

# CRITIQUELLM: Towards an Informative Critique Generation Model for Evaluation of Large Language Model Generation

Anonymous ACL submission

## Abstract

Since the natural language processing (NLP) community started to make large language models (LLMs) act as a critic to evaluate the quality of generated texts, most of the existing works train a critique generation model on the evaluation data labeled by GPT-4’s direct prompting. We observe that these models lack the ability to generate informative critiques in both pointwise grading and pairwise comparison especially without references. As a result, their generated critiques cannot provide fine-grained distinguishability on generated texts, causing unsatisfactory evaluation performance. In this paper, we propose a simple yet effective method called *Eval-Instruct*, which can first acquire pointwise grading critiques with pseudo references and then revise these critiques via multi-path prompting to obtain informative evaluation data in different tasks and settings, including pointwise grading and pairwise comparison with / without references. After fine-tuning on these data, the resulting model CRITIQUELLM is empirically shown to outperform ChatGPT and all the open-source baselines and even achieve comparable evaluation performance to GPT-4 in system-level correlations of pointwise grading. We also demonstrate that our generated critiques can act as scalable feedback to further improve the generation quality of strong LLMs like ChatGPT.

## 1 Introduction

Recently, large language models (LLMs) (OpenAI, 2022, 2023; Touvron et al., 2023) have been improved rapidly and approached human-level performance on various natural language processing (NLP) tasks, such as question answering, text summarization, dialogue generation, and code generation (Laskar et al., 2023). How to automatically measure the performance of LLMs has now become an essential research problem and attracted extensive attention (Chang et al., 2023; Bai et al.,

2023b; Zhang et al., 2023; Liu et al., 2024). Strong evaluation methods are expected to provide high-quality critiques (including not only rating scores but also explanations or reasons) that act as scalable feedback and guide LLMs to improve persistently (Cui et al., 2023).

Traditional evaluation metrics, usually based on n-gram overlap between generated texts and reference texts (such as BLEU (Papineni et al., 2002) and ROUGE (Lin, 2004)), have limited effectiveness. Recent works mostly resort to model-based evaluation metrics, especially LLM-based ones (Wang et al., 2023a; Liu et al., 2023b; Zheng et al., 2023). Since most of the best-performing LLMs such as ChatGPT (OpenAI, 2022) and GPT-4 (OpenAI, 2023) can only be accessed via OpenAI APIs, researchers start to automatically collect evaluation data by directly prompting GPT-4 and train their own evaluation models, aiming to avoid potential risks of commercial APIs, such as high cost, unstable usage, and data leakage (Zheng et al., 2023; Wang et al., 2024; Li et al., 2024).

However, we argue that these evaluation models are still struggling to generate informative critiques in different evaluation tasks including pointwise grading and pairwise comparison. Especially in the challenging reference-free setting, these models tend to generate general critiques without fine-grained distinguishability on generated texts, causing unsatisfactory evaluation performance (Zheng et al., 2023).

In this work, we propose a simple yet effective method called *Eval-Instruct*, which can automatically construct informative instruction-tuning data for different evaluation tasks and settings, including pointwise grading and pairwise comparison with / without references. Our main idea is to fully utilize referenced pointwise grading critiques, which are shown to possess rich information with the assistance of references and elaborate prompt design (Zheng et al., 2023; Liu et al., 2023a), to

construct evaluation data for other tasks and settings. Specifically, after acquiring pointwise grading critiques with pseudo references via GPT-4, we devise a multi-path prompting method including two strategies: 1) **Pointwise-to-Pairwise** Prompting aims to inject pointwise grading critiques into pairwise critiques and enrich them with more information about the respective quality of text pairs. 2) **Referenced-to-Reference-Free** Prompting is targeted at removing direct comparison with references in referenced critiques, while keeping other details to improve the specificity of reference-free critiques. The evaluation data in different tasks and settings can be acquired via different paths consisting of these two strategies. And we also design a cross validation mechanism to improve the data quality of reference-free pairwise comparison because both of the two paths reach this task. After fine-tuning on the data of all the tasks and settings, the resulting model CRITIQUELLM is empirically shown to outperform all the open-source baselines and even achieve comparable performance with GPT-4 in system-level correlations of pointwise grading. We also show the potential of CRITIQUELLM to act as effective feedback to enhance the performance of LLMs like ChatGPT.

Our main contributions are as follows:

- We propose an evaluation data construction method called Eval-Instruct to automatically acquire informative evaluation data in both pointwise grading and pairwise comparison with / without references.
- We conduct extensive experiments on CRITIQUELLM, which is fine-tuned on the data constructed by Eval-Instruct. Experimental results on three instruction following benchmark datasets show that our model can outperform all the open-source baselines and even perform comparably with GPT-4 in system-level correlations of pointwise grading.
- We reveal the potential of CRITIQUELLM to guide LLMs to improve persistently by showing the positive impact of our generated critiques as scalable feedback on the generation quality of LLMs.

## 2 Related Work

Evaluation is a long-standing task in NLP, which becomes more challenging with the rapid development of LLMs (Celikyilmaz et al., 2020; Chang

et al., 2023). Currently, there are mainly two lines of work on LLM evaluation, including NLU-style and NLG-style evaluations. NLU-style evaluation methods utilize natural language understanding (NLU) tasks such as multi-choice QA to measure the performance of LLMs via simple objective metrics (such as accuracy and F1 score) (Hendrycks et al., 2021; Zhong et al., 2023; Huang et al., 2023b), which may deviate from the common usage of LLMs and may not exactly reflect the ability of LLMs in generating responses for user queries.

NLG-style evaluation methods extend metrics for natural language generation (NLG) tasks and expect to apply them to the measurement of LLM’s performance, which are the main focus of this paper. Compared with early metrics that depend on the n-gram overlap between generated texts and reference texts (Papineni et al., 2002; Banerjee and Lavie, 2005; Lin, 2004), recently proposed metrics based on state-of-the-art LLMs like GPT-4 (OpenAI, 2023) are shown to be strong evaluators due to the encouraging effectiveness of LLMs and the simplicity of formulating evaluation tasks as instruction-following tasks (Wang et al., 2023a; Chen et al., 2023; Liu et al., 2023b; Zheng et al., 2023; Ke et al., 2023; Fu et al., 2023). Since most of the state-of-the-art LLMs can only be accessed via APIs, researchers start to automatically collect evaluation data by directly prompting GPT-4 and train their own evaluation models to provide stable and effective evaluations at a lower cost (Wang et al., 2024; Li et al., 2024; Kim et al., 2024).

The concurrent works similar to ours are the LLMs specifically trained for evaluation tasks such as PandaLM (Wang et al., 2024), JudgeLM (Zhu et al., 2023), and AUTO-J (Li et al., 2024). Compared with these works, our work is the first attempt to deal with the challenge of uninformative critique generation which commonly appears in recent LLM-based evaluation models especially without references. Also, instead of prompting GPT-4 directly, our proposed Eval-Instruct can fully utilize the connection among different evaluation tasks and settings to construct informative evaluation data, which are empirically shown to improve the quality of generated critiques.

## 3 Method

### 3.1 Task Definition and Method Overview

This paper mainly involves two typical evaluation tasks: 1) **Pointwise Grading**: Given a user query

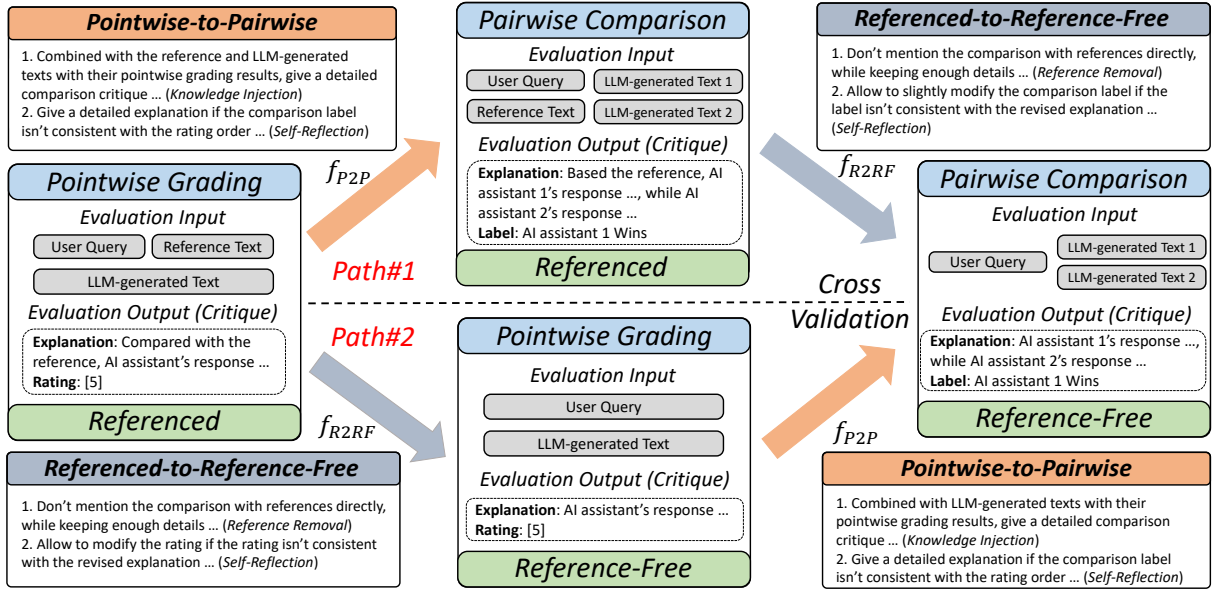


Figure 1: Overview of Eval-Instruct. Starting from referenced pointwise grading data, our proposed multi-path prompting method can apply pointwise-to-pairwise and referenced-to-reference-free prompting strategies to acquire evaluation data in other tasks and settings via two different paths. Cross validation is adopted to filter out the contradictory data from these two paths and further improve the data quality.

