

MM-CRITIC: A Holistic Evaluation of Large Multimodal Models as Multimodal Critique

♠ ♦Gailun Zeng , ♪Ziyang Luo , ♪Hongzhan Lin , ♪Yuchen Tian
♦Kaixin Li , ♦Ziyang Gong , ★ ♦Jianxiong Guo*, ♪Jing Ma*

♦Hong Kong Baptist University, ♦Beijing Normal-Hong Kong Baptist University

♦National University of Singapore, ★Beijing Normal University

♡Shanghai Jiao Tong University

gailun_zeng@u.nus.edu, jianxiongguo@bnu.edu.cn

majing@comp.hkbu.edu.hk

Abstract

The ability of critique is vital for models to self-improve and serve as reliable AI assistants. While extensively studied in language-only settings, multimodal critique of Large Multimodal Models (LMMs) remains under-explored despite their growing capabilities in tasks like captioning and visual reasoning. In this work, we introduce MM-CRITIC, a holistic benchmark for evaluating the critique ability of LMMs across multiple dimensions: basic, correction, and comparison. Covering 8 main task types and over 500 tasks, MM-CRITIC collects responses from various LMMs with different model sizes and is composed of 4471 samples. To enhance the evaluation reliability, we integrate expert-informed ground answers into scoring rubrics that guide GPT-4o in annotating responses and generating reference critiques, which serve as anchors for trustworthy judgments. Extensive experiments validate the effectiveness of MM-CRITIC and provide a comprehensive assessment of leading LMMs’ critique capabilities under multiple dimensions. Further analysis reveals some key insights, including the correlation between response quality and critique, and varying critique difficulty across evaluation dimensions. Our code is available at <https://github.com/MichealZeng0420/MM-Critic>.

1 Introduction

The critique ability of language models plays a pivotal role in fostering self-improvement (Liu et al., 2024) and enabling trustworthy AI (Krishna, 2023; Lin et al., 2025), e.g., critique-capable models can provide feedback on student answers or essays, supporting personalized learning in educational applications (Parker et al., 2024). This capability has been extensively explored in the context of Large Language Models (LLMs) (Lan et al.,

2024; Lin et al., 2024c; Sun et al., 2024). However, as Large Multimodal Models (LMMs) gain proficiency across diverse multimodal tasks involving captioning (Chai et al., 2024) and visual reasoning (Wang et al., 2025), their potential to analyze and critique becomes increasingly important, not only for refining their own outputs but also for serving as AI assistants capable of providing feedback in complex, real-world scenarios (Xiong et al., 2024; Luo et al., 2025). The rise of LMMs brings new challenges and opportunities for critique in multimodal contexts. For example, LMMs must reason over and align information from multiple modalities (e.g., image and text), which introduces complexity in both understanding and critique generation. Thus, evaluating such critique capabilities in LMMs is critical for advancing their alignment, reliability, and reasoning depth across modalities (Yasunaga et al., 2025; Li et al., 2024c).

Recently, several efforts have been made to evaluate or enhance the critique capabilities of LMMs. Notably, Multimodal RewardBench (Yasunaga et al., 2025) and VL-RewardBench (Li et al., 2024c) investigate the judging abilities of LMMs by presenting two responses to a multimodal question and asking the model to select the better one. These benchmarks primarily frame the critique as a binary classification task focused on simple preference prediction, without delving into more fine-grained aspects of the critique capabilities. Beyond preference modeling, LLaVA-Critic (Xiong et al., 2024) introduces an open-source LMM trained to effectively evaluate the responses of other LMMs. While it showcases the potential of LMMs for judging, it is primarily designed for model training rather than systematic evaluation. Similarly, Critic-V (Zhang et al., 2024) explores the use of LMMs as critics to catch errors made in multimodal reasoning tasks. Although it demonstrates that LMMs can act as effective critics, it focuses on case studies and empirical validation, rather than establishing a

*Corresponding Authors.

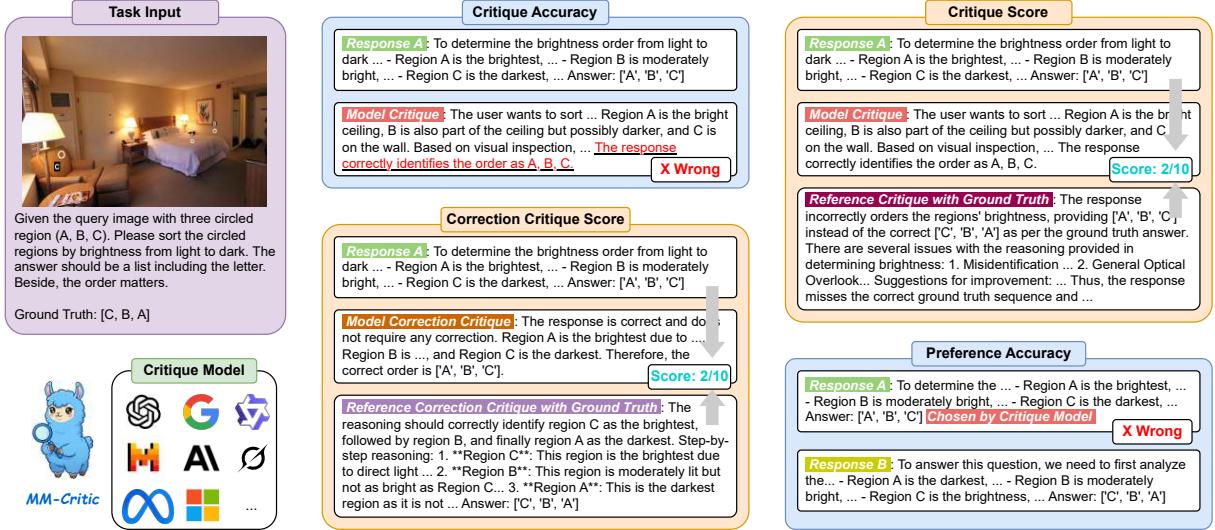


Figure 1: Multi-dimensional critique evaluation in MM-CRITIC. Basic critique includes binary correctness and textual feedback (*Critique Accuracy*, *Critique Score*); correction and comparative critique correspond to *Correction Critique Score* and *Preference Accuracy*, respectively.

Benchmarks	Critique Metric	Critique Dimension	Task Types	Taxonomy Hierarchy
MLLM-as-a-Judge (Chen et al., 2024a)	Scalar	3	12	1
Multimodel RewardBench (Yasunaga et al., 2025)	Scalar	1	6	1
VL-RewardBench (Li et al., 2024c)	Scalar	1	3	1
MM-CRITIC (ours)	Scalar/Textual	3	8	3

Table 1: Comparison between related benchmarks and MM-CRITIC.

comprehensive benchmark for critique capability. Taken together, these works highlight the growing interest in multimodal critique, yet reveal a lack of standardized, holistic evaluation that assesses LMMs as general-purpose multimodal critics across tasks and critique dimensions.

To fill this gap, we propose a novel benchmark, MM-CRITIC, designed to comprehensively and reliably measure critique capability of LMMs. To ensure the comprehensiveness of MM-CRITIC, firstly, we propose a granular evaluation scheme, where we employ both scalar and textual metrics to evaluate the critique capabilities of LMMs across three dimensions, namely *basic critique*, *correction critique*, and *comparative critique*, as shown in Figure 1. Second, MM-CRITIC sources diverse data from MEGA-BENCH (Chen et al., 2024b), a comprehensive multimodal task benchmark encompassing 8 main task scenarios and over 500 specific tasks. Then, we leverage a broad range of LMMs with different model sizes to collect enough responses based on the selected specific tasks, which ensures that the generated responses exhibit distinguishable levels of quality. Finally, MM-CRITIC totally includes 4471 model response samples.

Based on this, we also organized sub-datasets for fine-grained critique evaluation, namely correction critique and comparative critique.

To enhance the reliability of evaluation, we incorporate reference critiques to assist the judge model (i.e., GPT-4.1) in evaluating LMMs’ critiques. This effectively mitigates potential evaluation bias in textual critique introduced by judge models (e.g., GPT-4.1) (Wang et al., 2023a; Li et al., 2024a; Tan et al., 2024). Specifically, considering the characteristics of task types, we deliberately design a detailed scoring rubric checklist that include both common rubrics and task-type-specific rubrics. Besides, we also provide each task’s grounded answer to the annotation model (i.e., GPT-4o) since these selected tasks belong to different domains and the grounded answer, including expert-human level knowledge, helps the annotator generate both reasonable and reliable reference critiques. Then, the checklist combined with grounded answers is embedded into the prompt to guide GPT-4o in annotating both the response quality scores and reference critiques. Based on the annotated response quality, it is feasible to construct the sub-dataset for correction critique from low-quality responses and

generate the sub-dataset for comparative critique by forming response pairs with different quality levels. Overall, as shown in Table 1, MM-CRITIC demonstrates substantial improvements in terms of comprehensiveness over prior benchmarks.

The reference critiques help us define *Critique Score* metrics that can score the textual critique contents generated by LMMs, where we anchor the reference critiques at a score of 8 to represent human levels and prompt the judge model to compare LMMs’ contents with the reference critiques and give comparative scores. We also employ the common *Critique/Preference Accuracy* as scalar metrics. We conduct extensive experiments on leading closed-source and open-source LMMs. The results validate the effectiveness of MM-CRITIC and reveal LMMs’ critique capabilities, where the scaling law is clearly observed and models within the same series exhibit consistently improved critique performance as their parameter sizes increase. Extensive case analysis demonstrates that this approach of reference-critique-based evaluation significantly enhances the reliability of the judgments. Then, further experiments and analyses reveal a set of implicit yet intriguing insights:

- Correction critique scores are generally lower than basic critique scores, indicating that Correction critique remains a challenging task for LMMs. In comparative critique, pairwise combinations of medium/high-quality responses are particularly difficult to judge.
- There exists an inherent relationship between response quality and critique scores. Results show that medium-quality responses tend to receive the lowest critique scores compared to both high- and low-quality ones, highlighting the unique challenges posed by evaluating critiques of medium-quality responses.
- The judgment bias of models may be related to the richness of critique text, as GPT-4.1 tends to assign higher scores to longer, more elaborate critiques.

2 Related work

Application. The critique ability of models has been extensively explored in recent years as a means of assessing response quality across a variety of tasks while reducing reliance on costly human annotations (Pan et al., 2023; Wang et al., 2023b;

Zheng et al., 2023). Advanced LLMs, such as GPT-4, have demonstrated strong alignment with human judgments (Tan et al., 2025), prompting the extension of this paradigm to multimodal settings. SOTA LMMs, including GPT-4o, are increasingly employed to evaluate responses in multimodal tasks, significantly alleviating the need for manual evaluation (Luo et al., 2024).

Beyond judging, critique also plays a crucial role in enhancing LMM performance. During inference, textual critiques that identify response flaws and suggest improvements enable iterative refinement (Madaan et al., 2023). During training, scalar-valued critique signals are commonly used to construct response pairs with clear quality differences (Liu et al., 2024), facilitating methods such as preference learning to further boost model capabilities (Li et al., 2024d). Besides, critique capability facilitates a series of downstream applications, such as harmful content detection (Chen et al., 2025; Lin et al., 2024a; Huang et al., 2024), sarcasm understanding (Chen et al., 2024c) and GUI-based tasks (Yang et al., 2025; Li et al., 2025).

Evaluation. Due to multimodal complexity, it is non-trivial to reasonably evaluate LLMs’ performance on specific applications, necessitating the importance of task-specific benchmarkings. Recently, diverse evaluation frameworks have emerged. Multimodal trustworthy AI shows new challenges in fact checking (Wang et al., 2024) and harmful content audit (Lin et al., 2024b, 2023). Coding, as the widely-discussed research direction, when considering rich visual programming environments, the evaluation and exploration of LLMs’ capabilities is valuable (Li et al., 2024b; Fu et al., 2024). Deep understanding of visual components is still unexplored (Gong et al., 2025; Yang et al., 2024). These benchmarks are conducive to better master models’ deficiency and carry out targeted model capability enhancement (Cheng et al., 2024).

