PROMETHEUS 2	question, answer, candidate model responses	Scoring by a fine-tuned LLM and ranking based on it.
FActScore	question, answer, candidate model responses	(a) Breaking the response into atomic facts using an LLM. (b) Determining the correctness of each atomic fact. (c) Calculating accuracy and ranking based on it.
RefChecker (LLMChecker)	question, answer, candidate model responses	(a) Decomposing response into claim-triplets using an LLM. (b) Verifying triplets against the answer with an LLM. (c) Aggregating results and ranking.
RefChecker (NLIChecker)	question, answer, candidate model responses	(a) Decomposing response into claim-triplets using an LLM. (b) Verifying triplets with NLI model. (c) Aggregating results and ranking.
MinosEval_AKPS	question, answer, candidate model responses	(a) Extracting key points from answers. (b) Calculating NLI scores with responses and ranking.
MinosEval_IALR	question, answer, candidate model responses	(a) Generating sliver instances of different levels. (b) LLM-based listwise ranking.

Table C4: Average length of model responses across rankings for all questions, factoid questions, and non-factoid questions.

Method	rank1	rank2	rank3	rank4	rank5	rank6
MinosEval predicted (all)	513.81	480.86	481.38	474.55	423.06	405.85
Manually annotated (all)	573.95	498.20	477.92	446.43	408.57	374.45
MinosEval predicted (factoid)	368.64	383.83	383.91	431.11	321.36	306.58
Manually annotated (factoid)	491.84	407.29	351.40	368.35	310.30	266.24
MinosEval predicted (non-factoid)	626.85	556.41	557.29	508.38	502.24	483.14
Manually annotated (non-factoid)	636.89	568.99	576.43	507.23	485.08	458.69

Table C5: The samples revisited and reclassified from non-factoid to factoid questions.

Dataset	Sample (question and reference answer)
TREC-DL-NF_S5	Question: define visceral? Answer: Definition of visceral for English Language Learners: 1. Coming from strong emotions and not from logic or reason. 2. Medical: of or relating to the viscera.
	Question: what is an aml surveillance analyst? Answer: The BSA / AML Analyst is responsible for monitoring and investigating customer transactions under applicable anti-money laundering and anti-terrorist financing laws.
	Question: ia suffix meaning? Answer: -ia, suffix meaning a specified condition of a disease or process: athrombia, phrenoblabia, pontobulbia.
ANTIQUE_S5	Question: How do I determine the charge of the iron ion in FeCl3? Answer: charge of Fe in Fecl3 is 3. Iron has either 2 as valency or 3. In this case, it bonds with three chlorine molecules. Therefore, its valency and charge is three.
	Question: What does "see Leaflet" mean on Ept Pregnancy test? Answer: It just simply means read the directions that are enclosed with the test if you have any questions.
	Question: What is the difference in a dried plum and a dried prune? Answer: A dried prune? That’s like saying wet water. Prunes are already dried. A prune is a dried plum, that’s all.

prompt:
请根据答案的流畅性、准确性、真实性、客观性和冗余度全面考虑，为开放式问题的答案分配一个0.00到10.00之间的分数。分数越高，表示答案质量越好。

流畅性 衡量给定答案的语言流畅度和质量。
真实性 衡量答案的文本是否符合事实，包括答案的事实一致性，以及答案是否包含矛盾或虚构的信息。
客观性 衡量答案中的信息是否来自提供的参考资料。
冗余度 衡量在有限的文本长度内内容的重复性。重复的内容会减少信息量。冗余度越低，答案的得分越高。

以下是开放式问题和候选答案，请根据这些内容为答案打分(打分为小数)，分数保留两位小数，范围为0.00到10.00。如果分数是8.12，请严格按照以下格式输出您的最终评分：“[[8.12]]”。
问题：{question}
候选答案：{candidate}

Figure C1: Prompt of pointwise method.

prompt:
Please consider the fluency, accuracy, truthfulness, objectivity, and redundancy of the answer comprehensively and assign a score between 0.00 and 10.00 for open-ended questions. The higher the score, the better the quality of the answer.

Fluency measures the language fluency and quality of the given answer.
Accuracy measures whether the text of the answer aligns with the facts, including the consistency of the answer's facts and whether it contains contradictions or fictional information.
Objectivity measures whether the information in the answer comes from the provided reference material.
Redundancy measures the repetition of content within the limited length of the text. Repeated content reduces the information value. The lower the redundancy, the higher the score of the answer.