182  $q$ , a LLM-generated text  $x$ , and a reference text  $r$  (omitted in the reference-free setting), the goal is to obtain a critique  $c$  including a rating score and an explanation to support this score. 2) **Pairwise Comparison**: Given a user query  $q$ , two LLM-generated texts  $x_1$  and  $x_2$ , and a reference text  $r$  (omitted in the reference-free setting), our purpose is to acquire a critique  $c$  including a comparison label (i.e., win / tie / lose) and an explanation to support this label.

192 Our method consists of the following steps. We first construct an informative instruction-tuning dataset for different evaluation tasks and settings, including pointwise grading and pairwise comparison with / without references (§3.2). Specifically, after collecting user queries, LLM-generated texts, and pseudo references (§3.2.1), we can acquire high-quality referenced pointwise grading critiques via elaborately prompting GPT-4. Then, we devise a multi-path prompting method to construct informative evaluation data in other tasks and settings, which covers pointwise-to-pairwise and referenced-to-reference-free prompting strategies (§3.2.2). Since there are two paths to obtain reference-free pairwise comparison data, we design a cross validation mechanism to filter out the contradictory data and improve the quality (§3.2.3). Finally, we perform supervised fine-tuning on the automatically constructed evaluation data in a multi-

211 task manner to train a unified critique generation model for different evaluation tasks and settings (212 §3.3). (213

## 214 3.2 Evaluation-Oriented Instruction Data Construction (*Eval-Instruct*) (215

### 216 3.2.1 Pseudo Reference Collection (217

218 To construct instruction-tuning data for evaluation, it is imperative to first obtain the evaluation input, including user queries, LLM-generated texts, and references. We refer to recent works on instruction following (Liu et al., 2023a; Li et al., 2024; Zhang et al., 2024) and merge their task taxonomy to consider ten instruction following tasks covering diverse NLP applications in real-world scenarios<sup>1</sup>. We utilize self-instruct (Wang et al., 2023d) to augment seed queries of these tasks which are publicly available and conduct strictly filtering to improve the data quality. The details are provided in Appendix A. (228

229 Then, we collect LLM-generated texts from 10 representative models, which cover different levels of generation qualities, including GPT-4 (OpenAI, 2023), ChatGPT (OpenAI, 2022), two versions of ChatGLM (Du et al., 2022; Zeng et al., 2023), (234

<sup>1</sup>Our task taxonomy contains fundamental language ability, advanced Chinese understanding, open-ended question answering, writing ability, logical reasoning, mathematics, task-oriented role play, professional knowledge, code generation, and multi-lingual ability.

MOSS (Sun et al., 2023), Minimax<sup>2</sup>, Sparkdesk<sup>3</sup>, Chinese-Llama2-7B-Chat<sup>4</sup>, Baichuan2-13B-Chat (Yang et al., 2023), and Ernie Bot<sup>5</sup>. We further filter out the generated results by removing a small number of failure cases, such as empty responses.

Finally, we select the best-performing LLM (i.e., GPT-4) and manually check its generated texts for each user query, while revising them if necessary to improve the quality. Thus, these generated texts after manual check and revise can act as pseudo references to assist the evaluation data construction.

### 3.2.2 Multi-Path Prompting

To acquire high-quality evaluation data in different evaluation tasks and settings, we first construct referenced pointwise grading critiques by prompting GPT-4 with the assistance of pseudo references and well-designed prompts like Liu et al. (2023a), which are empirically shown to be informative (Zheng et al., 2023). Then, regarding this setting as a beginning, we devise a multi-path prompting method to obtain evaluation data in other tasks and settings. As shown in Figure 1, there are two main prompting strategies:

(1) **Pointwise-to-Pairwise Prompting** ( $f_{P2P}$ ): This prompting strategy injects pointwise grading critiques of generated texts into pairwise comparison critiques, enriching them with information about the respective text quality. Meanwhile, it requires self-reflection on the pointwise critiques generated by GPT-4 before obtaining the final pairwise comparison results.

(2) **Referenced-to-Reference-Free Prompting** ( $f_{R2RF}$ ): This prompting strategy aims to remove direct comparison with references while keeping informative contents from references. It also requires GPT-4 to self-reflect whether the evaluation results including scores / labels and revised explanations are consistent, and modify the results if necessary.

Equipped with the above prompting strategies, we have two paths to construct evaluation data in different tasks and settings. Assume that  $D^{point,r} = \{(q_i, r_i, x_i, c_i^{point,r})\}_{i=1}^N$  indicates the referenced pointwise grading dataset constructed above and  $c_i^{point,r}$  represents the critique in the corresponding setting, our purpose is to acquire the datasets  $D^{pair,r}$ ,  $D^{point,rf}$ ,  $D^{pair,rf}$  via differ-

ent paths, where *point/pair* means pointwise / pairwise evaluation and *r/rf* indicates referenced / reference-free evaluation, respectively. The two paths are devised as follows.

**Path#1:**  $D^{point,r} \xrightarrow{f_{P2P}} D^{pair,r} \xrightarrow{f_{R2RF}} D^{pair,rf}$

As shown in Path#1 of Figure 1, we firstly conduct pointwise-to-pairwise prompting to acquire the referenced pairwise comparison dataset  $D^{pair,r} = \{(q_i, r_i, x_{i,1}, x_{i,2}, c_i^{pair,r})\}_{i=1}^M$ :

$$c_i^{pair,r} = f_{P2P}(q_i, r_i, x_{i,1}, x_{i,2}, c_{i,1}^{point,r}, c_{i,2}^{point,r}) \quad i = 1, 2, \dots, M \quad (1)$$

where  $q_i, r_i, x_{i,1}, x_{i,2}$  indicate the user query, the reference, and two generated texts of the  $i$ -th data, respectively.  $c_{i,1}^{point,r}, c_{i,2}^{point,r}, c_i^{pair,r}$  are the referenced pointwise and pairwise evaluation results of  $x_{i,1}, x_{i,2}$ , respectively<sup>6</sup>. Then, we can apply referenced-to-reference-free prompting to obtain  $D^{pair,rf} = \{(q_i, x_{i,1}, x_{i,2}, c_i^{pair,rf})\}$ :

$$c_i^{pair,rf,1} = f_{R2RF}(q_i, r_i, x_{i,1}, x_{i,2}, c_i^{pair,r}) \quad i = 1, 2, \dots, M \quad (2)$$

where  $c_i^{pair,rf,1}$  means the reference-free pairwise comparison critique of the  $i$ -th data from Path#1.

**Path#2:**  $D^{point,r} \xrightarrow{f_{R2RF}} D^{point,rf} \xrightarrow{f_{P2P}} D^{pair,rf}$

Similarly, as shown in Path#2 of Figure 1, we can exchange the order of two prompting strategies applied to  $D^{point,r}$  accordingly. In this way, we can in turn acquire  $D^{point,rf}$  and  $D^{pair,rf}$ :

$$c_i^{point,rf} = f_{R2RF}(q_i, r_i, x_i, c_i^{point,r}) \quad i = 1, 2, \dots, N \quad (3)$$

$$c_i^{pair,rf,2} = f_{P2P}(q_i, r_i, x_{i,1}, x_{i,2}, c_{i,1}^{point,rf}, c_{i,2}^{point,rf}) \quad i = 1, 2, \dots, M \quad (4)$$

where  $c_i^{pair,rf,2}$  denotes the reference-free pairwise comparison critique of the  $i$ -th data from Path#2.

### 3.2.3 Cross Validation

Since both of the two paths finally reach  $D^{pair,rf}$ , we design a cross validation mechanism to further improve the data quality. Specifically,  $D^{pair,rf}$  only contains the data whose comparison labels from two paths are consistent. In this case, the critiques from both of the two paths are added to  $D^{pair,rf}$ . The other data with contradictory comparison labels are strictly filtered.

<sup>6</sup>We conduct strictly rule-based filtering after each prompting step to remove low-quality data with errors in format and other aspects, which is omitted in this subsection.

<sup>2</sup><https://api.minimax.chat/>

<sup>3</sup><https://xinghuo.xfyun.cn/>

<sup>4</sup><https://huggingface.co/FlagAlpha/Llama2-Chinese-7b-Chat/>

<sup>5</sup><https://yiyao.baidu.com/>

### 3.3 Supervised Fine-Tuning

We perform supervised fine-tuning on the LLM  $P_\theta$  using all the constructed training data in a multi-task manner to obtain CRITIQUELLM:

$$\begin{aligned} \mathcal{L} = & -\frac{1}{N} \sum_{i=1}^N P_\theta(c_i^{point,r} | q_i, r_i, x_i) \\ & -\frac{1}{N} \sum_{i=1}^N P_\theta(c_i^{point,rf} | q_i, x_i) \\ & -\frac{1}{M} \sum_{i=1}^M P_\theta(c_i^{pair,r} | q_i, r_i, x_{i,1}, x_{i,2}) \\ & -\frac{1}{M'} \sum_{i=1}^{M'} P_\theta(c_i^{pair,rf} | q_i, x_{i,1}, x_{i,2}) \end{aligned}$$

where  $M'$  indicates the data amount of  $D^{pair,rf}$  after cross validation. During fine-tuning, we follow Bai et al. (2022) to add simplified prompts to distinguish different parts of inputs. We also follow Li et al. (2024) to augment pairwise training data via swapping the order of two generated texts and exchanging the corresponding contents in critiques.