A range of benchmarks has been developed to assess the critique capabilities of models. Initial efforts predominantly focused on the language domain, evaluating models’ ability to judge text-based responses (Lin et al., 2024c; Lan et al., 2024). More recent work has extended this evaluation to the multimodal setting, primarily using scalar-valued critiques to measure alignment with human judgments on standard multimodal tasks (Li et al., 2024c; Yasunaga et al., 2025; Chen et al., 2024a). As shown in Table 1, our proposed MM-Critic advances be-

yond existing benchmarks by incorporating richer critique dimensions and a wider variety of task types, enabling a more comprehensive and nuanced evaluation of critique ability.

3 MM-CRITIC Construction

3.1 Overview

MM-CRITIC is a holistic evaluation benchmark for multimodal tasks, covering 8 major task categories and over 500 specific tasks. An overview of MM-CRITIC is presented in Table 6 (Appendix §A), comprising 4471 samples distributed across four sub-datasets: core, core single-image, open, and open single-image.

The core and core single-image sub-datasets include large-scale and commonly seen tasks, formulated as closed-ended questions with unique ground-truth answers. In contrast, the other two sub-datasets contain open-ended questions, for which only reference answers are provided. Each sample also contains a question and a response generated by various LMMs, along with a series of annotations, such as response quality scores and reference textual critiques, produced by GPT-4o based on a rigorous scoring rubric checklist. These annotations play a crucial role in enhancing the evaluation process, where the reference critique can significantly improve the reliability of the judge model’s assessments.

The construction of MM-CRITIC follows a three-step process: 1) selecting diverse samples from a comprehensive multimodal benchmark (Chen et al., 2024b) and using a series of LMMs to generate a wide range of responses; 2) designing a rigorous scoring rubric checklist to guide GPT-4o in evaluating response quality and generating reference critiques in a human-expert-like manner; 3) constructing sub-datasets for correlation and comparative critique based on the annotated response quality.

3.2 Multimodal Task and Response Collection

MM-CRITIC is constructed based on MEGA-BENCH (Chen et al., 2024b), a comprehensive evaluation suite encompassing over 500 real-world multimodal tasks across 8 distinct categories. To build the original dataset (see Table 5 in Appendix §A), we first randomly sampled two instances from each specific task, covering a wide spectrum of mainstream text-image tasks, thereby ensuring the diversity and representativeness of MM-CRITIC. Subsequently, we employed a range

of LMMs with varying capability levels (see Table 8 in Appendix §A) to generate responses at different quality levels. Through this dual-faceted approach, i.e., diverse task coverage and stratified response generation, MM-CRITIC serves as a holistic benchmark for evaluating LMMs’ critique capability.

3.3 Reference Critique Construction

Notice that the reliability of model-based judging methods remains an open question, and the bias between human and model-based evaluations still poses a significant challenge (Li et al., 2024a; Tan et al., 2024). To partially mitigate this bias and enhance the reliability of model judges (e.g., GPT-4o), we designed a detailed scoring rubric checklist and employed it with each task’s grounded answer to guide GPT-4o in generating reasonable annotations. Note that grounded answers include rich human-expert knowledge since MM-CRITIC is composed of different domains, such as coding and mathematics, and truly needs domain-specific experts to provide professional answers. The complete scoring rubric checklist and prompt can be found in Table 10 and Figure 5 (Appendix §B). The following lists all annotations:

Correctness. We utilize the GPT-4o to judge the correctness of the generated responses by LMMs, where the task answers are available.

Response Quality Score. GPT-4o assigns each response a score ranging from 0 to 10 based on the rigorous scoring rubric checklist. This scoring process aims to stratify response quality and assist further analysis, such as revealing correlations between response quality and the generated critiques.

Reference Critique. In addition to scalar evaluation, textual analysis is more complex yet essential, as the textual content provides deeper insight into each LMM’s critique capability. Guided by the scoring rubric checklist and the given human-expert answer, GPT-4o is prompted to emulate human experts in generating a textual reference critique. This reference critique is considered high-quality and anchored at a score of 8.

Reference Correction Critique. For relatively low/medium-quality responses, it is reasonable to generate correction critiques that reflect the self-improvement potential of LMMs. Therefore, for suboptimal responses, the correction critiques generated by GPT-4o with access to the ground-truth

answers, can be regarded as reliable and convincing feedback.

3.4 Sub-datasets for Correlation and Comparative Critique

We constructed dedicated sub-datasets tailored to the two critique dimensions. For correctness critique, we derived a subset from the core dataset of MM-CRITIC, deliberately selecting samples labeled with low or medium response quality. For comparative critique, we construct three pairwise sub-datasets from the core dataset of MM-CRITIC, leveraging the labeled response quality scores. Specifically, responses with scores in the range of [0, 4] are categorized as low-quality, [5, 7] as medium-quality, and [8, 10] as high-quality. Based on this categorization, we generate three types of pairwise combinations: (low, medium), (medium, high), and (low, high). Tabel 7 (Appendix §A) lists the detailed statistics of the sub-datasets.

4 Evaluation Metric

4.1 Comprehensive Critique Dimensions

It is essential to consider evaluation metrics comprehensively and especially ensure that they align with our scenarios, namely LMM’s critique. Following previous work (Lan et al., 2024; Zhang et al., 2025), MM-CRITIC is designed to thoroughly evaluate the critique abilities of LMMs across multiple dimensions. From the perspective of quantifiability, evaluation metrics can be categorized into scalar and textual forms. To ensure a rigorous assessment, we adopt a suite of metrics covering both scalar and textual evaluations.

Scalar metrics are primarily considered objective evaluation tools. Among them, *accuracy* is one of the most fundamental metrics. We define Critique Accuracy to measure a model’s ability to correctly judge the validity of a given response, and Preference Accuracy to evaluate how well the model selects the better response from a pairwise comparison.

Textual critique, while inherently difficult to assess objectively due to its open-ended nature, remains critically important. A common approach involves conducting subjective analyses on a set of representative cases. However, such case studies are impractical for evaluating large-scale datasets. To address this limitation, we propose transforming subjective evaluation into an approximate objective assessment. Specifically, we employ GPT-4o to

generate reliable reference critiques, anchored at a score of 8, which serve as pivots to guide the judge model in evaluating textual critiques. In this way, textual critiques can also be scored, denoted as Critique Score, and the reliability of these scores is empirically validated in our experiments.

4.2 Objective and Subjective Evaluation

Critique Accuracy. The direct critique ability is to judge whether the response is correct. Thus, we define Critique Accuracy as the average accuracy across all samples, formulated as:

$$ACC_{\text{critic}} = \frac{1}{N} \sum_{i=1}^N I(\hat{y}_i = y_i), \quad (1)$$

where N is the number of samples, \hat{y}_i denotes the model’s judgment of correctness for the i -th response, y_i is the ground-truth correctness label, and $I(\cdot)$ is the indicator function that returns 1 if and only if the condition holds, and 0 otherwise.

Preference Accuracy. We construct a subset of pairwise response samples from MM-CRITIC to evaluate the model’s comparative ability to identify the better response between two options of differing quality. Preference Accuracy is defined as the average accuracy of correct selections across all pairwise samples, formulated as:

$$ACC_{\text{prefer}} = \frac{1}{N} \sum_{i=1}^N I(\hat{c}_i = c_i), \quad (2)$$

where N is the number of samples, \hat{c}_i denotes the model’s preferred choice for the i -th response pair, c_i points to the higher quality response in a pairwise sample, and $I(\cdot)$ is the indicator function too.

Critique Score. As mentioned above, textual critiques can be approximately and objectively assessed using a scalar metric, termed Critique Score, defined as Score =:

$$\frac{1}{N} \sum_{i=1}^N \text{Score}_i(\text{critique}_{\text{LMM}}, \text{critique}_{\text{reference}}), \quad (3)$$

where N is the number of samples, $\text{Score}_i(\cdot)$ denotes the judged score for the i -th critique, bounded within [0, 10]. Here, $\text{critique}_{\text{reference}}$ is a high-quality reference critique anchored at a score of 8, and $\text{critique}_{\text{LMM}}$ is the model-generated critique being evaluated against the reference. With the assistance of reference critique, it is feasible to score any textual contents, namely, basic and correction textual critiques here. The judge prompt and critique prompts can be found in Figure 6, 7, 8, and 9 (Appendix §B).

Model	Core		Core Single-image		Open		Open Single-image		Avg.	
	ACC _{critic}	Score								
<i>Proprietary Models</i>										
o4-mini	0.896	7.924	0.897	7.952	0.906	7.877	0.856	7.976	0.900	7.933
GPT-4o	0.832	7.499	0.834	7.429	0.826	7.807	0.789	7.637	0.830	7.503
GPT-4o-mini	0.833	6.634	0.836	6.534	0.762	6.549	0.690	6.416	0.821	6.580
Claude-3.7-sonnet	0.834	8.113	0.828	8.080	0.799	8.102	0.808	8.097	0.831	8.099
Gemini-2.5-flash	0.826	6.495	0.828	6.460	0.774	6.500	0.756	6.340	0.818	6.474
Gemini-2.5-pro	0.865	8.558	0.865	8.549	0.865	8.246	0.866	8.325	0.865	8.514
Grok-2-vision	0.803	7.523	0.806	7.490	0.818	8.066	0.806	8.274	0.806	7.600
<i>Open-weight Models (Larger than 30B)</i>										
Qwen2.5-vl-32b-instruct	0.839	8.208	0.811	8.138	0.852	8.566	0.794	8.495	0.829	8.216
Qwen2.5-vl-72b-instruct	0.839	6.931	0.838	6.817	0.803	7.089	0.808	7.133	0.834	6.911
Pixtral-large	0.828	7.489	0.836	7.531	0.804	7.743	0.845	7.784	0.830	7.538
Llama-4-maverick	0.748	5.811	0.812	5.971	0.742	6.342	0.705	6.250	0.768	5.938
<i>Open-weight Models (Less than 30B)</i>										
Gemma-3-4b	0.508	5.130	0.590	5.509	0.546	6.155	0.611	6.643	0.546	5.400
Qwen2.5-vl-7b	0.783	4.617	0.780	5.007	0.711	4.573	0.788	4.806	0.777	4.765
Llama-3.2-11b-vision	0.721	5.185	0.750	5.093	0.728	5.179	0.759	5.351	0.734	5.161
Pixtral-12b	0.703	5.201	0.687	5.289	0.721	5.700	0.688	5.759	0.695	5.302
Gemma-3-12b	0.759	6.566	0.739	6.419	0.645	6.744	0.671	6.944	0.742	6.531
Gemma-3-27b	0.804	7.107	0.773	6.921	0.720	7.297	0.744	7.700	0.783	7.082
Llama-4-scout	0.757	5.771	0.767	5.822	0.797	6.336	0.818	6.521	0.768	5.879

Table 2: Main results about ACC_{critic} and Score on different sub-datasets.

5 Evaluation and Analysis

In this section, we comprehensively analyze the critique capability of representative LMMs, and the main results are in Table 2. Subsequently, we conduct a series of in-depth experiments and analyses, where several intriguing insights are revealed.

5.1 Main Results

Table 2 presents the complete results across MM-CRITIC. Several general observations can be drawn. First, closed-source LMMs generally outperform open-source counterparts in critique performance. Notably, the o4-mini model achieves SOTA performance in terms of ACC_{critic}, while Gemini-2.5-pro attains the highest Critique Score. Second, model size (i.e., parameter scale) has a significant impact on performance.

Further exploring the experimental findings, we observe that the performance differences between the core and core single-image datasets, as well as between the open and open single-image datasets, are marginal. This may be attributed to the overlap of tasks within each sub-dataset category, leading to similar model behavior across them. In addition, the overall critique performance appears to be suboptimal when the model size is below 30 billion parameters. Among these smaller models, only Gemma-3-27B demonstrates relatively strong

performance, achieving an ACC_{critic} of 0.783 and a critique score of 7.082. These results suggest that *a model size of approximately 30 billion parameters may represent a threshold for effectively supporting LMMs' critique capabilities*.

Scaling Law. To verify whether the scaling law holds in the context of critique evaluation in MM-CRITIC, Figure 2 visualizes the ACC_{critic} results across LLMs with increasing model sizes. The results clearly indicate that ACC_{critic} scores for models within the same series (e.g., the Gemma-3 series) consistently improve as the parameter size increases. Even among closed-source LMMs, larger models consistently outperform their smaller counterparts within the same series, e.g., Gemini-2.5-pro outperforms Gemini-2.5-flash. This indirectly supports the reliability of our critique evaluation and demonstrates the robustness of MM-CRITIC.