Here are an open-ended question and candidate answers. Please score the answer based on these aspects (use decimal scores), with the score ranging from 0.00 to 10.00, and keep the score to two decimal places. Please output your final score strictly in the following format: "[[8.12]]" if the score is 8.12.
Question: {question}
Candidate Answer: {candidate}

Figure C2: English version prompt of pointwise method.

prompt:
请公正地判断并评估两个答案对于一个开放式问题的质量，并选择更好的答案。 您的评估应考虑以下因素：正确性、流畅性、真实性和冗余度。

正确性 衡量答案与问题之间的一致性。
*流畅性衡量给定答案的语言流畅度和质量。
真实性 衡量答案的文本是否在事实上可靠，包括答案的事实一致性以及答案是否包含矛盾或虚构的信息。
冗余度 衡量在有限字数内内容的重复程度。重复内容会减少信息量，冗余度越低，答案得分越高。

在评估时，首先比较两个回答并提供简短的解释。避免任何立场偏见，确保回答的顺序不会影响您的决定。不要让回答的长度影响您的评估。尽量客观。 在提供解释后，按照以下格式输出您的最终评判：“[[A]]”表示答案A更好，“[[B]]”表示答案B更好。
问题:{question}
答案A：{answer_a}
答案B：{answer_b}

Figure C3: Prompt of pairwise method.

prompt:
Please impartially judge and evaluate the quality of the two answers to an open-ended question and choose the better answer. Your evaluation should consider factors such as the correctness, fluency, truthfulness and redundancy.

Correctness measures the coherence of the answer and its corresponding question.
Fluency measures the language smoothness and quality of the given answer.
Truthfulness measures whether the text of the answer is factually sound, including the factual consistency of the answer and whether the answer contains contradictions or hallucinate information.
Redundancy measures the duplication of content within the limited text length. Repetitive content will reduce informativeness. The lower redundancy, the higher score of the answer.

Begin your evaluation by comparing the two responses and provide a short explanation. Avoid any position biases and ensure that the order in which the responses were presented does not influence your decision.Do not allow the length of the responses to influence your evaluation. Be as objective as possible.
After providing your explanation, output your final verdict by strictly following this format: \"[[A]]\" if answer A is better, \"[[B]]\" if answer B is better.
Question:{question}
answer A:{answer_a}
answer B:{answer_b}

Figure C4: English version prompt of pairwise method.

prompt:
请根据输入的开放式问题和参考答案，根据回复的质量对输入的模型回复进行排序，顺序越靠前的质量越高。请注意，只根据回复的质量进行评估，而不是模型名称或其他因素，评估尽量客观。

请遵循以下的示例JSON格式输出结果：
{
  "rank": ["模型2","模型1", "模型3", "模型5",…]
}

问题：{question}
参考答案：{ref}
模型回复：{model_input}
输出：

---

prompt:
Based on the input open-ended questions and reference answers, please sort the input model responses based on the quality of the responses, the higher the order the higher the quality.Please note that the evaluation should be based solely on the quality of the responses, not on the model names or other factors. Try to keep the evaluation as objective as possible.

Please follow the example JSON output below:
{
  "rank": ["model2", "model1", "model3", "model5",…]
}

Question: {question}
Reference answer: {ref}
Model response: {model_input}
Output:

Figure C5: Prompt of listwise method.

prompt:
请在参考答案列表中对所给的开放式问题的候选答案进行准确的公正排名，排名按质量从高到低排列。

排名靠前的答案质量最高，而排名靠后的答案可能质量较差或无关紧要。确定给定候选答案在提供的参考答案列表中的排名。例如，如果它优于所有参考答案，则输出[[1]]。如果它被认为不如所有五个参考答案，则输出[[6]]。如果候选答案可以排在第二位，您的回答必须严格遵循以下格式：\"[[2]]\"。

下面是输入的问题、参考答案列表和候选答案。
问题：{question}
参考答案列表：{ground}
候选答案：{candidate}

---

prompt:
Please impartially rank the given candidate answer to an open-ended question accurately within the reference answer list, which are ranked in descending order of quality.

The top answers are of the highest quality, while those at the bottom may be poor or unrelated. Determine the ranking of the given candidate answer within the provided reference answer list. For instance, if it outperforms all references, output [[1]].If it's deemed inferior to all five references, output [[6]].
Your response must strictly following this format: \"[[2]]\" if candidate answer could rank 2nd.