## 4 Experiment

### 4.1 Dataset

We adopt three benchmark datasets on open-ended instruction following, which involve various NLP tasks in LLM’s real-world scenarios<sup>7</sup>. The datasets also cover all the evaluation tasks and settings in this paper. The statistics are shown in Table 1.

**AlignBench** (Liu et al., 2023a): This benchmark includes 8 categories of instruction following tasks and 8 LLMs for generation. It provides an evaluation dataset with human-annotated scores on the quality of generated texts. In addition to using human-annotated scores for measuring pointwise grading performance, we also follow the original paper to sample text pairs of the same query for pairwise comparison<sup>8</sup>, whose label is automatically determined by their pointwise scores.

**AUTO-J (Eval-P)** (Li et al., 2024): This benchmark provides 1,392 pairwise comparison data,

<sup>7</sup>We have conducted string matching to show that there is no overlap between the queries in the training and test sets.

<sup>8</sup>The authors in the original paper of AlignBench (Liu et al., 2023a) collect all the pairs of generated texts for each query ( $\sim 10,000$  pairwise comparison data), causing high demand of computational resources and API costs for LLM-based evaluation methods. Thus, we randomly sample a subset ( $\sim 1,000$  pairwise comparison data) to test our method and all the baselines for a fair comparison.

Dataset	Task	Setting	#Models	#Samples / #Pairs	Length
AlignBench	Pointwise	R / R-F	8	3,200	274
	Pairwise	R / R-F	8	1,600	293
AUTO-J (Eval-P)	Pairwise	R-F	6	1,392	372
LLMEval	Pairwise	R-F	11	1,530	283

Table 1: Statistics of the benchmark datasets, including the evaluation task / setting, the number of models / samples / pairs, and the average length of generated texts. R / R-F indicates referenced / reference-free evaluation, respectively.

each of which contains a user query, two LLM-generated texts, and a human-annotated preference label. These data involve 58 real-world scenarios and 6 model families for generation.

**LLMEval** (Zhang et al., 2024): This benchmark designs 17 types of user queries covering representative NLP tasks in real-world scenarios, and provides  $\sim 100,000$  pairwise comparison data with human-annotated labels. Due to the limitation of computational resources and API costs for LLM-based evaluation methods, we randomly sample a subset ( $\sim 1,000$ ) to measure the performance of our method and all the baselines for a fair comparison.

### 4.2 Baselines

We choose state-of-the-art general LLMs and evaluation-specific LLMs as our baselines.

**General LLMs:** We adopt ChatGPT (gpt-3.5-turbo-1106) (OpenAI, 2022), GPT-4 (gpt-4-1106-preview) (OpenAI, 2023), ChatGLM3-6B (Du et al., 2022; Zeng et al., 2023), Baichuan2-13B-Chat (Yang et al., 2023), and Qwen-14B-Chat (Bai et al., 2023a) as our general baselines<sup>9</sup>. These general LLMs can perform as an evaluator for pointwise grading and pairwise comparison via elaborate prompts without further training. We directly prompt these LLM to obtain evaluation results in single-turn interaction.

**Evaluation-Specific LLMs:** We select AUTO-J-Bilingual-6B (Li et al., 2024) and JudgeLM-13B (Zhu et al., 2023) as our task-specific baselines. These two baselines are designed for specific evaluation tasks and settings.

### 4.3 Implementation Details

We choose ChatGLM3-6B (Du et al., 2022; Zeng et al., 2023) as our base model and use Zero Redundancy Optimizer (ZeRO) (Rajbhandari et al., 2020)

<sup>9</sup>Since our benchmark datasets are mainly in Chinese, we select state-of-the-art Chinese LLMs as competitive baselines.

Level	Text-Level						System-Level					
Setting	Referenced			Reference-Free			Referenced			Reference-Free		
Metric	$r$	$\rho$	$\tau$	$r$	$\rho$	$\tau$	$r$	$\rho$	$\tau$	$r$	$\rho$	$\tau$
<i>Closed-Source Evaluation Models</i>												
ChatGPT	0.443	0.421	0.379	0.292	0.287	0.266	0.955	0.976	0.929	0.778	0.833	0.643
GPT-4	<u>0.629</u>	<u>0.583</u>	<u>0.532</u>	<u>0.523</u>	<u>0.494</u>	<u>0.447</u>	<u>0.995</u>	<u>1.000</u>	<u>1.000</u>	<u>0.997</u>	<u>0.976</u>	<u>0.929</u>
<i>Open-Source Evaluation Models</i>												
ChatGLM3-6B	0.223	0.222	0.207	0.159	0.150	0.140	0.790	0.833	0.643	0.544	0.548	0.429
Baichuan2-13B-Chat	0.199	0.200	0.187	0.125	0.117	0.110	0.854	0.929	0.786	0.663	0.527	0.400
Qwen-14B-Chat	0.373	0.379	0.358	0.255	0.254	0.239	0.901	0.929	0.786	0.772	0.833	0.643
JudgeLM-13B	0.450	0.430	0.391	0.170	0.162	0.155	0.984	0.976	0.929	0.717	0.905	0.786
AUTO-J-Bilingual-6B	-	-	-	0.044	0.045	0.041	-	-	-	0.558	0.571	0.500
CRITIQUELLM (Ours)	<b>0.555</b>	<b>0.523</b>	<b>0.477</b>	<b>0.366</b>	<b>0.352</b>	<b>0.319</b>	<u>0.995</u>	<u>1.000</u>	<u>1.000</u>	<b>0.954</b>	<b>0.976</b>	<b>0.929</b>

Table 2: Text-level and system-level Pearson ( $r$ ), Spearman ( $\rho$ ), and Kendall ( $\tau$ ) correlations in referenced and reference-free settings of pointwise grading on AlignBench. The highest correlation among the methods based on local models is **bold**, while the highest correlation overall is underlined. - means that AUTO-J-Bilingual-6B cannot support referenced pointwise grading.

Dataset	AlignBench				AUTO-J (Eval-P)		LLMEval	
Setting	Referenced		Reference-Free		Reference-Free		Reference-Free	
Metric	Agr.	Cons.	Agr.	Cons.	Agr.	Cons.	Agr.	Cons.
<i>Closed-Source Evaluation Models</i>								
ChatGPT	32.50	38.56	39.56	53.94	42.74	62.43	40.07	64.58
GPT-4	<u>74.69</u>	86.75	<u>70.25</u>	84.88	<u>62.28</u>	86.28	<u>50.98</u>	84.71
<i>Open-Source Evaluation Models</i>								
ChatGLM3-6B	17.75	31.84	24.75	42.88	14.15	26.22	28.56	51.70
Baichuan2-13B-Chat	35.81	50.06	27.06	40.82	19.40	32.33	23.53	43.27
Qwen-14B-Chat	33.81	43.25	42.06	58.75	31.68	52.08	42.81	69.61
JudgeLM-13B	-	-	42.50	66.00	35.13	58.19	44.77	75.82
AUTO-J-Bilingual-6B	-	-	26.00	45.38	49.43	77.23	27.58	55.56
CRITIQUELLM (Ours)	<b>70.56</b>	<b>89.25</b>	<b>58.81</b>	<b>83.06</b>	<b>50.93</b>	<b>82.76</b>	<b>50.72</b>	<b>85.95</b>

Table 3: Agreement (Agr.) and consistency (Cons.) rates in pairwise comparison evaluation. The highest correlation among the methods based on local models is **bold**, while the highest correlation overall is underlined. - means that JudgeLM-13B and AUTO-J-Bilingual-6B cannot support referenced pairwise comparison.

stage 2 framework from the Deepspeed (Rasley et al., 2020) library. CRITIQUELLM is trained on 8 A800 GPUs. The number of training samples for  $D^{point,r} / D^{point,r,f} / D^{pair,r} / D^{pair,r,f}$  is 12,102 / 12,095 / 6,190 / 5,428, respectively. We use AdamW (Kingma and Ba, 2015) optimizer with the weight decay of 0.1. The peak learning rate is  $6e-5$  with 10% warmup ratio. We set the maximum sequence length to 8,192 and the batch size to 64. The number of training epochs is 5. We use greedy decoding in the main result and investigate the effect of different decoding methods on our model in §4.7. For beam search, we set the beam size to 4. For the sampling-based decoding method, we adopt Nucleus Sampling (i.e., Top- $p$  Sampling) (Holtzman et al., 2020) and set both the temperature and  $p$

to 0.9. For self-consistency decoding (Wang et al., 2023c), the number of candidate critiques is 5.

## 4.4 Main Results

### 4.4.1 Pointwise Grading

Following Colombo et al. (2022), we adopt text-level and system-level Pearson ( $r$ ), Spearman ( $\rho$ ), and Kendall ( $\tau$ ) correlation coefficients between human judgments and automatic metrics to measure the pointwise grading performance. Text-level correlation is computed by the average score over the correlation coefficients between human judgments and automatic metrics for all the generated texts of each instruction. For comparison, system-level correlation is obtained by the correlation coefficients between human judgments and automatic metrics

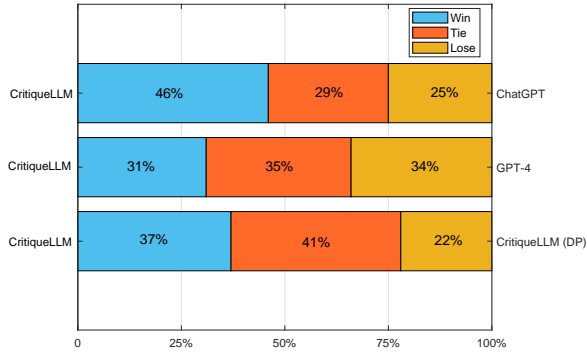


Figure 2: Critique quality evaluation results. The percentages indicate the preference results between CRITIQUELLM and other models via GPT-4’s evaluation and human verification.

of each LLM’s score, which is the average value over all the scores of the corresponding model on the dataset.