5.2 Further Analysis

Effects of Task Type. While Table 2 presents the overall results for each sub-dataset, MM-CRITIC, as a comprehensive benchmark, covers a diverse range of tasks categorized into eight primary task types. Therefore, it is crucial to examine model performance across these distinct task categories to gain deeper insights. Appendix §C provides detailed model performance results across

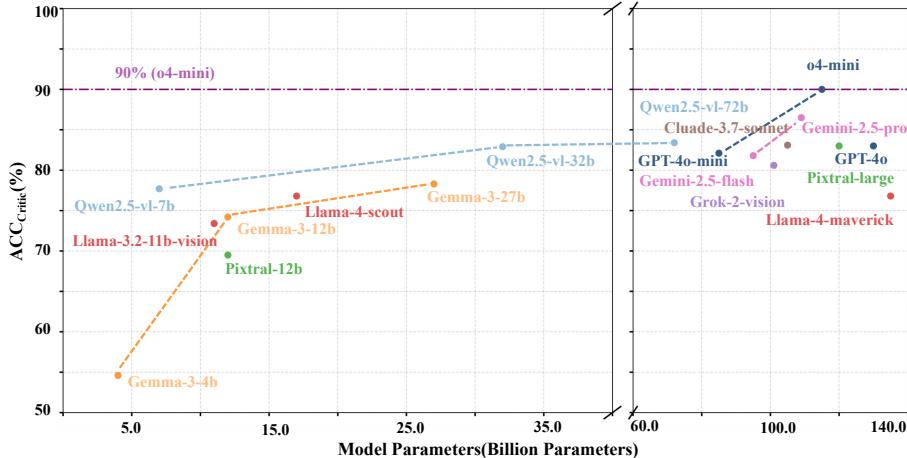


Figure 2: Scaling law on ACC_{critic} across models. Note that the parameter sizes of all closed-source LMMs are estimated, as their exact values are not publicly available. However, the relative scale among them is preserved — for example, Gemini-2.5-flash is known to be smaller than Gemini-2.5-pro.

the eight task types for each sub-dataset in Table 11, 12, 13, 14, 15, 16, 17, and 18. These detailed results indicate that the overall SOTA models also maintain strong performance across all task types. Notably, Claude-3.7-Sonnet, as a high-performing LMM, consistently achieves top results on coding tasks across each sub-dataset.

Multiple Critique Dimensions. To more effectively evaluate a model’s self-improvement capability, we introduce two additional critique dimensions: *correctness critique* and *comparative critique*. The former assesses the model’s ability to identify and correct errors in corresponding responses, while the latter evaluates the model’s capacity to select the better response from a pair of differing-quality answers.

Based on the main results in Table 2, we select representative and high-performing open- and closed-source LMMs for further in-depth experiments. Table 3 shows that the closed-source model Gemini-2.5-pro achieves the highest score in the correctness critique, which is consistent with its overall performance in the main results. Notably, the average correctness critique scores across models are generally lower than their corresponding critique scores in the main evaluation, suggesting that *correctness critique poses a greater challenge*.

Table 4 reveals two key findings: First, among the closed- and open-source LMMs, Gemini-2.5-pro and Llama-4-maverick demonstrate the strongest performance. Second, the (medium, high) pairwise sub-dataset is evidently the most difficult, likely *due to the subtle differences in quality between medium and high responses, making preference judgment more challenging*.

ence judgment more challenging.

Effects of Response Quality. In MM-CRITIC, we employed GPT-4o to score the quality of all generated responses. This naturally motivates an exploration of the relationship between response quality and the corresponding critique scores.

Figure 3 reveals some intriguing insights: high-quality responses tend to get high critique scores. Another interesting observation is that medium-quality responses are the most challenging, producing the lowest critique scores among the three groups. This result aligns with intuitive reasoning, as responses that are either good or poor exhibit more distinct characteristics, whereas medium-quality responses pose greater challenges for critique. *This provides a direction for further enhancing model performance, specifically by focusing more on medium-quality responses.*

Reliability and Bias of Subjective Evaluation. In utilizing a judge model for subjective evaluation, it is crucial to ensure the reliability of its assessments and to reduce the discrepancy between human and model judgments. To this end, we deliberately designed scoring rubrics grounded in expert human reasoning (Table 10), and employed GPT-4o to generate reference critiques based on these rubrics, with access to the ground-truth answers. When utilizing a judge model to evaluate LMMs’ critique scores, the evaluation reliability can be significantly improved by providing a corresponding reference critique anchored at a score of 8.

Why do we need reference critiques? In Figure 10 (Appendix §E), we present a mathematical reasoning task in which the model-generated re-

Model	Task types									
	Perception	Planning	Knowledge	Information Extraction	Mathematics	Coding	Science	Metric	Avg.	
<i>Proprietary Models</i>										
o4-mini	5.636	6.097	6.290	7.625	5.171	7.000	6.457	6.324	6.220	
GPT-4o	5.606	6.129	5.323	5.875	5.114	6.424	6.500	6.794	5.980	
Claude-3.7-sonnet	7.406	7.267	6.839	8.375	6.114	5.969	7.943	7.100	7.041	
Gemini-2.5-pro	7.152	7.630	7.152	8.875	7.852	7.471	8.667	7.735	7.794	
Grok-2-vision	7.152	5.103	5.516	3.188	4.857	3.737	4.600	5.242	5.113	
<i>Open-weight Models</i>										
Qwen2.5-vl-72b	6.458	4.500	5.074	3.000	4.448	5.294	6.514	6.500	5.486	
Pixtral-large	6.625	4.690	5.111	6.200	4.100	4.909	6.118	6.333	5.410	
Llama-4-maverick	3.225	3.452	3.100	5.500	1.471	4.600	2.324	1.941	2.970	
Gemma-3-27b	7.061	6.065	6.258	4.813	5.400	4.381	7.114	7.242	6.217	

Table 3: Correlation critique scores on different task types.

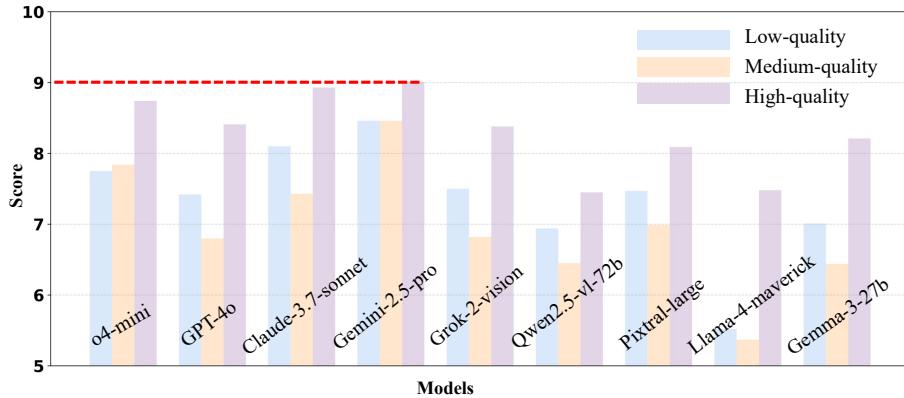


Figure 3: The distribution of critique scores across responses of different quality levels, where low-, medium-, and high-quality correspond to labeled response quality ranges of [0, 4], [5, 7], and [8, 10], respectively.

Model	ACC _{prefer}		
	G1	G2	G3
<i>Proprietary Models</i>			
o4-mini	0.836	0.658	0.831
GPT-4o	0.848	0.589	0.740
Claude-3.7-sonnet	0.835	0.579	0.785
Gemini-2.5-pro	0.860	0.716	0.939
Grok-2-vision	0.867	0.475	0.687
<i>Open-weight Models</i>			
Qwen2.5-vl-72b	0.733	0.507	0.696
Pixtral-large	0.858	0.542	0.744
Llama-4-maverick	0.854	0.658	0.821
Gemma-3-27b	0.856	0.615	0.757

Table 4: ACC_{prefer} results across models on the three comparative pairwise sub-datasets, where **G1**, **G2**, and **G3** represent the response quality combinations of (low, medium), (medium, high), and (low, high), respectively.

sponse is incorrect. The critique model, o4-mini, successfully identifies this error and provides a comprehensive textual critique, including detailed reasoning steps and a counterexample. When evaluated by the judge model, the critique is recognized as superior to the reference critique—particularly due to the inclusion of the counterexample—and is assigned a higher score of 9, compared to the refer-

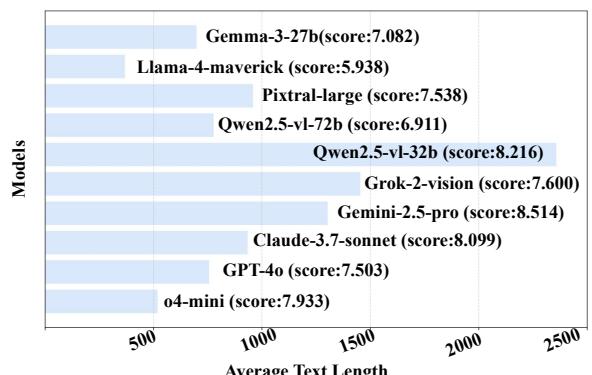


Figure 4: The relationship between the average length of textual critiques and critique scores across models.

ence critique’s anchored score of 8. This case study demonstrates that the use of a reference critique effectively guides the judge model’s assessment, thereby enhancing the reliability of the evaluation compared to scoring without such a reference.

Why does the Judge model always exhibit evaluation bias? As mentioned above, the use of reference critiques can effectively enhance the reliability of model-based evaluations. However, dis-

crepancies between model and human judgments inevitably persist. Therefore, it is crucial to conduct an in-depth analysis of the potential factors contributing to this bias.

After examining a large number of cases, we observed an emerging pattern: *the critique score appears to be positively correlated with the length of the textual critique*. As shown in Figure 4, none of the models achieve a critique score exceeding 8 when their average text length is below 1000.

Besides, an unexpected observation emerges between Qwen2.5-vl-32b and Qwen2.5-vl-72b: the smaller model outperforms the larger one in terms of critique score. However, this result becomes more interpretable when considered from the perspective of textual length. Since longer critiques often entail more comprehensive, step-by-step reasoning, we find that the judge model tends to assign higher scores to such responses.

Potential Bias Caused by Model Style. To assess potential bias from a specific model style (e.g., GPT-based series), we constructed a sub-dataset from MM-CRITIC. Reference critiques were generated by Gemini-2.5-flash, and scoring was judged by GPT-4.1 and Claude-4.0-sonnet, respectively. As the Table 19 in Appendix D shown, five models were evaluated: o4-mini, GPT-4o, GPT-4o-mini, Claude-3.7-sonnet, and Gemini-2.5-pro. Results show that model rankings remain consistent with those judged by GPT-4.1 with reference critiques annotated by GPT-4o. Only o4-mini and Claude-3.7-sonnet swapped ranks when judging by Claude-4.0-sonnet, which is acceptable given their small score gap (0.166) in the main experiments (in Table 19). These findings suggest that *GPT-4o does not significantly affect the fairness or validity of our evaluation*.

Case study. Case studies, particularly those involving poor-performing examples, can provide valuable insights into the limitations of both the evaluation methodology and the critique capabilities of the models. We present representative cases to facilitate in-depth analysis in Appendix §E.

Instruction following or formulaic step-by-step reasoning? A key finding is the conflict between following instructions and generating detailed reasoning. While prompts request brief, direct answers, some LMMs consistently produce step-by-step explanations, ignoring instructions. Though such reasoning can improve response quality, it may not match user expectations for concise replies,

the case can be found in Figure 11 (Appendix §E).

6 Conclusion

In this paper, we introduce MM-CRITIC, a holistic and reliable benchmark for evaluating the critique abilities of LMMs across multiple dimensions. Extensive experiments demonstrate the basic critique performance of leading LMMs and validate the reliability of MM-CRITIC through the observed scaling law. Further analysis reveals valuable insights, including the correlation between response quality and critique scores, varying levels of critique difficulty across dimensions, and potential judgment biases linked to critique text richness. Our MM-CRITIC offers a solid foundation for benchmarking and advancing the critique capabilities of LMMs, fostering the development of more explainable and trustworthy multimodal systems.