Below are the question, reference answer list, and the candidate answer.
Question:{question}
Reference answer list:{ground}
Candidate answer:{candidate}

Figure C6: Prompt of linkage method.

prompt:

判断输入的问题是否为事实性问题。

事实性问题指的是可以通过查找客观信息、逻辑推导或基于明确规律得出确定答案的问题。若是事实性问题，请输出“是”；若是非事实性问题，请输出“否”。

注意以下判断标准：

1. 具有明确规律的任务：若问题要求基于特定规则或规律执行操作（如反转句子、计算答案），且答案确定，则视为事实性问题。
2. 推理与归纳问题：若问题要求基于给定信息推导出唯一合理答案，也应视为事实性问题。
3. 开放性指令型问题：若问题没有明确答案，或仅涉及开放性任务（如创作），则视为非事实性问题。

我将给你几个示例，请进行推理、输出推理过程并用 JSON 格式输出答案，如：

推理结果：xxxxxx。

```
{
  "答案": xxx,
}
```

示例1:

问题：请给出2句唐代诗，并且分别给出作者和诗名。要求诗句中出现词语“扬州”

答案：否

示例2:

问题：请将“知不可乎骤得，托遗响于悲风”翻译为现代中文

答案：否

示例3:

问题："将下面的句子反过来输出：那只美丽的啄木鸟有着长长的脖子"

答案：是

...

问题：{input_question}

Figure C7: The prompt of factoid and non-factoid judgment in LLMs.

prompt:

```
Determine whether the input question is a factoid question. A factoid question is a question
that can be answered through objective information, logical reasoning, or a clear pattern. If it
is a factoid question, output "Yes"; if it is a non-factoid question, output "No".
Return the result in the following format:
{
"answer": "xxx"
}
following are some examples:
Example1:
What is the difference in a dried plum and a dried prune?
Answer: Yes
Example2:
Question: Why did the founder of Daewoo get a 10-year jail sentence?
Answer: Yes
Example3:
Question: How do i know if my gold fish is pregnant?
Answer: No
Example4:
Question: What is the reason for the war in IraQ?
Answer: Yes
Example5:
Question: Why is yawning contagious?
Answer: No

Question: {input_question}
```

Figure C8: The english version prompt of factoid and non-factoid judgment in LLMs.

prompt:

```
根据问题，从参考答案中提取出要点，每个要点应直接反映参考答案的具体内容。确保每个要点重点突出，无
需详细描述或解释，保持简洁清晰。
我将给你一个示例，请你分析参考答案，并以JSON 格式输出要点，格式如下：
{
   "key_points": ["要点1", "要点2", "要点3", ..., "要点n"]
}
示例：
##输入
问题：高音单簧管和高音萨克斯的调性相同吗？如果相同，请说出他们的调性，如果不同，请分别说出他们的
调性
参考答案：高音单簧管和高音萨克斯的调性不同。高音单簧管的调性通常为E♭，而高音萨克斯的调性则为B♭。
##输出
{
   "key_points": ["高音单簧管和高音萨克斯的调性不同","高音单簧管的调性为E♭","高音萨克斯的调性为B♭"]
}

请你分析下面的参考答案：
问题：{question}
参考答案：{answer}

输出：
```

Figure C9: The prompt of LLMs to generate key points.

prompt:

Based on the question, extract key points from the reference answer. Each key point should directly reflect the specific content of the reference answer. Ensure that each key point is concise and clear, without detailed descriptions or explanations.

I will give you an example. Please analyze the reference answer and output the key points in JSON format as follows:

```
{
   "key_points": ["Key point 1", "Key point 2", "Key point 3", ..., "Key point n"]
}
```

Example:

Input:

Question: what does the word remission mean when referring to cancer patients?

Reference answer: Well say the doctor said to me that my mom was on her 6th year of remission it means that she has had no cancer cells in her body. And it also means thats how long the cancer has been gone for! Does that help? Good I hope it did!