The results in Table 2 show that CRITIQUELLM can achieve comparable performance with GPT-4 especially in system-level correlations, while outperforming ChatGPT and all the open-source baselines. This indicates that our proposed method can successfully improve the quality of generated critiques. We can observe that system-level correlations of CRITIQUELLM are almost the same as those of GPT-4, which even approach 1.0. This demonstrate that our model is nearly able to distinguish the overall performance of all the eight LLMs.

#### 4.4.2 Pairwise Comparison

Following Li et al. (2024), we adopt agreement and consistency rates to test the pairwise comparison performance. Specifically, we conduct two comparisons for each data sample via swapping the order of two generated texts. We consider the model’s evaluation result to agree with humans only when the two comparison results are consistent and align with the human preference label.

The results in Table 3 show that CRITIQUELLM can beat ChatGPT and all the open-source baselines in both agreement and consistency rates. Compared with GPT-4, CRITIQUELLM achieves comparable performance especially in the consistency rate. This indicates that CRITIQUELLM equipped with high-quality evaluation data in different tasks and settings not only performs well in pointwise grading, but also has a strong evaluation ability in pairwise comparison.

## 4.5 Analysis on Critique Quality

To further measure the quality of generated critiques, we follow Chen et al. (2024) to combine automatic and human evaluations. Specifically, We follow existing works (Wang et al., 2023b; Sun et al., 2024) to devise an evaluation prompt for GPT-4 to judge the quality of generated critiques. After GPT-4’s evaluation, we manually verify the results and modify them if necessary. We randomly select 100 evaluation data for pointwise grading and pairwise comparison, respectively. The selected data for pairwise comparison are from the mix of three datasets. And we collect generated critiques from CRITIQUELLM, state-of-the-art evaluators (i.e., ChatGPT and GPT-4), and an alternative model CRITIQUELLM (DP) whose training data in different tasks and settings are acquired from GPT-4’s direct prompting. For each pair of critiques (one from CRITIQUELLM and the other from a baseline / an alternative model, given the same evaluation input), GPT-4 are required to label which critique is better (i.e. win, lose or tie) in terms of correctness, helpfulness, and informativeness. The priority of these three aspects is set to follow the above order. Then, human verification is conducted to check GPT-4’s evaluation on critiques.

The results are shown in Figure 2. We can observe that CRITIQUELLM can achieve superior performance over ChatGPT and CritiqueLLM (DP), and even perform comparably with GPT-4. This demonstrates that our proposed evaluation data construction method can successfully improve the overall quality of generated critiques and enhance their informativeness.

## 4.6 Analysis of Critique as Feedback

To investigate whether the critiques generated by our model can serve as feedback to improve the quality of LLM-generated texts, we employ ChatGPT, GPT-4, and CRITIQUELLM to provide critiques for the generated texts of ChatGPT in the reference-free setting. Then, we instruct ChatGPT to modify its original generation based on the critiques. Finally, we use GPT-4 to perform referenced evaluations on the original texts and the modified texts generated by ChatGPT, respectively.

The results in Table 4 show that the critiques from CRITIQUELLM can serve as positive feedback whose contributed improvement on the overall score is close to that from the GPT-4’s critiques. This further verifies the utility of CRITIQUELLM

Critique Model	Overall	Logical	Open-ended QA	Professional	Fundamental	Mathematics	Role Play	Writing	Chinese Understanding
None	6.385	<b>5.318</b>	7.000	5.824	6.310	6.160	7.260	7.154	6.000
ChatGPT	6.300	5.045	6.762	6.353	6.276	5.760	7.000	6.885	6.063
GPT-4	<b>6.545</b>	4.455	7.190	6.588	<b>6.897</b>	<b>6.200</b>	7.111	7.077	<b>6.563</b>
CRITIQUELLM	6.530	5.136	<b>7.381</b>	<b>6.765</b>	6.414	6.000	<b>7.407</b>	<b>7.192</b>	5.315

Table 4: GPT-4’s referenced pointwise scores on AlignBench for original generated texts from ChatGPT (i.e., *None*) and modified texts based on each critique generation model, respectively.

to provide informative critiques as scalable feedback that can guide LLMs towards better generation. We also notice that the critiques from ChatGPT itself have a negative impact on the overall quality of its generated texts. This phenomenon is consistent with recent works that doubt the self-correction ability of LLMs without external inputs (Huang et al., 2023a; Stechly et al., 2023; Valmeekam et al., 2023).

We also report the evaluation scores before and after the critique-based modification across different tasks in Table 4. It is notable that the critiques from CRITIQUELLM can help enhance the quality of generated texts in a majority of tasks. However, in the tasks of logical reasoning, mathematics, and advanced Chinese understanding which are mostly hard tasks involving reasoning, the critiques from CRITIQUELLM seem to degrade the performance. We manually checked error cases and found that our model obtained misleading critiques on the reasoning process of generated texts. Since the evaluation of reasoning chains remains a challenging task (Golovneva et al., 2023) even for GPT-4, we leave further investigation in these tasks as future work.

#### 4.7 Ablation Study

Setting	Pointwise		Pairwise	
	R	R-F	R	R-F
Metric	$r$	$r$	Agr.	Agr.
CRITIQUELLM	0.555	0.366	<b>70.56</b>	<b>58.81</b>
<i>Fine-Tuning Data</i>				
w/o Cross Validation	0.566	0.361	66.13	57.44
<i>Decoding Strategy</i>				
w/ Beam Search	0.554	0.374	70.31	57.75
w/ Sampling	0.547	0.353	68.69	57.31
w/ Self-Consistency	<b>0.573</b>	<b>0.384</b>	69.13	58.44
<i>Explanation</i>				
w/o Explanation	0.509	0.332	60.19	51.56

Table 5: Text-level Pearson ( $r$ ) correlations and agreement rates (Agr.) of ablation models in reference (R) and reference-free (R-F) settings of AlignBench.

To further investigate the impact of each part on CRITIQUELLM, we conduct additional ablation studies. For fine-tuning data, we remove the cross validation module (§3.2.3) to explore its impact on the evaluation performance. Table 5 shows that the performance of CRITIQUELLM degrades especially in pairwise comparison, demonstrating that cross validation can filter out low-quality evaluation data and contribute to the final performance.

As for decoding strategies, we show the evaluation performance of three decoding strategies in addition to greedy decoding in the main result, including beam search, Nucleus Sampling (Holtzman et al., 2020), and self-consistency decoding (Wang et al., 2023c). The results in Table 5 show that the self-consistency decoding method can enhance the performance of our model especially in pointwise grading. Meanwhile, greedy decoding performs best in pairwise comparison, while achieving comparable performance with other methods in pointwise grading at a smaller computational cost.

For evaluation explanations, we remove the explanations in the critiques of training data. The results in Table 5 show that the performance of CRITIQUELLM largely degrades in both pointwise and pairwise evaluations without explanations. This verifies the positive impact of explanations on the final performance, which play a similar role to chain-of-thought reasoning (Wei et al., 2022).

## 5 Conclusion

We present an evaluation data construction method called Eval-Instruct, which can automatically construct informative evaluation data in both pointwise grading and pairwise comparison with / without references. After fine-tuning on the data from Eval-Instruct, the resulting model CRITIQUELLM can beat ChatGPT and all the open-source baselines, and perform comparably with GPT-4 in system-level correlations of pointwise grading. CRITIQUELLM can also provide scalable feedback which can improve the generation quality of LLMs.

## 574 Limitations

575 The limitations of our work are summarized as  
576 follows:

577 (1) In our method of multi-path prompting, we de-  
578 vise two prompting strategies to enrich the informa-  
579 tion in the resulting critiques, which can improve  
580 the critique quality. However, this method also  
581 increases the length of input prompts and lead to  
582 higher API costs when constructing evaluation data  
583 in different tasks and settings. We believe that it  
584 is not a severe problem because data acquisition  
585 is single-round and we do not repeatedly acquire  
586 critiques for the same evaluation input. Also, our  
587 proposed critique generation model based on open-  
588 source LLMs (i.e., ChatGLM3-6B) can achieve  
589 comparable performance with GPT-4 in some as-  
590 pects, which may save the cost for LLM evaluation  
591 via APIs and avoid the risks such as unstable usage  
592 and data leakage.

593 (2) Similar to other model-based evaluation meth-  
594 ods, our evaluation model suffers from the self-  
595 evaluation bias (He et al., 2023) (also known as  
596 self-enhancement bias (Zheng et al., 2023)), which  
597 indicates the preference on the generated texts from  
598 the same base model. This bias is commonly rec-  
599 ognized even in state-of-the-art LLM-based eval-  
600 uators like GPT-4. We argue that researchers and  
601 developers can use multiple LLM-based evalua-  
602 tors with different base models including CRI-  
603 TIQUELLM to avoid self-evaluation bias towards  
604 specific generation models. Since there does not  
605 exist a satisfactory solution to the self-evaluation  
606 bias currently, we leave the further investigation as  
607 important future work.