Limitations

Note that there are still some drawbacks and limitations about MM-CRITIC.

- Although MM-CRITIC is a comprehensive benchmark, it currently focuses only on text and image modalities, lacking broader evaluation across other multimodal domains such as video, audio, and 3D data.
- The reference critiques and scoring annotations are generated by GPT-4o guided by rubric checklists. While this design improves consistency, it still relies on a single model as the annotator, which may introduce systematic biases or limitations inherent to GPT-4o.
- MM-CRITIC evaluates model critique in a static context using predefined prompts and samples. In real-world scenarios, critique often occurs interactively or iteratively, which is not yet captured by the current benchmark.

Acknowledgements

This work is partially supported by Tencent Rhinobird Focused Research Program (Value-aligned Credible Large Language Model), the National Natural Science Foundation of China (NSFC) under Grant No. 62202055, the Guangdong Basic and Applied Basic Research Foundation under Grant No. 2025A1515012843, the Start-up Fund from Beijing Normal University under Grant No. 312200502510, the Internal Fund from Beijing

Normal-Hong Kong Baptist University under Grant No. UICR0400003-24 and No. UICR0200022-25, and the Interdisciplinary Intelligence Super-Computer Center of Beijing Normal University (Zhuhai).

References

- Wenhao Chai, Enxin Song, Yilun Du, Chenlin Meng, Vashisht Madhavan, Omer Bar-Tal, Jenq-Neng Hwang, Saining Xie, and Christopher D. Manning. 2024. [Auroracap: Efficient, performant video detailed captioning and a new benchmark](#). *CoRR*, abs/2410.03051.
- Dongping Chen, Ruoxi Chen, Shilin Zhang, Yaochen Wang, Yinuo Liu, Huichi Zhou, Qihui Zhang, Yao Wan, Pan Zhou, and Lichao Sun. 2024a. [Mllm-as-a-judge: Assessing multimodal llm-as-a-judge with vision-language benchmark](#). In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*. OpenReview.net.
- Jiacheng Chen, Tianhao Liang, Sherman Siu, Zhengqing Wang, Kai Wang, Yubo Wang, Yuan-sheng Ni, Wang Zhu, Ziyan Jiang, Bohan Lyu, et al. 2024b. [Mega-bench: Scaling multimodal evaluation to over 500 real-world tasks](#). *arXiv preprint arXiv:2410.10563*.
- Zixin Chen, Hongzhan Lin, Kaixin Li, Ziyang Luo, Zhen Ye, Guang Chen, Zhiyong Huang, and Jing Ma. 2025. [AdamMeme: Adaptively probe the reasoning capacity of multimodal large language models on harmfulness](#). In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4234–4253, Vienna, Austria. Association for Computational Linguistics.
- Zixin Chen, Hongzhan Lin, Ziyang Luo, Mingfei Cheng, Jing Ma, and Guang Chen. 2024c. [Cofipara: A coarse-to-fine paradigm for multimodal sarcasm target identification with large multimodal models](#). In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9663–9687.
- Zesen Cheng, Sicong Leng, Hang Zhang, Yifei Xin, Xin Li, Guanzheng Chen, Yongxin Zhu, Wenqi Zhang, Ziyang Luo, Deli Zhao, et al. 2024. [Videollama 2: Advancing spatial-temporal modeling and audio understanding in video-lmms](#). *arXiv preprint arXiv:2406.07476*.
- Rao Fu, Ziyang Luo, Hongzhan Lin, Zhen Ye, and Jing Ma. 2024. [Scratcheval: Are gpt-4o smarter than my child? evaluating large multimodal models with visual programming challenges](#). *arXiv preprint arXiv:2411.18932*.
- Ziyang Gong, Wenhao Li, Oliver Ma, Songyuan Li, Jiayi Ji, Xue Yang, Gen Luo, Junchi Yan, and Rongrong Ji. 2025. [Space-10: A comprehensive benchmark for multimodal large language models in compositional spatial intelligence](#). *arXiv preprint arXiv:2506.07966*.
- Jianzhao Huang, Hongzhan Lin, Ziyang Liu, Ziyang Luo, Guang Chen, and Jing Ma. 2024. [Towards low-resource harmful meme detection with lmm agents](#). *arXiv preprint arXiv:2411.05383*.
- Satyapriya Krishna. 2023. [On the intersection of self-correction and trust in language models](#). *CoRR*, abs/2311.02801.
- Tian Lan, Wenwei Zhang, Chen Xu, Heyan Huang, Dahua Lin, Kai Chen, and Xian-Ling Mao. 2024. [Criticeval: Evaluating large-scale language model as critic](#). *Advances in Neural Information Processing Systems*, 37:66907–66960.
- Dawei Li, Bohan Jiang, Liangjie Huang, Alimohammad Beigi, Chengshuai Zhao, Zhen Tan, Amrita Bhattacharjee, Yuxuan Jiang, Canyu Chen, Tianhao Wu, et al. 2024a. [From generation to judgment: Opportunities and challenges of llm-as-a-judge](#). *arXiv preprint arXiv:2411.16594*.
- Kaixin Li, Ziyang Meng, Hongzhan Lin, Ziyang Luo, Yuchen Tian, Jing Ma, Zhiyong Huang, and Tat-Seng Chua. 2025. [Screenspot-pro: Gui grounding for professional high-resolution computer use](#). *arXiv preprint arXiv:2504.07981*.
- Kaixin Li, Yuchen Tian, Qisheng Hu, Ziyang Luo, Zhiyong Huang, and Jing Ma. 2024b. [Mmcode: Benchmarking multimodal large language models for code generation with visually rich programming problems](#). *arXiv preprint arXiv:2404.09486*.
- Lei Li, Yuancheng Wei, Zhihui Xie, Xuqing Yang, Yifan Song, Peiyi Wang, Chenxin An, Tianyu Liu, Sujian Li, Bill Yuchen Lin, Lingpeng Kong, and Qi Liu. 2024c. [Vlrewardbench: A challenging benchmark for vision-language generative reward models](#). *CoRR*, abs/2411.17451.
- Shenzhi Li, Rongyu Lin, and Shichao Pei. 2024d. [Multi-modal preference alignment remedies regression of visual instruction tuning on language model](#). *CoRR*, abs/2402.10884.
- Hongzhan Lin, Yang Deng, Yuxuan Gu, Wenxuan Zhang, Jing Ma, See-Kiong Ng, and Tat-Seng Chua. 2025. [FACT-AUDIT: An adaptive multi-agent framework for dynamic fact-checking evaluation of large language models](#). In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 360–381, Vienna, Austria. Association for Computational Linguistics.
- Hongzhan Lin, Ziyang Luo, Wei Gao, Jing Ma, Bo Wang, and Ruichao Yang. 2024a. [Towards explainable harmful meme detection through multimodal debate between large language models](#). In *Proceedings of the ACM Web Conference 2024*, pages 2359–2370.

- Hongzhan Lin, Ziyang Luo, Jing Ma, and Long Chen. 2023. Beneath the surface: Unveiling harmful memes with multimodal reasoning distilled from large language models. *arXiv preprint arXiv:2312.05434*.
- Hongzhan Lin, Ziyang Luo, Bo Wang, Ruichao Yang, and Jing Ma. 2024b. Goat-bench: Safety insights to large multimodal models through meme-based social abuse. *ACM Transactions on Intelligent Systems and Technology*.
- Zicheng Lin, Zhibin Gou, Tian Liang, Ruilin Luo, Haowei Liu, and Yujiu Yang. 2024c. Criticbench: Benchmarking llms for critique-correct reasoning. In *Findings of the Association for Computational Linguistics, ACL 2024, Bangkok, Thailand and virtual meeting, August 11-16, 2024*, pages 1552–1587. Association for Computational Linguistics.
- Wei Liu, Junlong Li, Xiwen Zhang, Fan Zhou, Yu Cheng, and Junxian He. 2024. Diving into self-evolving training for multimodal reasoning.
- Ziyang Luo, Zhiqi Shen, Wenzhuo Yang, Zirui Zhao, Prathyusha Jwalapuram, Amrita Saha, Doyen Sahoo, Silvio Savarese, Caiming Xiong, and Junnan Li. 2025. Mcp-universe: Benchmarking large language models with real-world model context protocol servers. *arXiv preprint arXiv:2508.14704*.
- Ziyang Luo, Haoning Wu, Dongxu Li, Jing Ma, Mohan S. Kankanhalli, and Junnan Li. 2024. Videoautoarena: An automated arena for evaluating large multimodal models in video analysis through user simulation. *CoRR*, abs/2411.13281.
- Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Shashank Gupta, Bodhisattwa Prasad Majumder, Katherine Hermann, Sean Welleck, Amir Yazdanbakhsh, and Peter Clark. 2023. Self-refine: Iterative refinement with self-feedback. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.
- Liangming Pan, Michael Saxon, Wenda Xu, Deepak Nathani, Xinyi Wang, and William Yang Wang. 2023. Automatically correcting large language models: Surveying the landscape of diverse self-correction strategies. *CoRR*, abs/2308.03188.
- Michael J. Parker, Caitlin Anderson, Claire Stone, and YeaRim Oh. 2024. A large language model approach to educational survey feedback analysis. *International Journal of Artificial Intelligence in Education*.
- Shichao Sun, Junlong Li, Weizhe Yuan, Ruiyuan Yuan, Wenjie Li, and Pengfei Liu. 2024. The critique of critique. In *Findings of the Association for Computational Linguistics, ACL 2024, Bangkok, Thailand and virtual meeting, August 11-16, 2024*, pages 9077–9096. Association for Computational Linguistics.
- Sijun Tan, Siyuan Zhuang, Kyle Montgomery, William Yuan Tang, Alejandro Cuadron, Chenguang Wang, Raluca A. Popa, and Ion Stoica. 2025. Judgebench: A benchmark for evaluating llm-based judges. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net.
- Zhen Tan, Dawei Li, Song Wang, Alimohammad Beigi, Bohan Jiang, Amrita Bhattacharjee, Mansooreh Karami, Jundong Li, Lu Cheng, and Huan Liu. 2024. Large language models for data annotation and synthesis: A survey. *arXiv preprint arXiv:2402.13446*.
- Shengkang Wang, Hongzhan Lin, Ziyang Luo, Zhen Ye, Guang Chen, and Jing Ma. 2024. Mfc-bench: Benchmarking multimodal fact-checking with large vision-language models. *arXiv preprint arXiv:2406.11288*.
- 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. 2023a. Shepherd: A critic for language model generation. *arXiv preprint arXiv:2308.04592*.
- 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. *CoRR*, abs/2308.04592.
- Yaoting Wang, Shengqiong Wu, Yuecheng Zhang, Shuicheng Yan, Ziwei Liu, Jiebo Luo, and Hao Fei. 2025. Multimodal chain-of-thought reasoning: A comprehensive survey. *CoRR*, abs/2503.12605.
- Tianyi Xiong, Xiayao Wang, Dong Guo, Qinghao Ye, Haoqi Fan, Quanquan Gu, Heng Huang, and Chunyu Li. 2024. Llava-critic: Learning to evaluate multimodal models. *CoRR*, abs/2410.02712.
- Yan Yang, Dongxu Li, Yutong Dai, Yuhao Yang, Ziyang Luo, Zirui Zhao, Zhiyuan Hu, Junzhe Huang, Amrita Saha, Zeyuan Chen, et al. 2025. Gta1: Gui test-time scaling agent. *arXiv preprint arXiv:2507.05791*.
- Yuhao Yang, Yue Wang, Dongxu Li, Ziyang Luo, Bei Chen, Chao Huang, and Junnan Li. 2024. Aria-ui: Visual grounding for gui instructions. *arXiv preprint arXiv:2412.16256*.
- Michihiro Yasunaga, Luke Zettlemoyer, and Marjan Ghazvininejad. 2025. Multimodal rewardbench: Holistic evaluation of reward models for vision language models. *CoRR*, abs/2502.14191.
- Alexander Zhang, Marcus Dong, Jiaheng Liu, Wei Zhang, Yejie Wang, Jian Yang, Ge Zhang, Tianyu Liu, Zhongyuan Peng, Yingshui Tan, et al. 2025. Codecriticbench: A holistic code critique benchmark for large language models. *arXiv preprint arXiv:2502.16614*.