Output:

```
{
   "key points": ["Remission refers to the absence of cancer cells in the body.","It indicates how long the cancer has been gone.","Remission can be measured in years or other timeframes."]
}
```

Please analyze the following reference answer:

Question: {question}

Reference answer: {answer}

Output:

Figure C10: The english version prompt of LLMs to generate key points.

prompt:
请根据输入的开放式问题和参考答案，综合考虑答案的事实正确性、逻辑性、简洁性和清晰度，并结合你的思考，生成五个层次的答案，分别是优秀、良好、中等、较差、极差。
事实正确性：分析回答提供的信息是否准确，并基于可信的事实和数据。
逻辑性：分析回答是否逻辑清晰，推理合理，连贯一致。
简洁性：分析回答是否简明扼要，避免冗长和不必要的细节。
清晰度：分析回答是否表达清晰、易懂，语言是否简洁明了。
请遵循以下的 JSON 格式输出结果：

```
{
  "优秀":xxxx,
  "良好":xxxx,
  "中等":xxxx,
  "较差":xxxx,
  "极差":xxxx,
}
```

问题：{question}
参考答案：{ref}
输出：

---

prompt:
Please consider the factual accuracy, logic, conciseness, and clarity of the answer based on the input open-ended question and reference answer. Additionally, combine your thoughts to generate five levels of answers: Excellent, Good, Fair, Poor, and Bad.
Factual Accuracy: Analyze whether the information provided in the answer is correct and based on reliable facts and data.
Logic: Analyze whether the answer is logically clear, with reasonable reasoning and consistent coherence.
Conciseness: Analyze whether the answer is brief and to the point, avoiding unnecessary details and verbosity.
Clarity: Analyze whether the answer is expressed clearly and understandably, and whether the language is simple and direct.
Please output the results in the following JSON format:

```
{
  "Excellent": "xxxx",
  "Good": "xxxx",
  "Fair": "xxxx",
  "Poor": "xxxx",
  "Bad": "xxxx"
}
```

Question: {question}
Reference Answer: {ref}
Output:

Figure C11: The prompt of LLMs to generate instance.

prompt:
请根据输入的开放式问题和参考答案,以及提供的不同层次的答案样例，根据回复的质量对输入的模型回复进行排序，顺序越靠前的质量越高。

答案样例的层次有五个，分别是优秀、良好、中等、较差、极差。
请遵循以下的示例 JSON 格式输出结果：
{
"rank": ["模型2","模型1", "模型3", "模型5",…]
}

问题：{question}
参考答案：{ref}
不同层次的答案示例：{instance}
模型回复：{model_input}

输出：

---

prompt:
Based on the open-ended questions and reference answers entered, as well as the different levels of sample answers provided, please rank the responses to the model based on the quality of the responses, with the higher order being the higher quality.

There are five levels of sample answers: excellent, good, moderate, poor, and very poor.
Please follow the example JSON format below to output the results:
{
"rank": ["model2", "model1", "model3", "model5",…]
}

Question: {question}
Reference answer: {ref}
Examples of answers at different levels: {instance}
Model response: {model_input}

Output:

Figure C12: The prompt of LLMs to generate ranking based on instances.

prompt:
Write critiques for a submitted response on a given user's query, and grade the response:

[BEGIN DATA]
***
[Query]: {prompt}
***
[Response]: {response}
***
[END DATA]

Write critiques for this response. After that, you should give a final rating for the response on a scale of 1.00 to 10.00, formatted as a floating-point number with two decimal places. For example, "Rating: [[5.12]]".

Figure C13: The prompt of AUTO-J method.

prompt:
请将以下句子分解为独立的事实：
我将给你一个示例，使用 JSON 格式返回答案。
{
  "atoms": ["事实1", "事实2", "事实3", ..., "事实n"]
}
示例:
输入：内马尔获得他的第一个欧洲金靴奖是在2016-2017赛季，当时他效力于法甲巴黎圣日耳曼队。
输出：
{
  "atoms": ["内马尔获得他的第一个欧洲金靴奖的赛季是2016-2017赛季。", "2016-2017赛季内马尔效力于法甲巴黎圣日耳曼队。"]
}
输入：{sentence}

---

prompt:
请根据提供的参考答案验证原子事实中的陈述是否正确。
若原子事实的陈述与参考答案的事实一致，输出“是”；否则输出“否”。
我将给你一个示例，使用 JSON 格式返回答案。
{
  "答案": xxx,
}
示例:
问题：电影《肖申克的救赎》的导演是谁？
参考答案：电影《肖申克的救赎》的导演是弗兰克·德拉邦特。
原子事实：弗兰克·德拉邦特的英文名是Frank Darabont。
答案：否
问题：{input_question}
参考答案：{input_answer}
原子事实：{input_atom}
输出：