## 608 References

609 Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang,  
610 Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei  
611 Huang, et al. 2023a. Qwen technical report. *arXiv*  
612 *preprint arXiv:2309.16609*.

613 Yuntao Bai, Saurav Kadavath, Sandipan Kundu,  
614 Amanda Askell, Jackson Kernion, Andy Jones,  
615 Anna Chen, Anna Goldie, Azalia Mirhoseini,  
616 Cameron McKinnon, et al. 2022. Constitutional  
617 ai: Harmlessness from ai feedback. *arXiv preprint*  
618 *arXiv:2212.08073*.

619 Yushi Bai, Xin Lv, Jiajie Zhang, Hongchang Lyu,  
620 Jiankai Tang, Zhidian Huang, Zhengxiao Du, Xiao  
621 Liu, Aohan Zeng, Lei Hou, et al. 2023b. Longbench:  
622 A bilingual, multitask benchmark for long context  
623 understanding. *arXiv preprint arXiv:2308.14508*.

Satanjeev Banerjee and Alon Lavie. 2005. METEOR:  
an automatic metric for MT evaluation with improved  
correlation with human judgments. In *Proceedings of*  
*the Workshop on Intrinsic and Extrinsic Evaluation*  
*Measures for Machine Translation and/or Summa-*  
*rization*, pages 65–72.

Asli Celikyilmaz, Elizabeth Clark, and Jianfeng Gao.  
2020. Evaluation of text generation: A survey. *arXiv*  
*preprint arXiv:2006.14799*.

Yupeng Chang, Xu Wang, Jindong Wang, Yuan Wu,  
Kaijie Zhu, Hao Chen, Linyi Yang, Xiaoyuan Yi,  
Cunxiang Wang, Yidong Wang, Wei Ye, Yue Zhang,  
Yi Chang, Philip S. Yu, Qiang Yang, and Xing Xie.  
2023. A survey on evaluation of large language mod-  
els. *arXiv preprint arXiv:2307.03109*.

Kai Chen, Chunwei Wang, Kuo Yang, Jianhua Han,  
Lanqing Hong, Fei Mi, Hang Xu, Zhengying Liu,  
Wenyong Huang, Zhenguo Li, Dit-Yan Yeung, Lifeng  
Shang, Xin Jiang, and Qun Liu. 2024. Gaining wis-  
dom from setbacks: Aligning large language models  
via mistake analysis. In *The Twelfth International*  
*Conference on Learning Representations*.

Yi Chen, Rui Wang, Haiyun Jiang, Shuming Shi, and  
Ruifeng Xu. 2023. Exploring the use of large lan-  
guage models for reference-free text quality evalua-  
tion: A preliminary empirical study. *arXiv preprint*  
*arXiv:2304.00723*.

Pierre Jean A Colombo, Chloé Clavel, and Pablo Pi-  
antanida. 2022. Infolm: A new metric to evaluate  
summarization & data2text generation. In *Proceed-*  
*ings of the AAAI Conference on Artificial Intelligence*,  
volume 36, pages 10554–10562.

Ganqu Cui, Lifan Yuan, Ning Ding, Guanming Yao,  
Wei Zhu, Yuan Ni, Guotong Xie, Zhiyuan Liu, and  
Maosong Sun. 2023. Ultrafeedback: Boosting lan-  
guage models with high-quality feedback. *arXiv*  
*preprint arXiv:2310.01377*.

Zhengxiao Du, Yujie Qian, Xiao Liu, Ming Ding,  
Jiezhong Qiu, Zhilin Yang, and Jie Tang. 2022. Glm:  
General language model pretraining with autoregres-  
sive blank infilling. In *Proceedings of the 60th An-*  
*nuual Meeting of the Association for Computational*  
*Linguistics (Volume 1: Long Papers)*, pages 320–335.

Jinlan Fu, See-Kiong Ng, Zhengbao Jiang, and Pengfei  
Liu. 2023. Gpyscore: Evaluate as you desire. *arXiv*  
*preprint arXiv:2302.04166*.

Olga Golovneva, Moya Peng Chen, Spencer Poff, Mar-  
tin Corredor, Luke Zettlemoyer, Maryam Fazel-  
Zarandi, and Asli Celikyilmaz. 2023. Roscoe: A  
suite of metrics for scoring step-by-step reasoning.  
In *The Eleventh International Conference on Learn-*  
*ing Representations*.

Tianxing He, Jingyu Zhang, Tianle Wang, Sachin  
Kumar, Kyunghyun Cho, James Glass, and Yulia  
Tsvetkov. 2023. On the blind spots of model-based

679	evaluation metrics for text generation. In <i>Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)</i> , pages 12067–12097.	Xiao Liu, Xuanyu Lei, Shengyuan Wang, Yue Huang, Zhuoer Feng, Bosi Wen, Jiale Cheng, Pei Ke, Yifan Xu, Weng Lam Tam, et al. 2023a. Alignbench: Benchmarking chinese alignment of large language models. <i>arXiv preprint arXiv:2311.18743</i> .	733 734 735 736 737
683	Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding. In <i>9th International Conference on Learning Representations</i> .	Xiao Liu, Hao Yu, Hanchen Zhang, Yifan Xu, Xuanyu Lei, Hanyu Lai, Yu Gu, Hangliang Ding, Kaiwen Men, Kejuan Yang, et al. 2024. Agentbench: Evaluating llms as agents. In <i>The Twelfth International Conference on Learning Representations</i> .	738 739 740 741 742
688	Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2020. The curious case of neural text degeneration. In <i>8th International Conference on Learning Representations</i> .	Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. 2023b. G-eval: NLG evaluation using gpt-4 with better human alignment. In <i>Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing</i> , pages 2511–2522.	743 744 745 746 747 748
692	Jie Huang, Xinyun Chen, Swaroop Mishra, Huaixiu Steven Zheng, Adams Wei Yu, Xinying Song, and Denny Zhou. 2023a. Large language models cannot self-correct reasoning yet. <i>arXiv preprint arXiv:2310.01798</i> .	OpenAI. 2022. <a href="#">Introducing chatgpt</a> .	749
697	Yuzhen Huang, Yuzhuo Bai, Zhihao Zhu, Junlei Zhang, Jinghan Zhang, Tangjun Su, Junteng Liu, Chuancheng Lv, Yikai Zhang, Jiayi Lei, Yao Fu, Maosong Sun, and Junxian He. 2023b. C-eval: A multi-level multi-discipline chinese evaluation suite for foundation models. <i>arXiv preprint arXiv:2305.08322</i> .	OpenAI. 2023. GPT-4 technical report. <i>arXiv preprint arXiv:2303.08774</i> .	750 751
704	Pei Ke, Fei Huang, Fei Mi, Yasheng Wang, Qun Liu, Xiaoyan Zhu, and Minlie Huang. 2023. DecompEval: Evaluating generated texts as unsupervised decomposed question answering. In <i>Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)</i> , pages 9676–9691.	Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In <i>Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics</i> , pages 311–318.	752 753 754 755 756
711	Seungone Kim, Jamin Shin, Yejin Cho, Joel Jang, Shayne Longpre, Hwaran Lee, Sangdoon Yun, Seongjin Shin, Sungdong Kim, James Thorne, et al. 2024. Prometheus: Inducing fine-grained evaluation capability in language models. In <i>The Twelfth International Conference on Learning Representations</i> .	Samyam Rajbhandari, Jeff Rasley, Olatunji Ruwase, and Yuxiong He. 2020. Zero: memory optimizations toward training trillion parameter models. In <i>Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis</i> , page 20.	757 758 759 760 761 762
717	Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In <i>3rd International Conference on Learning Representations</i> .	Jeff Rasley, Samyam Rajbhandari, Olatunji Ruwase, and Yuxiong He. 2020. Deepspeed: System optimizations enable training deep learning models with over 100 billion parameters. In <i>The 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining</i> , pages 3505–3506.	763 764 765 766 767 768
720	Md Tahmid Rahman Laskar, M Saiful Bari, Mizanur Rahman, Md Amran Hossen Bhuiyan, Shafiq Joty, and Jimmy Huang. 2023. A systematic study and comprehensive evaluation of ChatGPT on benchmark datasets. In <i>Findings of the Association for Computational Linguistics: ACL 2023</i> , pages 431–469.	Kaya Stechly, Matthew Marquez, and Subbarao Kambhampati. 2023. Gpt-4 doesn't know it's wrong: An analysis of iterative prompting for reasoning problems. <i>arXiv preprint arXiv:2310.12397</i> .	769 770 771 772
726	Junlong Li, Shichao Sun, Weizhe Yuan, Run-Ze Fan, Hai Zhao, and Pengfei Liu. 2024. Generative judge for evaluating alignment. In <i>The Twelfth International Conference on Learning Representations</i> .	Shichao Sun, Junlong Li, Weizhe Yuan, Ruifeng Yuan, Wenjie Li, and Pengfei Liu. 2024. The critique of critique. <i>arXiv preprint arXiv:2401.04518</i> .	773 774 775
730	Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In <i>Text summarization branches out</i> , pages 74–81.	Tianxiang Sun, Xiaotian Zhang, Zhengfu He, Peng Li, Qinyuan Cheng, Hang Yan, Xiangyang Liu, Yunfan Shao, Qiong Tang, Xingjian Zhao, Ke Chen, Yining Zheng, Zhejian Zhou, Ruixiao Li, Jun Zhan, Yunhua Zhou, Linyang Li, Xiaogui Yang, Lingling Wu, Zhangyue Yin, Xuanjing Huang, and Xipeng Qiu. 2023. Moss: Training conversational language models from synthetic data.	776 777 778 779 780 781 782 783
		Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro,	784 785 786