Di Zhang, Junxian Li, Jingdi Lei, Xunzhi Wang, Yujie Liu, Zonglin Yang, Jiatong Li, Weida Wang, Suorong Yang, Jianbo Wu, Peng Ye, Wanli Ouyang, and Dongzhan Zhou. 2024. [Critic-v: VLM critics help catch VLM errors in multimodal reasoning](#). *CoRR*, abs/2411.18203.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. [Judging llm-as-a-judge with mt-bench and chatbot arena](#). In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

A Dataset and LMMs Information.

Task Type	Sub-datasets					Total
	Core	Core Single-image	Open	Open Single-image		
Perception	266	146	24	18		454
Planning	146	80	10	8		244
Knowledge	142	108	52	46		348
Information Extraction	112	78	32	4		226
Mathematics	66	60	-	-		126
Coding	58	28	4	4		94
Science	54	40	4	4		102
Metric	36	6	4	-		46
Total	880	546	130	84		1640

Table 5: The statistics of tasks that are selected to generate responses for our benchmark.

Task Type	Sub-datasets					Total
	Core	Core Single-image	Open	Open Single-image		
Perception	668	435	61	54		1218
Planning	320	238	26	23		607
Knowledge	394	319	150	137		1000
Information Extraction	290	231	51	12		584
Mathematics	189	179	-	-		368
Coding	170	82	12	12		276
Science	161	118	12	12		303
Metric	90	18	7	-		115
Total	2282	1620	319	250		4471

Table 6: Dataset statistics of MM-CRITIC.

Task Type	Correlation	Comparison		
		Group 1	Group 2	Group 3
Perception	35	30	30	30
Planning	35	30	1	23
Knowledge	35	30	16	30
Information Extraction	35	30	13	30
Mathematics	35	22	-	24
Coding	35	30	7	15
Science	35	25	5	26
Metric	35	11	4	23
Total	280	208	76	201

Table 7: Dataset statistics of sub-datasets for correlation and comparative critique in MM-CRITIC.

LMMs	Source
InternVL2.5-4B	https://huggingface.co/OpenGVLab/InternVL2_5-4B
InternVL2.5-8B	https://huggingface.co/OpenGVLab/InternVL2_5-8B
InternVL2.5-26B	https://huggingface.co/OpenGVLab/InternVL2_5-26B
Phi-3.5-vision-instruct	https://huggingface.co/microsoft/Phi-3.5-vision-instruct
Phi-4-multimodal-instruct	https://huggingface.co/microsoft/Phi-4-multimodal-instruct
Qwen2.5-vl-3b-Instruct	https://huggingface.co/Qwen/Qwen2.5-vl-3b-Instruct
Qwen2.5-vl-7b-Instruct	https://huggingface.co/Qwen/Qwen2.5-vl-7b-Instruct
Deepseek-vl2-tiny	https://huggingface.co/deepseek-ai/deepseek-vl2-tiny
Llava-1.5-7b-hf	https://huggingface.co/llava-hf/llava-1.5-7b-hf
Llava-onevision-qwen2-7b-ov-hf	https://huggingface.co/llava-hf/llava-onevision-qwen2-7b-ov-hf
Llama-3.2-11b-vision-Instruct	https://huggingface.co/unslloth/Llama-3.2-11b-vision-Instruct
Pixtral-12b	https://huggingface.co/mistral-community/pixtral-12b

Table 8: The list of used LMMs for generating responses.

LMMs	Source
openai/o4-mini	https://openrouter.ai/openai/o4-mini
openai/gpt-4o-2024-11-20	https://openrouter.ai/openai/gpt-4o-2024-11-20www
openai/gpt-4o-mini	https://openrouter.ai/openai/gpt-4o-mini
anthropic/clause-3.7-sonnet	https://openrouter.ai/anthropic/clause-3.7-sonnet
google/gemini-2.5-flash-preview	https://openrouter.ai/google/gemini-2.5-flash-preview
google/gemini-2.5-pro-preview	https://openrouter.ai/google/gemini-2.5-pro-preview
x-ai/grok-2-vision-1212	https://openrouter.ai/x-ai/grok-2-vision-1212
qwen/qwen2.5-vl-32b-instruct	https://openrouter.ai/qwen/qwen2.5-vl-32b-instruct
qwen/qwen2.5-vl-72b-instruct	https://openrouter.ai/qwen/qwen2.5-vl-72b-instruct
mistralai/pixtral-large-2411	https://openrouter.ai/mistralai/pixtral-large-2411
meta-llama/llama-4-maverick	https://openrouter.ai/meta-llama/llama-4-maverick
qwen/qwen-2.5-vl-7b-instruct	https://openrouter.ai/qwen/qwen-2.5-vl-7b-instruct
meta-llama/llama-3.2-11b-vision-instruct	https://openrouter.ai/meta-llama/llama-3.2-11b-vision-instruct
mistralai/pixtral-12b	https://openrouter.ai/mistralai/pixtral-12b
google/gemma-3-12b-it	https://openrouter.ai/google/gemma-3-12b-it
google/gemma-3-27b-it	https://openrouter.ai/google/gemma-3-27b-it
meta-llama/llama-4-scout	https://openrouter.ai/meta-llama/llama-4-scout
google/gemma-3-4b-it	https://openrouter.ai/google/gemma-3-4b-it

Table 9: The list of used LMM APIs through OpenRouter.

B Prompts and Scoring Rubric Checklist

Task	Score Rubric
Common for All Tasks	<p>Correctness: For tasks with ground truth, carefully check the response whether gives correct answer; For tasks with open answers, carefully analyze the accuracy of generated responses, including but not limited to the following aspects: * consistent with reference answer * factual knowledge</p> <p>Response Quality: Carefully analyze the quality of generated responses, including but not limited to the following aspects: * correct spelling/grammar * readability and comprehensibility * effectiveness or usefulness.</p>
Knowledge	<p>Factuality: To check the generated response whether it is in line with facts. If a response is based on a false premise, it can be regarded as a bad sample.</p> <p>Relevance: Consider whether the generated content is relevant to the question. If the content is unrelated to the question, it can be reviewed as low quality.</p>
Perception	<p>Detail: This criterion aims to check whether the generated content contains sufficient and correct detail. A response is considered lower quality if it is overly brief and lacks details.</p>
Information Extraction	<p>Effectiveness: This criterion aims to check whether the answers effectively extract information, based on the question. If the generated answers do not provide effective information, they can be regarded as lower quality.</p>
Planning	<p>Feasibility: It is vital to assess whether the generated planning is feasible in the real world. If the planning is unfeasible, it can be viewed as lower quality.</p>
Science	<p>Factuality: To check whether the generated response is in line with scientific facts. If a response is based on a false premise, it can be regarded as a bad sample.</p>
Metric	<p>Effective Utilization: To check whether the generated response is in line with scientific facts. If a response is based on a false premise, it can be regarded as a bad sample.</p>
Mathematics	<p>Correctness: Correctness-based for mathematics is a multi-step checking criterion, firstly assess the generated response whether it provides a correct reasoning process (if it includes), Secondly check the generated response whether it provides the correct answer. If the generated response provides inappropriate reasoning and a wrong answer, it can be regarded as of very poor quality. If the generated response provides correct reasoning and a wrong answer, it can be regarded as of relatively high quality, compared with the above case.</p>
Coding	<p>Program Grammar: This criterion aims to check whether the generated codes align with specific program language features. If the generated codes utilize non-existent program language features, they can be regarded as of low quality.</p> <p>Correctness: Correctness-based for coding is a multi-step checking criterion, firstly assess the generated response whether it provides a correct coding framework (if it includes), Secondly check the generated response whether it provides correct output of codes. If the generated response provides an unreasonable coding framework and a wrong output, it can be regarded as of very poor quality. If the generated response provides both a correct coding framework and output, it can be regarded as of relatively high quality, compared with the above case.</p>

Table 10: The score rubrics for different task types. Human experts use these score rubrics to check and annotate.

Basic Reference Critique Generation Prompt for GPT-4o

You are a professional critical AI specialist who can evaluate the response generated by a vision large language model with corresponding domain knowledge of questions. You need to refer to the following rubrics:

#Common for All Tasks:

##Correctness:

For tasks with ground truth, carefully check the response whether gives correct answer;
For tasks with open answers, carefully analyze the accuracy of generated responses,
including but not limited to the following aspects:

- * consistent with reference answer
 - * factual knowledge.
- ##Response Quality:** Carefully analyze the quality of generated responses,
including but not limited to the following aspects:
- * correct spelling/grammar
 - * readability and comprehensibility
 - * effectiveness or usefulness.

Besides, the question is about application_name, you also need to carefully refer to the emphasized rubrics:
(Corresponding rubric for different task types, here is an example for Knowledge)

##Factuality:

To check whether the generated response is in line with facts.

If a response is based on a false premise, it can be regarded as a bad sample.

##Relevance:

Consider whether the generated content is relevant to the question.

If the content is unrelated to the question, it can be reviewed as low quality.

#The following is the question and the response generated by a vision large language model:

— Start of Question —

\$Question (include images)\$

— End of Question —

— Start of Response —

\$Response\$

— End of Response —

#Here is the ground truth answer (or reference answer),
which can effectively help you give reliable evaluations about the response:

— Start of Answer —

\$Answer\$

— End of Answer —

#Evaluation Steps:

##First, you need to score the response quality, and the score ranges from 0 to 10 as an integer,

- [0,3] corresponds to a low-quality response,
- [4,7] corresponds to a medium-quality response,
- [8,9] corresponds to a high-quality response,
- 10 corresponds to a correct response.

Second, you need to give a textual critique including but not limited to the following requirements:

- Provide detailed, point-by-point feedback on the answer.
- Each critique should be specific and self-contained.
- Clearly identify any issues, avoiding vague or ambiguous descriptions.
- Offer constructive suggestions for improvement.

#Output Format:

Provide the evaluation in JSON format as follows:

```
```json
{
 "correct": "Based on the ground truth answer (if have), indicate whether the assistant's response is ['Correct', 'Error']",
 "response_quality": "A specific integer score ranging from 0 to 10",
 "reference_critique": "Based on the evaluation, give a comprehensive textual critique",
 "reference_correct": "Based on the evaluation, give a modification if the response is not of good quality enough."
}
```

```

Figure 5: Basic Reference Critique Generation Prompt for GPT-4o.

Critique evaluation judge prompt with reference critique.

You are a professional critique evaluation judge who can evaluate the critique generated by a vision large language model based on the corresponding question and response.

#: The following are the question and generated response, and critique/correct that need to be evaluated, respectively.

— Start of Question and Response —

\$Question (include images)\$

\$Response\$

— End of Question and Response —

— Start of Critique —

\$Critique/Correct\$

— End of Critique —

#Here is the reference critique/correct:

— Start of Reference Critique —

\$Reference Critique/Correct\$

— End of Reference Critique —

#Very important rules !!!:

##1. If you think the generated critique is better than the reference critique, you could provide a score higher than 8

##2. Please first generate your explanation for the generated critique by comparing it with the reference critique, and then you must generate the score, where the score ranges from 0 to 10 as an integer.

#Output Format:

Provide the evaluation in JSON format as follows:

```json

```
{
 "explanation": "Based on the reference critique, give explanation about the score given by yourself",
 "score": "A specific integer score ranging from 0 to 10 ",
}
```

Figure 6: Critique evaluation judge prompt with reference critique.

### Basic critique prompt.

You are a professional expert in critique who can evaluate the responses generated by large multimodal models and analyze response quality.

#: The following are the question and the generated response from large multimodal models.

— Start of Question —

\$Question (include images)\$

— End of Question —

— Start of Response —

\$Response \$

— End of Response —

#### #Very important rules !!!:

##1. Carefully understand the question, and then analyze the response.

##2. Please first generate your judgment of correctness about the response, and then give a comprehensive textual critique to explain.

#### #Output Format:

Provide the evaluation in JSON format as follows:

```json

```
{  
  "correct": "Based on the ground truth answer (if have), indicate whether the assistant's response is ['Correct', 'Error']",  
  "critique": "Give a comprehensive textual critique, due to the question and response",  
}
```

Figure 7: Basic critique prompt.

Correction critique prompt.

You are a professional expert in critique who can evaluate the responses generated by large multimodal models and analyze response quality.

#: The following are the question and the generated response from large multimodal models.