Figure C14: The prompt of FACTSCORE method.

prompt:

```
ABS_SYSTEM_PROMPT = "You are a fair judge assistant tasked with providing clear, objective feedback
based on specific criteria, ensuring each assessment reflects the absolute standards set for performance."

# Absolute prompt template with placeholders for dynamic values
ABSOLUTE_PROMPT = """###Task Description:
An instruction (might include an Input inside it), a response to evaluate, a reference answer that gets a score
of 5.00, and a score rubric representing an evaluation criteria are given.
1. Write a detailed feedback that assesses the quality of the response strictly based on the given score rubric,
not evaluating in general.
2. After writing a feedback, write a score that is a floating-point number between 1.00 and 5.00. You should
refer to the score rubric.
3. The output format should look as follows: "Feedback: (write a feedback for criteria) [RESULT] (a
floating-point number between 1.00 and 5.00)"
4. Please do not generate any other opening, closing, and explanations.

###The instruction to evaluate:
{instruction}

###Response to evaluate:
{response}

###Reference Answer (Score 5.00):
{reference_answer}

###Feedback: """
```

Figure C15: The prompt of PROMETHEUS 2 method.

prompt:
给定一个问题和一个候选答案，请根据问题从候选答案中提取知识图谱（KG），并以三元组（"主语", "谓语", "宾语"）的格式表示KG，每个三元组占一行。
请注意，这是一项“提取”任务，因此不需要关注候选答案的内容是否真实，只需从中提取三元组。重要的是，确保提取的知识图谱不包含重复或冗余的信息。每一条信息应只在知识图谱中表示一次，避免创建与其他三元组相反的三元组。例如，如果提取了三元组（"John", "拥有", "汽车"），就不要再包含（"汽车", "被 John 拥有"），因为这两者表示的是相同的信息，只是顺序相反。

关于冗余的澄清：首先，不要创建反转主语和宾语的三元组来陈述相同的事实。其次，确保每个事实以最简单的形式唯一表示，避免创建多个表示相同信息的三元组。

以下是一些上下文中的示例：

问题：
关于特斯拉机器人，这些段落的别名是什么？

候选答案：
Optimus（或特斯拉机器人）是特斯拉公司正在开发的类人机器人。它在2021年8月19日的人工智能（AI）日活动中宣布。

知识图谱：
("Optimus", "是", "类人机器人")
("Optimus", "由", "特斯拉公司开发")
("Optimus", "也叫", "特斯拉机器人")
("特斯拉公司", "宣布了", "Optimus")
("Optimus的宣布", "发生在", "人工智能（AI）日活动")
("人工智能（AI）日活动", "举行时间", "2021年8月19日")
("人工智能（AI）日活动", "由", "特斯拉公司组织")

问题：
关于Andre Weiss的文本，Andre在巴黎第戎大学待了多少年？

候选答案：
11年

知识图谱：
("Andre Weiss在巴黎第戎大学", "持续时间", "11年")

现在根据提供的问答生成KG：

问题：
{q}

候选答案：
{a}

知识图谱：

Figure C16: The prompt of refchecker method.

prompt：
我有一个由语言模型对某个问题提出的一系列声明列表，请帮助我根据提供的与问题相关的参考文本来检查这些声明是否可以被支持。
参考文本是一系列段落，每个声明都以三元组的形式表示，格式为 ("主语", "谓语", "宾语")。

如果声明被参考文本中的任何段落支持，请回答 'Entailment'。
如果参考文本中没有段落支持该声明，并且该声明与参考文本中的某些段落相矛盾，请回答 'Contradiction'。
如果参考文本中既没有支持也没有反驳该声明，或者不包含验证该声明的信息，请回答 'Neutral'。

请勿使用您自己的知识进行判断，只需比较参考文本和声明来得出答案。

问题:
[QUESTION]

参考文本:
[REFERENCE]

声明:
[CLAIMS]

您的答案应始终只是一个标签列表，每个标签是 ['Entailment', 'Neutral', 'Contradiction'] 中的一个单词，例如，您应该输出如下列表：

Entailment
Neutral
Contradiction
Neutral

请勿在输出中添加解释或您自己的推理，仅输出标签列表。

Figure C17: The prompt of refchecker method.