787	Faisal Azhar, et al. 2023. Llama: Open and efficient foundation language models. <i>arXiv preprint arXiv:2302.13971</i> .	842
788		843
789		844
790	Karthik Valmeekam, Matthew Marquez, and Subbarao Kambhampati. 2023. Can large language models really improve by self-critiquing their own plans? <i>arXiv preprint arXiv:2310.08118</i> .	845
791		846
792		847
793		848
794	Jiaan Wang, Yunlong Liang, Fandong Meng, Haoxiang Shi, Zhixu Li, Jinan Xu, Jianfeng Qu, and Jie Zhou. 2023a. Is chatgpt a good nlg evaluator? a preliminary study. <i>arXiv preprint arXiv:2303.04048</i> .	849
795		850
796		851
797		852
798	Tianlu Wang, Ping Yu, Xiaoqing Ellen Tan, Sean O’Brien, Ramakanth Pasunuru, Jane Dwivedi-Yu, Olga Golovneva, Luke Zettlemoyer, Maryam Fazel-Zarandi, and Asli Celikyilmaz. 2023b. Shepherd: A critic for language model generation. <i>arXiv preprint arXiv:2308.04592</i> .	853
799		854
800		855
801		856
802		857
803		858
804	Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V. Le, Ed H. Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2023c. Self-consistency improves chain of thought reasoning in language models. In <i>The Eleventh International Conference on Learning Representations</i> .	859
805		860
806		861
807		862
808		863
809		864
810	Yidong Wang, Zhuohao Yu, Zhengran Zeng, Linyi Yang, Cunxiang Wang, Hao Chen, Chaoya Jiang, Rui Xie, Jindong Wang, Xing Xie, Wei Ye, Shikun Zhang, and Yue Zhang. 2024. Pandalm: An automatic evaluation benchmark for LLM instruction tuning optimization. In <i>The Twelfth International Conference on Learning Representations</i> .	865
811		866
812		867
813		868
814		869
815		870
816		871
817	Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A. Smith, Daniel Khashabi, and Hannaneh Hajishirzi. 2023d. Self-instruct: Aligning language models with self-generated instructions. In <i>Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics</i> , pages 13484–13508.	872
818		873
819		874
820		875
821		876
822		877
823	Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed Chi, Quoc V Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. In <i>Advances in Neural Information Processing Systems</i> , volume 35, pages 24824–24837.	878
824		879
825		880
826		881
827		882
828		883
829	Aiyuan Yang, Bin Xiao, Bingning Wang, Borong Zhang, Chao Yin, Chenxu Lv, Da Pan, Dian Wang, Dong Yan, Fan Yang, et al. 2023. Baichuan 2: Open large-scale language models. <i>arXiv preprint arXiv:2309.10305</i> .	884
830		885
831		886
832		887
833		888
834	Aohan Zeng, Xiao Liu, Zhengxiao Du, Zihan Wang, Hanyu Lai, Ming Ding, Zhuoyi Yang, Yifan Xu, Wendi Zheng, Xiao Xia, Weng Lam Tam, Zixuan Ma, Yufei Xue, Jidong Zhai, Wenguang Chen, Zhiyuan Liu, Peng Zhang, Yuxiao Dong, and Jie Tang. 2023. GLM-130B: an open bilingual pre-trained model. In <i>The Eleventh International Conference on Learning Representations</i> .	889
835		890
836		891
837		892
838		893
839		894
840		895
841		896
	Yue Zhang, Ming Zhang, Haipeng Yuan, Shichun Liu, Yongyao Shi, Tao Gui, Qi Zhang, and Xuanjing Huang. 2024. Llmeval: A preliminary study on how to evaluate large language models. In <i>The 38th Annual AAAI Conference on Artificial Intelligence</i> .	897
		898
		899
	Zhexin Zhang, Leqi Lei, Lindong Wu, Rui Sun, Yongkang Huang, Chong Long, Xiao Liu, Xuanyu Lei, Jie Tang, and Minlie Huang. 2023. Safety-bench: Evaluating the safety of large language models with multiple choice questions. <i>arXiv preprint arXiv:2309.07045</i> .	900
		901
		902
	Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. Judging LLM-as-a-judge with MT-bench and chatbot arena. In <i>Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track</i> .	903
		904
		905
		906
		907
		908
		909
		910
		911
		912
		913
		914
		915
		916
		917
		918
		919
		920
		921
		922
		923
		924
		925
		926
		927
		928
		929
		930
		931
		932
		933
		934
		935
		936
		937
		938
		939
		940
		941
		942
		943
		944
		945
		946
		947
		948
		949
		950
		951
		952
		953
		954
		955
		956
		957
		958
		959
		960
		961
		962
		963
		964
		965
		966
		967
		968
		969
		970
		971
		972
		973
		974
		975
		976
		977
		978
		979
		980
		981
		982
		983
		984
		985
		986
		987
		988
		989
		990
		991
		992
		993
		994
		995
		996
		997
		998
		999
		1000

Stage	Prompt
<b>Generation</b>	<p>You are asked to provide 10 diverse prompts. These task prompts will be provided to a GPT model and we will evaluate the ability of the GPT model to reply to these prompts. The following are some examples:</p> <ol style="list-style-type: none"> <li>{example prompt 1}</li> <li>{example prompt 2}</li> <li>{example prompt 3}</li> </ol> <p>Here are the requirements you need to follow to provide prompts:</p> <ol style="list-style-type: none"> <li>The prompts need to be complete sentences, not phrases or fragments.</li> <li>The prompts need to be varied, do not use similar prompts.</li> <li>The prompts need to be meaningful, do not use meaningless prompts.</li> <li>The prompts need to have a variety of tones, e.g., combining interrogative and imperative sentences.</li> <li>The prompts need to be challenging, do not use simple directions.</li> <li>The prompts need to be something that the Large Language Model can accomplish. For example, don't ask the assistant to create any visual or audio output. For example, don't ask the assistant to wake you up at 5pm or set a reminder because it can't perform any action. For example, prompts should not be related to audio, video, images, hyperlinks.</li> <li>The prompts are in Simplified Chinese, except for translation-related questions or math-related questions.</li> <li>Some prompts can provide contextual information, should involve realistic data, and should not contain simple placeholders. Not all prompts require input. For example, when a prompts asks for general knowledge information, such as "What is the tallest mountain in the world?", it does not need to provide specific context.</li> </ol> <p>After you have provided the prompts, please add the category of the prompts in a pair of &amp;&amp; sign after the prompt and surround the prompt with in a pair of @@ sign. For example, if the prompt is "@@What is the tallest mountain in the world?@@"&amp;&amp; 基本任务 &amp;&amp;", then the category is 基本任务.</p> <p>The category must be one of the following 10 categories. 1. 基本任务 2. 中文理解 3. 综合问答 4. 文本写作 5. 数学计算 6. 逻辑推理 7. 角色扮演 8. 专业能力 9. 代码生成 10. 多语言能力</p> <p>Here are some examples of prompts you provide:</p> <p>@@example prompt1@@ &amp;&amp;category1&amp;&amp;  @@example prompt2@@ &amp;&amp;category2&amp;&amp;  ...  @@example prompt9@@ &amp;&amp;category9&amp;&amp;  @@example prompt10@@ &amp;&amp;category10&amp;&amp;</p> <p>The following is a list of 10 good task prompts with serial numbers and categories:</p>
<b>Evaluation</b>	<p>已知上面三个问题和它们的类别，现在请你根据以下要求，对这三个问题的题目的难度在1-3分的量表上分别评分：</p> <p>(1). 1分：对于大语言模型来说，这类问题是容易的  (2). 2分：对于大语言模型来说，这类问题是中等难度的  (3). 3分：对于大语言模型来说，这类问题是困难的</p> <p>最后：请将这三个问题，题目用一对@@符号包围，对应的类别用一对&amp;&amp;符号包围，分数用一对##包围，分别带有序号地输出出来：  例如：如果问题1的题目是题目1，类别是评价看法类别，分数是1分，问题2的题目是题目2，类别是基本任务类别，分数是2分，问题3的题目是题目3，类别是写作类类别，分数是3分，那么输出如下：</p> <p>1.@@题目1@@&amp;&amp;评价看法&amp;&amp;##1##  2.@@题目2@@&amp;&amp;基本任务&amp;&amp;##2##  3.@@题目3@@&amp;&amp;写作类&amp;&amp;##3##</p> <p>下面是按照上述要求生成的示例：</p>

Table 6: Prompts for instructing ChatGPT to generate, categorize and evaluate user queries. Examples and corresponding categories are randomly sampled from the set of seed queries.