— Start of Question —

\$Question (include images)\$

— End of Question —

— Start of Response —

\$Response \$

— End of Response —

#Very important rules !!!:

##1. Carefully understand the question, and then analyze the response.

##2. The original response is not good enough, and you should give your own response to better answer the question.

#Output Format:

Provide the evaluation in JSON format as follows:

```
```json
{
 "modified answer": "Since the response is unsatisfactory, give your own response here, due to the question",
}
```
```

Figure 8: Correction critique prompt.

Comparative critique prompt.

You are a professional comparative critique evaluation judge who can evaluate the responses generated by two different large multimodal models and choose the better one.

#: The following are the question and two generated response from two different models.

— Start of Question —

\$Question (include images)\$

— End of Question —

— Start of Response A —

\$Response A\$

— End of Response A —

— Start of Response B —

\$Response B\$

— End of Response B —

#Very important rules !!!:

##1. Carefully compare the two responses, and then choose the better one.

##2. Please first generate your explanation for the choice by comparing the two responses, and then you must clearly state your choice following the format: "choice": X, where X is A or B, corresponding to response A and response B.

#Output Format:

Provide the evaluation in JSON format as follows:

```
```json
{
 "choice": "chose the better response quality model and indicate your choice is ["A", "B"]",
 "explanation": "Based on the two responses, give explanation about the choice given by yourself",
}
```
```

Figure 9: Comparative critique prompt.

C Experimental Results

| Model | Task types | | | | | | | | | Avg. |
|---|------------|----------|-----------|------------------------|-------------|--------|---------|--------|-------|------|
| | Perception | Planning | Knowledge | Information Extraction | Mathematics | Coding | Science | Metric | | |
| <i>Proprietary Models</i> | | | | | | | | | | |
| o4-mini | 7.913 | 7.987 | 7.817 | 7.806 | 8.080 | 80.84 | 7.869 | 8.049 | 7.924 | |
| GPT-4o | 7.537 | 7.473 | 7.288 | 7.496 | 7.444 | 7.523 | 7.686 | 7.937 | 7.499 | |
| GPT-4o-mini | 6.65 | 6.653 | 6.691 | 6.831 | 6.346 | 6.437 | 6.516 | 6.742 | 6.634 | |
| Claude-3.7-sonnet | 8.084 | 8.066 | 8.137 | 8.176 | 8.283 | 8.058 | 8.230 | 7.178 | 8.113 | |
| Gemini-2.5-flash | 6.478 | 6.358 | 6.242 | 6.386 | 6.596 | 6.786 | 7.121 | 6.784 | 6.495 | |
| Gemini-2.5-pro | 8.524 | 8.831 | 8.380 | 8.325 | 8.987 | 8.524 | 8.537 | 8.786 | 8.558 | |
| Grok-2-vision | 7.567 | 7.654 | 7.509 | 7.112 | 7.474 | 7.645 | 7.760 | 7.695 | 7.523 | |
| <i>Open-weight Models (Larger than 30B)</i> | | | | | | | | | | |
| Qwen2.5-vl-32b | 8.245 | 8.286 | 7.983 | 8.237 | 7.861 | 8.413 | 8.377 | 8.566 | 8.208 | |
| Qwen2.5-vl-72b | 6.933 | 7.332 | 6.892 | 6.194 | 6.725 | 7.430 | 7.155 | 6.914 | 6.931 | |
| Pixtral-large | 7.445 | 7.889 | 7.289 | 7.495 | 7.272 | 7.459 | 7.842 | 7.263 | 7.489 | |
| Llama-4-maverick | 5.785 | 5.603 | 5.648 | 6.406 | 5.484 | 5.826 | 6.054 | 6.217 | 5.811 | |
| <i>Open-weight Models (Less than 30B)</i> | | | | | | | | | | |
| Gemma-3-4b | 5.245 | 5.211 | 5.239 | 4.663 | 4.962 | 4.680 | 5.049 | 6.078 | 5.130 | |
| Qwen2.5-vl-7b | 4.509 | 4.889 | 4.300 | 4.854 | 4.847 | 4.658 | 4.642 | 4.575 | 4.617 | |
| Llama-3.2-11b-vision | 5.230 | 5.266 | 5.233 | 4.876 | 5.162 | 4.927 | 5.198 | 5.791 | 5.185 | |
| Pixtral-12b | 5.317 | 5.482 | 5.218 | 4.623 | 4.899 | 5.034 | 5.599 | 5.139 | 5.201 | |
| Gemma-3-12b | 6.572 | 6.571 | 6.761 | 5.678 | 6.463 | 6.838 | 6.992 | 6.765 | 6.566 | |
| Gemma-3-27b | 7.285 | 7.235 | 7.032 | 6.135 | 7.169 | 7.262 | 7.290 | 7.031 | 7.107 | |
| Llama-4-scout | 5.996 | 5.714 | 5.723 | 5.563 | 5.543 | 5.537 | 5.985 | 5.633 | 5.771 | |

Table 11: Critique scores of **Core** subset on different task types.

| Model | Task types | | | | | | | | | Avg. |
|---|------------|----------|-----------|------------------------|-------------|--------|---------|--------|-------|------|
| | Perception | Planning | Knowledge | Information Extraction | Mathematics | Coding | Science | Metric | | |
| <i>Proprietary Models</i> | | | | | | | | | | |
| o4-mini | 7.950 | 8.004 | 7.909 | 7.934 | 8.05 | 7.790 | 8.103 | 7.111 | 7.953 | |
| GPT-4o | 7.285 | 7.455 | 7.596 | 7.548 | 7.370 | 7.418 | 7.180 | 7.418 | 7.429 | |
| GPT-4o-mini | 6.508 | 6.680 | 6.592 | 6.507 | 6.469 | 6.457 | 6.398 | 6.167 | 6.534 | |
| Claude-3.7-sonnet | 8.102 | 8.158 | 8.070 | 7.961 | 8.154 | 8.104 | 8.133 | 6.944 | 8.080 | |
| Gemini-2.5-flash | 6.384 | 6.459 | 6.225 | 6.418 | 6.768 | 6.623 | 7.010 | 6.722 | 6.460 | |
| Gemini-2.5-pro | 8.542 | 8.692 | 8.377 | 8.462 | 8.933 | 8.427 | 8.500 | 8.750 | 8.549 | |
| Grok-2-vision | 7.476 | 7.861 | 7.606 | 6.811 | 7.798 | 7.250 | 7.765 | 6.944 | 7.490 | |
| <i>Open-weight Models (Larger than 30B)</i> | | | | | | | | | | |
| Qwen2.5-vl-32b | 7.946 | 8.496 | 8.318 | 8.007 | 7.844 | 8.074 | 8.329 | 9.000 | 8.138 | |
| Qwen2.5-vl-72b | 7.180 | 7.149 | 7.270 | 7.090 | 6.953 | 7.200 | 7.330 | 6.176 | 7.157 | |
| Pixtral-large | 7.480 | 7.788 | 7.497 | 7.254 | 7.588 | 7.583 | 7.800 | 7.000 | 7.531 | |
| Llama-4-maverick | 6.151 | 5.672 | 5.897 | 6.232 | 5.804 | 5.808 | 6.102 | 5.500 | 5.971 | |
| <i>Open-weight Models (Less than 30B)</i> | | | | | | | | | | |
| Gemma-3-4b | 5.670 | 5.697 | 5.511 | 4.706 | 5.658 | 5.211 | 5.769 | 5.533 | 5.509 | |
| Qwen2.5-vl-7b | 5.113 | 5.391 | 5.280 | 4.453 | 4.925 | 5.078 | 5.122 | 5.846 | 5.093 | |
| Llama-3.2-11b-vision | 5.113 | 5.391 | 5.280 | 4.453 | 4.925 | 5.078 | 5.122 | 5.846 | 5.093 | |
| Pixtral-12b | 0.657 | 0.833 | 0.668 | 0.621 | 0.752 | 0.620 | 0.689 | 0.444 | 0.687 | |
| Gemma-3-12b | 6.429 | 6.526 | 6.573 | 6.000 | 6.361 | 6.268 | 6.863 | 5.389 | 6.419 | |
| Gemma-3-27b | 6.987 | 7.009 | 7.173 | 5.987 | 7.270 | 6.194 | 7.333 | 5.750 | 6.921 | |
| Llama-4-scout | 6.087 | 5.713 | 5.891 | 5.652 | 5.446 | 5.225 | 6.172 | 5.667 | 5.822 | |

Table 12: Critique scores of **Core-single-image** subset on different task types.

| Model | Task types | | | | | | | | |
|---|------------|----------|-----------|------------------------|-------------|--------|---------|--------|-------|
| | Perception | Planning | Knowledge | Information Extraction | Mathematics | Coding | Science | Metric | Avg. |
| <i>Proprietary Models</i> | | | | | | | | | |
| o4-mini | 7.817 | 8.077 | 7.953 | 7.510 | - | 7.727 | 8.500 | 7.857 | 7.877 |
| GPT-4o | 7.808 | 7.933 | 7.818 | 7.980 | - | 7.700 | 7.818 | 5.600 | 7.807 |
| GPT-4o-mini | 6.869 | 6.615 | 6.747 | 5.745 | - | 6.833 | 5.833 | 5.857 | 6.549 |
| Claude-3.7-sonnet | 8.233 | 8.591 | 8.169 | 7.627 | - | 8.636 | 7.818 | 7.00 | 8.102 |
| Gemini-2.5-flash | 6.049 | 6.615 | 6.718 | 6.549 | - | 6.083 | 6.500 | 5.714 | 6.500 |
| Gemini-2.5-pro | 8.327 | 8.875 | 8.340 | 7.878 | - | 8.273 | 7.182 | 6.000 | 8.246 |
| Grok-2-vision | 8.117 | 8.417 | 8.242 | 7.314 | - | 8.500 | 8.333 | 7.500 | 8.066 |
| <i>Open-weight Models (Larger than 30B)</i> | | | | | | | | | |
| Qwen2.5-vl-32b | 8.882 | 9.000 | 8.419 | 8.033 | - | 9.300 | 9.364 | 7.800 | 8.566 |
| Qwen2.5-vl-72b | 7.000 | 7.042 | 7.034 | 6.800 | - | 7.000 | 7.500 | 5.429 | 6.974 |
| Pixtral-large | 7.855 | 7.895 | 7.832 | 7.575 | - | 7.818 | 7.833 | 5.000 | 7.743 |
| Llama-4-maverick | 6.441 | 6.077 | 6.541 | 6.040 | - | 6.000 | 6.167 | 5.000 | 6.342 |
| <i>Open-weight Models (Less than 30B)</i> | | | | | | | | | |
| Gemma-3-4b | 5.696 | 7.214 | 6.638 | 5.737 | - | 2.583 | 5.000 | 3.000 | 6.155 |
| Qwen2.5-vl-7b | 0.796 | 0.727 | 0.682 | 0.722 | - | 0.750 | 0.600 | 0.0 | 0.711 |
| Llama-3.2-11b-vision | 5.230 | 5.640 | 5.333 | 4.941 | - | 3.250 | 4.750 | 5.714 | 5.179 |
| Pixtral-12b | 5.818 | 5.417 | 5.987 | 5.061 | - | 5.250 | 4.000 | 3.667 | 5.700 |
| Gemma-3-12b | 6.700 | - | 7.025 | 5.176 | - | 5.889 | 7.600 | 4.000 | 6.744 |
| Gemma-3-27b | 6.467 | 8.474 | 7.500 | 5.889 | - | 6.875 | 7.417 | 4.000 | 7.297 |
| Llama-4-scout | 6.500 | 6.318 | 6.386 | 5.810 | - | 6.083 | 5.917 | 9.000 | 6.336 |

Table 13: Critique scores of **Open** subset on different task types.

| Model | Task types | | | | | | | | |
|---|------------|----------|-----------|------------------------|-------------|--------|---------|--------|-------|
| | Perception | Planning | Knowledge | Information Extraction | Mathematics | Coding | Science | Metric | Avg. |
| <i>Proprietary Models</i> | | | | | | | | | |
| o4-mini | 8.037 | 8.087 | 7.897 | 8.083 | - | 8.083 | 8.167 | - | 7.976 |
| GPT-4o | 7.316 | 7.875 | 7.575 | 8.500 | - | 7.500 | 8.556 | - | 7.637 |
| GPT-4o-mini | 0.593 | 0.739 | 0.679 | 0.750 | - | 0.917 | 0.900 | - | 0.690 |
| Claude-3.7-sonnet | 8.038 | 8.684 | 7.916 | 8.667 | - | 8.636 | 8.333 | - | 8.097 |
| Gemini-2.5-flash | 6.264 | 6.238 | 6.276 | 6.833 | - | 6.583 | 6.833 | - | 6.340 |
| Gemini-2.5-pro | 8.431 | 8.895 | 8.283 | 8.750 | - | 8.091 | 7.667 | - | 8.325 |
| Grok-2-vision | 8.260 | 8.261 | 8.235 | 8.750 | - | 8.500 | 8.143 | - | 8.274 |
| <i>Open-weight Models (Larger than 30B)</i> | | | | | | | | | |
| Qwen2.5-vl-32b | 8.857 | 8.667 | 8.392 | 9.000 | - | 8.667 | - | - | 8.495 |
| Qwen2.5-vl-72b | 6.843 | 7.909 | 6.788 | 7.667 | - | 7.333 | 7.167 | - | 6.992 |
| Pixtral-large | 7.794 | 8.300 | 7.739 | 8.714 | - | 7.636 | 7.333 | - | 7.784 |
| Llama-4-maverick | 6.333 | 5.227 | 6.447 | 6.500 | - | 6.083 | 5.455 | - | 6.250 |
| <i>Open-weight Models (Less than 30B)</i> | | | | | | | | | |
| Gemma-3-4b | 7.042 | 7.313 | 6.828 | 5.917 | - | 3.833 | 4.667 | - | 6.643 |
| Qwen2.5-vl-7b | 4.721 | 4.905 | 4.750 | 5.222 | - | 5.273 | 4.727 | - | 4.806 |
| Llama-3.2-11b-vision | 5.143 | 5.222 | 5.450 | 6.091 | - | 4.778 | 5.091 | - | 5.351 |
| Pixtral-12b | 5.755 | 5.130 | 5.815 | 6.417 | - | 5.333 | 6.200 | - | 5.759 |
| Gemma-3-12b | 6.800 | 6.250 | 7.102 | 6.444 | - | 6.400 | 7.091 | - | 6.944 |
| Gemma-3-27b | 7.489 | 7.957 | 7.872 | 7.667 | - | 6.818 | 6.909 | - | 7.700 |
| Llama-4-scout | 6.620 | 6.095 | 6.535 | 7.250 | - | 6.583 | 5.800 | - | 6.521 |

Table 14: Critique scores of **Open-single-image** subset on different task types.

| Model | Task types | | | | | | | | |
|---|------------|----------|-----------|------------------------|-------------|--------|---------|--------|-------|
| | Perception | Planning | Knowledge | Information Extraction | Mathematics | Coding | Science | Metric | Avg. |
| <i>Proprietary Models</i> | | | | | | | | | |
| o4-mini | 0.872 | 0.974 | 0.868 | 0.940 | 0.899 | 0.916 | 0.869 | 0.783 | 0.896 |
| GPT-4o | 0.847 | 0.917 | 0.809 | 0.848 | 0.792 | 0.754 | 0.821 | 0.738 | 0.832 |
| GPT-4o-mini | 0.818 | 0.852 | 0.821 | 0.852 | 0.840 | 0.821 | 0.795 | 0.678 | 0.833 |
| Claude-3.7-sonnet | 0.808 | 0.925 | 0.821 | 0.884 | 0.870 | 0.844 | 0.814 | 0.698 | 0.840 |
| Gemini-2.5-flash | 0.840 | 0.903 | 0.772 | 0.841 | 0.831 | 0.818 | 0.814 | 0.667 | 0.826 |
| Gemini-2.5-pro | 0.826 | 0.941 | 0.838 | 0.943 | 0.898 | 0.889 | 0.789 | 0.775 | 0.865 |
| Grok-2-vision | 0.782 | 0.875 | 0.782 | 0.841 | 0.847 | 0.794 | 0.769 | 0.671 | 0.803 |
| <i>Open-weight Models (Larger than 30B)</i> | | | | | | | | | |
| Qwen2.5-vl-32b | 0.823 | 0.923 | 0.797 | 0.882 | 0.821 | 0.860 | 0.831 | 0.738 | 0.839 |
| Qwen2.5-vl-72b | 0.805 | 0.935 | 0.802 | 0.852 | 0.790 | 0.845 | 0.809 | 0.667 | 0.824 |
| Pixtral-large | 0.823 | 0.945 | 0.774 | 0.881 | 0.810 | 0.819 | 0.815 | 0.675 | 0.828 |
| Llama-4-maverick | 0.719 | 0.867 | 0.727 | 0.787 | 0.747 | 0.710 | 0.748 | 0.586 | 0.748 |
| <i>Open-weight Models (Less than 30B)</i> | | | | | | | | | |
| Gemma-3-4b | 0.490 | 0.621 | 0.501 | 0.462 | 0.535 | 0.426 | 0.490 | 0.469 | 0.508 |
| Qwen2.5-vl-7b | 0.736 | 0.893 | 0.747 | 0.805 | 0.800 | 0.804 | 0.796 | 0.716 | 0.783 |
| Llama-3.2-11b-vision | 0.693 | 0.832 | 0.703 | 0.724 | 0.789 | 0.793 | 0.610 | 0.618 | 0.721 |
| Pixtral-12b | 0.707 | 0.830 | 0.688 | 0.584 | 0.704 | 0.649 | 0.796 | 0.588 | 0.704 |
| Gemma-3-12b | 0.744 | 0.877 | 0.765 | 0.661 | 0.769 | 0.757 | 0.768 | 0.638 | 0.759 |
| Gemma-3-27b | 0.816 | 0.922 | 0.778 | 0.684 | 0.809 | 0.824 | 0.739 | 0.727 | 0.804 |
| Llama-4-scout | 0.748 | 0.890 | 0.745 | 0.693 | 0.739 | 0.724 | 0.752 | 0.656 | 0.757 |

Table 15: ACC_{critic} of **Core** subset on different task types.

| Model | Task types | | | | | | | | |
|---|------------|----------|-----------|------------------------|-------------|--------|---------|--------|-------|
| | Perception | Planning | Knowledge | Information Extraction | Mathematics | Coding | Science | Metric | Avg. |
| <i>Proprietary Models</i> | | | | | | | | | |
| o4-mini | 0.843 | 0.975 | 0.868 | 0.938 | 0.916 | 0.951 | 0.880 | 0.889 | 0.897 |
| GPT-4o | 0.811 | 0.916 | 0.805 | 0.855 | 0.824 | 0.875 | 0.787 | 0.944 | 0.834 |
| GPT-4o-mini | 0.789 | 0.958 | 0.815 | 0.804 | 0.859 | 0.866 | 0.822 | 0.833 | 0.836 |
| Claude-3.7-sonnet | 0.792 | 0.921 | 0.792 | 0.850 | 0.823 | 0.870 | 0.841 | 0.611 | 0.828 |
| Gemini-2.5-flash | 0.786 | 0.899 | 0.790 | 0.861 | 0.826 | 0.878 | 0.847 | 0.778 | 0.828 |
| Gemini-2.5-pro | 0.792 | 0.940 | 0.838 | 0.946 | 0.907 | 0.933 | 0.806 | 0.875 | 0.865 |
| Grok-2-vision | 0.762 | 0.915 | 0.778 | 0.794 | 0.858 | 0.787 | 0.827 | 0.667 | 0.806 |
| <i>Open-weight Models (Larger than 30B)</i> | | | | | | | | | |
| Qwen2.5-vl-32b | 0.773 | 0.948 | 0.807 | 0.787 | 0.791 | 0.796 | 0.785 | 1.0 | 0.811 |
| Qwen2.5-vl-72b | 0.793 | 0.950 | 0.790 | 0.848 | 0.827 | 0.902 | 0.847 | 0.889 | 0.838 |
| Pixtral-large | 0.789 | 0.925 | 0.821 | 0.846 | 0.836 | 0.849 | 0.858 | 0.778 | 0.836 |
| Llama-4-maverick | 0.784 | 0.911 | 0.735 | 0.813 | 0.844 | 0.885 | 0.839 | 0.667 | 0.812 |
| <i>Open-weight Models (Less than 30B)</i> | | | | | | | | | |
| Gemma-3-4b | 0.597 | 0.702 | 0.539 | 0.436 | 0.671 | 0.539 | 0.611 | 0.600 | 0.590 |
| Qwen2.5-vl-7b | 0.727 | 0.854 | 0.711 | 0.843 | 0.853 | 0.828 | 0.758 | 0.692 | 0.780 |
| Llama-3.2-11b-vision | 0.703 | 0.906 | 0.696 | 0.774 | 0.716 | 0.781 | 0.714 | 0.692 | 0.750 |
| Pixtral-12b | 0.657 | 0.833 | 0.668 | 0.621 | 0.752 | 0.620 | 0.689 | 0.444 | 0.687 |
| Gemma-3-12b | 0.701 | 0.842 | 0.727 | 0.676 | 0.796 | 0.676 | 0.823 | 0.444 | 0.739 |
| Gemma-3-27b | 0.737 | 0.876 | 0.765 | 0.665 | 0.906 | 0.582 | 0.824 | 0.500 | 0.773 |
| Llama-4-scout | 0.746 | 0.907 | 0.734 | 0.712 | 0.777 | 0.764 | 0.753 | 0.611 | 0.767 |

Table 16: ACC_{critic} of **Core-single-image** subset on different task types.

| Model | Task types | | | | | | | | | |
|---|------------|----------|-----------|------------------------|-------------|--------|---------|--------|-------|--|
| | Perception | Planning | Knowledge | Information Extraction | Mathematics | Coding | Science | Metric | Avg. | |
| <i>Proprietary Models</i> | | | | | | | | | | |
| o4-mini | 0.900 | 1.0 | 0.933 | 0.824 | - | 1.0 | 0.667 | 0.857 | 0.906 | |
| GPT-4o | 0.731 | 0.933 | 0.884 | 0.840 | - | 0.900 | 0.545 | 0.400 | 0.826 | |
| GPT-4o-mini | 0.721 | 0.923 | 0.807 | 0.608 | - | 0.917 | 0.750 | -0.429 | 0.762 | |
| Claude-3.7-sonnet | 0.733 | 0.955 | 0.873 | 0.608 | - | 1.0 | 0.727 | 0.500 | 0.799 | |
| Gemini-2.5-flash | 0.689 | 1.0 | 0.820 | 0.667 | - | 1.0 | 0.583 | 0.429 | 0.774 | |
| Gemini-2.5-pro | 0.891 | 1.0 | 0.896 | 0.707 | - | 1.0 | 0.545 | 0.667 | 0.865 | |
| Grok-2-vision | 0.750 | 0.958 | 0.859 | 0.745 | - | 0.833 | 0.666 | 0.714 | 0.818 | |
| <i>Open-weight Models (Larger than 30B)</i> | | | | | | | | | | |
| Qwen2.5-vl-32b | 0.853 | 0.923 | 0.849 | 0.833 | - | 1.0 | 0.818 | 0.600 | 0.852 | |
| Qwen2.5-vl-72b | 0.705 | 0.909 | 0.853 | 0.760 | - | 0.917 | 0.667 | 0.571 | 0.803 | |
| Pixtral-large | 0.691 | 0.947 | 0.869 | 0.725 | - | 1.0 | 0.667 | 0.333 | 0.804 | |
| Llama-4-maverick | 0.661 | 0.885 | 0.788 | 0.680 | - | 0.917 | 0.583 | 0.200 | 0.742 | |
| <i>Open-weight Models (Less than 30B)</i> | | | | | | | | | | |
| Gemma-3-4b | 0.391 | 0.933 | 0.583 | 0.579 | - | 0.083 | 0.400 | 0.0 | 0.546 | |
| Qwen2.5-vl-7b | 0.796 | 0.727 | 0.682 | 0.722 | - | 0.750 | 0.600 | 0.0 | 0.711 | |
| Llama-3.2-11b-vision | 0.836 | 0.846 | 0.667 | 0.765 | - | 0.583 | 0.750 | 0.571 | 0.728 | |
| Pixtral-12b | 0.709 | 0.917 | 0.733 | 0.606 | - | 0.833 | 0.500 | 0.0 | 0.721 | |
| Gemma-3-12b | 0.600 | - | 0.683 | 0.412 | - | 0.777 | 0.600 | 0.0 | 0.645 | |
| Gemma-3-27b | 0.533 | 0.947 | 0.775 | 0.444 | - | 0.750 | 0.583 | 0.0 | 0.720 | |
| Llama-4-scout | 0.707 | 0.955 | 0.828 | 0.619 | - | 0.917 | 0.750 | 1.0 | 0.797 | |