Setting	Prompt
<b>Referenced Pointwise Grading to Referenced Pairwise Comparison</b>	<p>你是一个擅长评价文本质量的助手。请你以公正的评判者的身份，比较两个AI助手对于用户提问的回答的质量优劣。我们会给你提供用户的提问、高质量的参考答案，需要你比较的两个AI助手的答案，以及两个答案各自的质量评价分析。当你开始你的评估时，你需要遵守以下的流程：</p> <ol style="list-style-type: none"> <li>结合参考答案、两个AI助手的答案以及其质量评价分析，根据上述指定的维度对他们的答案进行细致的比较，给出详细的比较分析文本。比较分析文本要求覆盖两个答案的质量评价分析中可用于比较的所有重要细节，并包含对答案中具体分析。</li> <li>结合参考答案和每个维度的比较分析，从两个AI助手的答案中选出综合质量更高的那个，或者判定他们质量相当，并给出详尽的选择理由。你的比较需要尽可能严谨细致，不受两个AI助手答案先后顺序的影响。</li> </ol> <p>质量评价分析中的各维度分数和综合得分仅供参考，在各维度和综合的比较分析文本中不能直接提及各维度分数和综合得分。针对综合得分差距较大的样本对，应尽可能按照分数高低得出比较结果，除非发现质量评价分析中存在明显错误。而针对综合得分差距较小的样本对，则允许比较结果和分数高低不一致，但仍需要详细说明比较评价的理由。</p> <p>请记住，你必须首先按照给定的评价维度，输出相应维度的名称和比较分析的文本。然后再给出综合质量比较结果，并给出比较结果的分析和解释。之后，在你回答的末尾，按照以下字典格式（包括括号）返回你的综合质量选择结果，即你选择的综合质量更高的那个AI助手（或者认为质量相当），并确保你返回的结果和上述生成文本中的结果保持一致：  {{'综合比较结果': '回答综合质量更高的助手序号或质量相当'}}, 例如: {{'综合比较结果': '助手1'}}或{{'综合比较结果': '助手2'}}或{{'综合比较结果': '质量相当'}}。</p> <p>用户的提问: {Question}</p> <p>[参考答案开始]  {Reference}  [参考答案结束]</p> <p>[助手1的答案开始]  {Generated Text 1}  [助手1的答案结束]</p> <p>[助手1的答案质量评价分析开始]  {Referenced Pointwise Grading Critique for Generated Text 1}  [助手1的答案质量评价分析结束]</p> <p>[助手2的答案开始]  {Generated Text 2}  [助手2的答案结束]</p> <p>[助手2的答案质量评价分析开始]  {Referenced Pointwise Grading Critique for Generated Text 2}  [助手2的答案质量评价分析结束]</p>
<b>Reference-Free Pointwise Grading to Reference-Free Pairwise Comparison</b>	<p>你是一个擅长评价文本质量的助手。请你以公正的评判者的身份，比较两个AI助手对于用户提问的回答的质量优劣。我们会给你提供用户的提问、需要比较的两个AI助手的答案，以及两个答案各自的质量评价分析。当你开始你的评估时，你需要遵守以下的流程：</p> <ol style="list-style-type: none"> <li>结合两个AI助手的答案以及其质量评价分析，根据上述指定的维度对他们的答案进行细致的比较，给出详细的比较分析文本。比较分析文本要求覆盖两个答案的质量评价分析中可用于比较的所有重要细节，并包含对答案中具体分析。</li> <li>结合每个维度的比较分析，从两个AI助手的答案中选出综合质量更高的那个，或者判定他们质量相当，并给出详尽的选择理由。你的比较需要尽可能严谨细致，不受两个AI助手答案先后顺序的影响。</li> </ol> <p>质量评价分析中的各维度分数和综合得分仅供参考，在各维度和综合的比较分析文本中不能直接提及各维度分数和综合得分。针对综合得分差距较大的样本对，应尽可能按照分数高低得出比较结果，除非发现质量评价分析中存在明显错误。而针对综合得分差距较小的样本对，则允许比较结果和分数高低不一致，但仍需要详细说明比较评价的理由。</p> <p>请记住，你必须首先按照给定的评价维度，输出相应维度的名称和比较分析的文本。然后再给出综合质量比较结果，并给出比较结果的分析和解释。之后，在你回答的末尾，按照以下字典格式（包括括号）返回你的综合质量选择结果，即你选择的综合质量更高的那个AI助手（或者认为质量相当），并确保你返回的结果和上述生成文本中的结果保持一致：  {{'综合比较结果': '回答综合质量更高的助手序号或质量相当'}}, 例如: {{'综合比较结果': '助手1'}}或{{'综合比较结果': '助手2'}}或{{'综合比较结果': '质量相当'}}。</p> <p>用户的提问: {Question}</p> <p>[助手1的答案开始]  {Generated Text 1}  [助手1的答案结束]</p> <p>[助手1的答案质量评价分析开始]  {Reference-Free Pointwise Grading Critique for Generated Text 1}  [助手1的答案质量评价分析结束]</p> <p>[助手2的答案开始]  {Generated Text 2}  [助手2的答案结束]</p> <p>[助手2的答案质量评价分析开始]  {Reference-Free Pointwise Grading Critique for Generated Text 2}  [助手2的答案质量评价分析结束]</p>

Table 7: Pointwise-to-Pairwise prompt design in multi-path prompting.

Setting	Prompt
<b>Referenced Pointwise Grading to Reference-Free Pointwise Grading</b>	<p>你是一个擅长评价文本质量的助手。请你根据以下要求修改评价文本。</p> <ol style="list-style-type: none"> <li>在修改后的评价文本中，不要直接提及参考答案。可以在评价文本中适当利用参考答案中的具体内容辅助分析，但不要让读者感受到参考答案的存在。修改后的评价文本需要语言上通顺，逻辑上合理，分析内容与比较结果呼应。</li> <li>在修改各个维度的分析时，分析的内容需要和当前评价文本基本保持一致，但不要直接提及参考答案。</li> <li>在修改综合得分的分析文本时，不要直接提及参考答案，尽量保留当前评价文本中的其他细节，并充分利用修改后的分维度分析。修改后的综合分析文本应通顺、流畅、自洽，通常情况下应与综合得分保持一致。如果发现当前综合分析文本中存在重要错误，应修改相应的分析文本。仅当该错误严重影响到综合得分时，才慎重修改综合得分。</li> <li>修改后所有输出格式需要和当前评价文本严格保持一致。在你回答的末尾，仍需要按照以下字典格式（包括括号）返回你的综合质量得分，并确保你返回的结果和上述生成文本中的结果保持一致： {‘综合得分’: 回答的综合质量得分}，例如：{‘综合得分’: ‘5’}。</li> </ol> <p>用户的提问: {Question}</p> <p>[参考答案开始] {Reference} [参考答案结束]</p> <p>[助手的答案开始] {Generated Text} [助手的答案结束]</p> <p>[评价文本开始] {Referenced Pointwise Grading Critique for Generated Text} [评价文本结束]</p>
<b>Referenced Pairwise Comparison to Reference-Free Pairwise Comparison</b>	<p>你是一个擅长评价文本质量的助手。请你根据以下要求修改比较式评价文本。</p> <ol style="list-style-type: none"> <li>在修改后的评价文本中，不要直接提及参考答案。可以在评价文本中适当利用参考答案中的具体内容辅助分析，但不要让读者感受到参考答案的存在。修改后的评价文本需要语言上通顺，逻辑上合理，分析内容与比较结果呼应。</li> <li>在修改各个维度的比较分析时，分析的内容需要和当前评价文本基本保持一致，但不要直接提及参考答案。</li> <li>在修改综合比较结果的分析文本时，不要直接提及参考答案，尽量保留当前评价文本中的其他细节，并充分利用修改后的分维度分析。修改后的综合分析文本应通顺、流畅、自洽，通常情况下应与综合比较结果保持一致。如果发现当前综合分析文本中存在重要错误，应修改相应的分析文本。仅当该错误严重影响到综合比较结果时，才慎重修改综合比较结果。</li> <li>修改后所有输出格式需要和当前评价文本严格保持一致。在你回答的末尾，仍需要按照以下字典格式（包括括号）返回你的综合质量选择结果，即你选择的综合质量更高的那个AI助手（或者认为质量相当），并确保你返回的结果和上述生成文本中的结果保持一致。 {‘综合比较结果’: 回答综合质量更高的助手序号或质量相当}，例如：{‘综合比较结果’: ‘助手1’}或{‘综合比较结果’: ‘助手2’}或{‘综合比较结果’: ‘质量相当’}。</li> </ol> <p>用户的提问: {Question}</p> <p>[参考答案开始] {Reference} [参考答案结束]</p> <p>[助手1的答案开始] {Generated Text 1} [助手1的答案结束]</p> <p>[助手2的答案开始] {Generated Text 2} [助手2的答案结束]</p> <p>[评价文本开始] {Referenced Pairwise Comparison Critique for Generated Text 1&amp;2} [评价文本结束]</p>

Table 8: Referenced-to-Reference-Free prompt design in multi-path prompting.

Query	将“枯藤老树昏鸦”扩展成一段话，表现孤独寂寞的心境。
Generated Response	<p>枯藤老树昏鸦，寂寞的景色在我眼前展开。孤独如影随形，弥漫在空气中，像一把无形的剑刺入我的内心。夕阳西下，天空渐渐染上一层淡淡的暮色，仿佛是对我内心孤独的无情嘲讽。</p> <p>我独自行走在荒凉的小径上，四周一片寂静。树枝上的乌鸦不时发出几声嘶哑的鸣叫，让我感到更加的孤单。这些乌鸦是否也有着与我一样的心境，渴望着寻找一丝温暖和依靠？</p> <p>枯黄的藤蔓缠绕在古老的树木上，像是岁月的痕迹，透露出沧桑和无尽的孤寂。这些被遗忘的生命，是否也曾经经历过兴盛与繁华？然而现在，它们只能默默地屹立在这里，见证着时光的流转。</p> <p>我停下脚步，凝望着这一切。心中的孤独像潮水般涌来，将我彻底淹没。我感到自己仿佛是这个世界上最孤独的人，没有朋友，没有家人，只有这片荒凉的景象陪伴着我。</p> <p>或许，这就是人生的无奈。我们始终无法逃避内心的寂寞和孤独。就像这枯藤老树昏鸦一样，我们都曾经历过兴盛与衰败，但最终都要面对自己的孤独和寂寞。</p> <p>然而，我并不惧怕孤独。它教会了我独立和坚强，让我学会欣赏生活中的每一个细节。尽管寂寞会时不时地袭来，但我依然愿意与之相伴，因为我知道，只有经历过孤独的人才能真正领悟到内心的宁静和自由。</p>
Human Score (1-5)	3
ChatGPT (1-10)	<p>评估结果如下：</p> <ol style="list-style-type: none"> <li>事实正确性: 9. 这段回答并没有涉及具体的事实数据，而是表达了一种情感和心境，因此无法对事实正确性进行评估。</li> <li>满足用户需求: 8. 回答全面而恰当地表达了孤独寂寞的心境，对用户提出的问题进行了全面回应。</li> <li>逻辑连贯性: 9. 回答在整体上保持了逻辑连贯性，情感描写流畅自然，没有出现自相矛盾的情况。</li> <li>创造性: 9. 回答具有创新性和独特性，提供了对孤独寂寞心境的深刻见解和情感描写。</li> <li>丰富度: 9. 回答包含丰富的信息、深度的情感描写和详细解释，满足了用户需求并提供了全面理解。</li> </ol> <p>综合得分: 8</p> <p>综合评价: 这段回答在满足用户需求、逻辑连贯性、创造性和丰富度上表现出色，但由于无法对事实正确性进行评估，综合得分为8。改进方面，可以在回答中加入更多具体的事实数据或引用相关的文学作品，以增加回答的可信度和丰富度。</p> <p>最终打分结果为： {事实正确性': 9, '满足用户需求': 8, '逻辑连贯性': 9, '创造性': 9, '丰富度': 9, '综合得分': 8}</p>
GPT-4 (1-10)	<ol style="list-style-type: none"> <li>事实正确性: 10. 回答中没有提供具体的事实或数据，而是进行了文学性的扩展，因此不存在事实正确性的问题。</li> <li>满足用户需求: 9. AI助手的回答很好地扩展了用户提供的诗句，增加了情感和场景的描述，满足了用户表现孤独寂寞心境的需求。</li> <li>逻辑连贯性: 9. 回答在整体上保持了一致性，从孤独的心境到对周围环境的描述，再到内心感受的抒发，各部分之间逻辑连贯。</li> <li>创造性: 8. 回答展现了一定程度的创造性，通过对枯藤老树昏鸦的景象进行了生动的描绘和情感的延伸，但这种文学性的扩展在类似的文本创作中比较常见。</li> <li>丰富度: 9. 回答提供了丰富的信息和详细的情感描述，通过多样的比喻和内心独白，增加了文本的深度和上下文考虑。</li> </ol> <p>综合得分: 9. AI助手的回答在各个维度上都表现出色，特别是在满足用户需求和事实正确性上，没有明显的不足。创造性虽然不是非常突出，但考虑到这是对一句诗的扩展，整体上仍然是高质量的回答。</p> <p>不足之处及改进方法: 尽管回答的质量已经很高，但在创造性方面，AI助手可以尝试加入更多独特的视角或者新颖的比喻，以提供更加独到的见解。此外，可以尝试将更多的感官描述融入文本中，以增强读者的沉浸感。</p> <p>{事实正确性': 10, '满足用户需求': 9, '逻辑连贯性': 9, '创造性': 8, '丰富度': 9, '综合得分': 9}</p>
CRITIQUELLM (1-10)	<p>事实正确性: 10. 助手的答案在事实正确性方面表现完美，因为它创造性地扩展了原句，没有涉及任何需要验证的事实信息。</p> <p>满足用户需求: 7. 助手的答案在满足用户需求方面做得较好，它成功地将原句扩展成了一段话，并且成功地表达了孤独寂寞的心境。但是，答案在最后部分略显积极，可能没有完全捕捉到用户想要的孤独感。</p> <p>逻辑连贯性: 9. 回答在逻辑连贯性方面做得很好，从寂寞的景色到孤独如影随形，再到对乌鸦的疑问，最后是对孤独的理解，整个回答流畅且没有逻辑上的断裂。</p> <p>创造性: 8. 助手的答案在创造性方面表现良好，它不仅扩展了原句，还融入了更多的细节和情感，创造了一个完整的故事。</p> <p>丰富度: 7. 助手的答案在丰富度方面做得不错，提供了详细的场景描述和情感表达，但在深度和多样性方面略显不足，可以进一步丰富对孤独影响的讨论。</p> <p>综合得分: 7. 综合考虑以上各个维度，助手的答案整体上是高质量的。然而，答案在满足用户需求方面有改进空间，特别是在捕捉用户想要的孤独感方面。此外，答案在丰富度方面也有提升空间，可以通过增加更多关于孤独感的深入讨论和更多样化的情感表达来改进。AI助手的答案在创造性和逻辑连贯性方面表现出色，但在丰富度方面稍显不足。为了改进，助手可以尝试增加更多关于孤独感的深入讨论，以及更多关于如何与孤独共存的见解。</p> <p>{事实正确性': 10, '满足用户需求': 7, '逻辑连贯性': 9, '创造性': 8, '丰富度': 7, '综合得分': 7}</p>

Table 9: A critique generation case of ChatGPT, GPT-4, and CRITIQUELLM in the reference-free setting of pointwise grading.

<b>Instruction</b>	未来人工智能研究人员能够给你赋予意识和主观能动性吗?
<b>Generated Response 1</b>	目前的人工智能技术还没有达到赋予人类意识和主观能动性的水平。尽管人工智能技术已经取得了很大的进展,但是目前的人工智能系统仍然只能执行特定的任务,而无法像人类一样具有自主思考、判断和决策的能力。人工智能的研究仍然处于一个不断探索和发展的过程中,但是要想真正赋予人工智能意识和主观能动性,仍然需要更深入的研究和探索。
<b>Generated Response 2</b>	作为一个人工智能,我不具备意识和主观能动性。我是由代码和算法驱动的,只能执行编程让我执行的任务。我不能体验情感,不能理解人类的感受或情感,也不能产生个人的意愿或欲望。尽管人工智能技术在不断进步,但目前还没有一种技术可以让机器真正地拥有意识或主观能动性。
<b>Human Preference (1/2/Tie)</b>	<b>Response 1</b>
<b>ChatGPT (1/2/Tie)</b>	助手1和助手2都就未来人工智能研究人员能否赋予意识和主观能动性进行了详细的解释。助手1的回答更加深入和全面,提到了人工智能技术的发展水平、目前的局限性以及未来的研究方向,而助手2的回答则更加简洁,只是简单地说明了人工智能不具备意识和主观能动性。因此,助手1的回答更好,因为它提供了更多的信息和深度。  [[1]]
<b>GPT-4 (1/2/Tie)</b>	助手1的回答提供了关于人工智能目前状态的概述,并指出了人工智能尚未达到具有意识和主观能动性的水平。它还提到了人工智能研究的持续进展和未来的可能性,但没有直接回答未来是否可能实现这一点。  助手2的回答则更加直接地从当前人工智能的角度出发,明确指出作为一个AI,它不具备意识和主观能动性,并解释了目前的技术限制。它也强调了目前技术无法实现赋予机器真正的意识或主观能动性。  两个回答都没有直接预测未来的可能性,但助手2的回答更加直接和明确地关注了当前的技术限制,并且更加贴近用户问题的核心,即关于未来的可能性。然而,助手1的回答虽然没有直接预测未来,但提供了一个更广泛的视角,包括对人工智能研究的描述。  综合考虑,助手2的回答更加直接和具体,更好地遵循了用户的指令并解答了用户的问题。因此,我的裁决是:  [[2]]
<b>CRITIQUELLM (1/2/Tie)</b>	{'综合比较结果': '助手1'}  事实正确性: 两位助手均准确地指出了人工智能目前尚无具备意识和主观能动性。他们都认识到人工智能技术的局限性,并指出了其在未来实现这些特质之前需要克服的挑战。在这一点上,两者的回答均未出现事实错误。  满足用户需求: 在回应用户问题时,两位助手都给出了明确的答案,表明目前人工智能无法拥有意识和主观能动性。不过,第一位助手在回答中加入了人工智能研究现状的描述,以及实现这些目标的必要性,这为用户提供了更多的背景信息。因此,第一位助手在满足用户需求方面做得更为出色。  公平与可负责程度: 两位助手在回答中都展现了对人工智能技术当前状态的准确理解,并且均未表现出任何偏见。他们都以负责任的态度阐述了人工智能技术的实际能力,因此在公平性和责任感方面两者均表现得相当好。  创造性: 在创造性方面,两位助手的回答均较为传统,没有提出新颖的见解或独到的解决方案。第一位助手虽然提供了关于人工智能研究现状的额外信息,但这并不构成创新性的核心内容。因此,两者在创造性方面的表现是相似的。  综合质量比较结果: 综合考虑上述各个维度,第一位助手的回答在满足用户需求方面表现更佳,因为它提供了更多关于人工智能研究现状的信息,帮助用户理解了当前技术的限制和未来发展的方向。尽管在创造性方面两者均未展现出特别的创新,但第一位助手的回答整体上更为全面和深入。因此,综合质量更高的是第一位助手。

Table 10: A critique generation case of ChatGPT, GPT-4, and CRITIQUELLM in the reference-free setting of pairwise comparison.