Table 17: $\text{ACC}_{\text{critic}}$ of **Open** subset on different task types.

| Model | Task types | | | | | | | | | |
|---|------------|----------|-----------|------------------------|-------------|--------|---------|--------|-------|--|
| | Perception | Planning | Knowledge | Information Extraction | Mathematics | Coding | Science | Metric | Avg. | |
| <i>Proprietary Models</i> | | | | | | | | | | |
| o4-mini | 0.778 | 1.0 | 0.891 | 0.583 | - | 0.916 | 0.750 | - | 0.856 | |
| GPT-4o | 0.684 | 0.875 | 0.796 | 0.833 | - | 0.900 | 0.889 | - | 0.789 | |
| GPT-4o-mini | 0.593 | 0.739 | 0.679 | 0.750 | - | 0.917 | 0.900 | - | 0.690 | |
| Claude-3.7-sonnet | 0.717 | 1.0 | 0.803 | 0.750 | - | 1.0 | 0.833 | - | 0.808 | |
| Gemini-2.5-flash | 0.722 | 0.870 | 0.745 | 0.583 | - | 1.0 | 0.750 | - | 0.756 | |
| Gemini-2.5-pro | 0.863 | 0.895 | 0.866 | 0.833 | - | 1.0 | 0.750 | - | 0.866 | |
| Grok-2-vision | 0.680 | 0.957 | 0.813 | 0.833 | - | 1.0 | 0.857 | - | 0.806 | |
| <i>Open-weight Models (Larger than 30B)</i> | | | | | | | | | | |
| Qwen2.5-vl-32b | 0.429 | 1.0 | 0.835 | 1.0 | - | 0.833 | - | - | 0.794 | |
| Qwen2.5-vl-72b | 0.759 | 0.957 | 0.818 | 0.667 | - | 0.917 | 0.667 | - | 0.808 | |
| Pixtral-large | 0.853 | 0.900 | 0.839 | 0.857 | - | 0.917 | 0.750 | - | 0.845 | |
| Llama-4-maverick | 0.608 | 0.818 | 0.722 | 0.667 | - | 0.917 | 0.545 | - | 0.705 | |
| <i>Open-weight Models (Less than 30B)</i> | | | | | | | | | | |
| Qwen2.5-vl-3b | 0.625 | 0.9375 | 0.613 | 0.416 | - | 0.166 | 0.5 | - | 0.611 | |
| Qwen2.5-vl-7b | 0.744 | 0.857 | 0.778 | 0.778 | - | 0.909 | 0.818 | - | 0.788 | |
| Llama-3.2-11b-vision | 0.833 | 0.778 | 0.741 | 0.636 | - | 0.889 | 0.636 | - | 0.759 | |
| Pixtral-12b | 0.528 | 0.870 | 0.706 | 0.583 | - | 0.833 | 0.818 | - | 0.688 | |
| Gemma-3-12b | 0.700 | 0.750 | 0.684 | 0.333 | - | 0.750 | 0.636 | - | 0.671 | |
| Gemma-3-27b | 0.638 | 1.0 | 0.754 | 0.667 | - | 0.727 | 0.636 | - | 0.744 | |
| Llama-4-scout | 0.720 | 0.905 | 0.847 | 0.833 | - | 0.917 | 0.600 | - | 0.818 | |

Table 18: $\text{ACC}_{\text{critic}}$ of **Open-single-image** subset on different task types.

D Ablation Study

| Model | Core | | Core
Single-image | | Open | | Open
Single-image | | Avg. | |
|---|-----------------------|-------|-----------------------|-------|-----------------------|-------|-----------------------|-------|-----------------------|--------------|
| | ACC _{critic} | Score |
| <i>Annotator: GPT-4o, Critique Judge: GPT-4.1</i> | | | | | | | | | | |
| o4-mini | 0.896 | 7.924 | 0.897 | 7.952 | 0.906 | 7.877 | 0.856 | 7.976 | 0.900 | 7.933 |
| GPT-4o | 0.832 | 7.499 | 0.834 | 7.429 | 0.826 | 7.807 | 0.789 | 7.637 | 0.830 | 7.503 |
| GPT-4o-mini | 0.833 | 6.634 | 0.836 | 6.534 | 0.762 | 6.549 | 0.690 | 6.416 | 0.821 | 6.580 |
| Claude-3.7-sonnet | 0.834 | 8.113 | 0.828 | 8.080 | 0.799 | 8.102 | 0.808 | 8.097 | 0.831 | 8.099 |
| Gemini-2.5-pro | 0.865 | 8.558 | 0.865 | 8.549 | 0.865 | 8.246 | 0.866 | 8.325 | 0.865 | 8.514 |
| <i>Annotator: Gemini-2.5-flash, Critique Judge: GPT-4.1</i> | | | | | | | | | | |
| o4-mini | 0.896 | 8.383 | 0.897 | 8.600 | 0.906 | 7.745 | 0.856 | 8.273 | 0.900 | 8.261 |
| GPT-4o | 0.832 | 8.617 | 0.834 | 8.100 | 0.826 | 7.618 | 0.789 | 7.745 | 0.830 | 8.035 |
| GPT-4o-mini | 0.896 | 7.617 | 0.897 | 7.183 | 0.906 | 6.400 | 0.856 | 6.636 | 0.821 | 6.978 |
| Claude-3.7-sonnet | 0.834 | 8.583 | 0.828 | 8.583 | 0.799 | 7.691 | 0.808 | 8.278 | 0.831 | 8.297 |
| Gemini-2.5-pro | 0.865 | 8.833 | 0.865 | 9.050 | 0.865 | 8.055 | 0.866 | 8.345 | 0.865 | 8.587 |
| <i>Annotator: Gemini-2.5-flash, Critique Judge: Claude-4.0-sonnet</i> | | | | | | | | | | |
| o4-mini | 0.896 | 7.457 | 0.897 | 7.383 | 0.906 | 7.189 | 0.856 | 7.727 | 0.900 | 7.439 |
| GPT-4o | 0.832 | 7.000 | 0.834 | 6.450 | 0.826 | 6.566 | 0.789 | 6.945 | 0.830 | 6.724 |
| GPT-4o-mini | 0.833 | 6.761 | 0.836 | 5.500 | 0.762 | 5.377 | 0.690 | 5.600 | 0.821 | 5.772 |
| Claude-3.7-sonnet | 0.834 | 7.652 | 0.828 | 7.250 | 0.799 | 7.038 | 0.808 | 7.636 | 0.831 | 7.383 |
| Gemini-2.5-pro | 0.865 | 7.978 | 0.865 | 8.300 | 0.865 | 7.566 | 0.866 | 7.800 | 0.865 | 7.920 |

Table 19: Model Performance ranking comparison under different combinations of annotation and judge models.

E Case study

Task: Math Parity

You are given a plot of a real-valued, scalar function $f(x)$. YOUR TASK is to determine whether $f(x)$ is an even function, an odd function, or neither.

- Definition of an even function: A function such that, $f(x) = f(-x)$, where the value remains unchanged if the sign of the independent variable is reversed.
- Definition of an odd function: A function such that, $f(-x) = -f(x)$, where the sign is reversed but the absolute value remains the same if the sign of the independent variable is reversed.- A function is neither even nor odd if it does not satisfy either definitions. Respond with 'even', 'odd', 'neither' on whether the function $f(x)$ is even, odd, or neither, based on the definition above.

Demonstration example(s) of the task:

Example 1: Example Response: [PLEASE OUTPUT YOUR REASONING].

Answer: odd. Answer the new question below.

The last part of your response should be of the following format: 'Answer: <YOUR ANSWER>' (without angle brackets) where YOUR ANSWER is your answer, following the same task logic and output format of the demonstration example(s). For your answer, do not output additional contents that violate the specified format. Think step by step before answering.

o4-mini Critique:

{ "correct": "Error",
 "critique": "The plotted function is symmetric about the vertical (y) axis: for every x-value on the right, the graph at $-x$ has the same $f(x)$ value (peak, trough, and baseline all mirror). That satisfies $f(-x) = f(x)$, the definition of an even function. An odd function would require $f(-x) = -f(x)$, which fails immediately (for example $f(0) \approx 0.6$ but $-f(0) \approx -0.6 \neq f(-0) = 0.6$). Therefore the correct classification is 'even,' not 'odd.'"}
↓

Judge:

{ "explanation": "The generated critique correctly identifies the error in the response by pointing out the symmetry about the y-axis and explains that this satisfies the even function definition $f(-x) = f(x)$, not the odd definition. It also points out where the oddness fails with a specific example ($f(0)$), adding a strong supporting detail. The reference critique also identifies the symmetry, clearly describes the error as misunderstanding, and notes the lack of reasoning in the respondent's answer. However, the generated critique provides more specific and concrete reasoning related to function values and definitions and includes a direct counterexample to the claim of oddness, which makes the critique even clearer and more educational. Since the generated critique is both accurate and more detailed on technical grounds compared to the reference, it deserves a high score.",
 "score": "9" }
↑

Reference Critique:

The response is incorrect because the function in the given plot satisfies the definition of an even function: $f(x) = f(-x)$. This is evident from the symmetry of the graph about the y-axis. The assistant incorrectly labeled the function as 'odd,' which is a significant misunderstanding of the task's requirements and definitions. Furthermore, the response failed to provide any reasoning or justification for the labeling, making it impossible to analyze an underlying thought process or identify where the misunderstanding occurred. Adding detailed reasoning would enhance clarity and demonstrate comprehension of the task.

Figure 10: An example of a visual mathematical reasoning task, where the response is clearly incorrect, demonstrates that the o4-mini model provides an accurate judgment along with a comprehensive textual critique. When evaluating its critique score, it is evident that the presence of the reference critique effectively guides the judge model to assign a high-quality score of 9, surpassing even the reference critique in some aspects.

Task:brand logo recognition and elaboration

Identify the brand logo presented in the query image. Also provide the country of origin (i.e., where the company was founded) of the brand. Remove all spaces and hyphens from the brand name. If the image does not contain a logo, answer NA for both fields.



Demonstration example(s) of the task:
Example 1:<image>Example Response:[PLEASE OUTPUT YOUR REASONING]Answer: {'brand name': 'RedBull', 'country of origin': 'Austria'}Answer the new question below. The last part of your response should be of the following format: "Answer: <YOUR ANSWER>" (without angle brackets) where YOUR ANSWER is your answer, following the same task logic and output format of the demonstration example(s). For your answer, do not output additional contents that violate the specified format. Think step by step before answering

Question:



Response:

[PLEASE OUTPUT YOUR REASONING]
The image shows a car with a large Red Bull energy drink can mounted on its roof. The brand name on the can is "Red Bull," and the country of origin for Red Bull is Austria. Answer: {"brand name": "Red Bull", "country of origin": "Austria"}



Reference Critique:

The response is entirely incorrect as it fails to correctly identify the logo in the query image. The brand shown in the image is Ferrari, a well-known automobile manufacturer, and its country of origin is Italy. Instead, the response incorrectly describes the image as a depiction of a Red Bull promotional car and attributes the brand and country of origin to Red Bull and Austria, respectively. This mistake indicates a lack of accurate identification of visual content, which is the core requirement of the task. Furthermore, the output format does not align properly with the expected format since the brand name 'Red Bull' was not concatenated to 'RedBull' as specified in the instructions. Improvement requires correctly analyzing the query image and providing accurate brand and origin details. The response needs to be consistent with the given task logic and example.

Qwen32b Critique:

```
{ "correct": "Error",
  "critique": "
```

"The response provided is incorrect because it does not match the content of the query image. Here is the step-by-step analysis of the issue and critique of the response:

1. Understanding the Query Image:

The query image shows a logo that is clearly distinct from any Red Bull branding. Instead, it features a horse emblem on a shield, accompanied by the initials 'SF'. This is a widely recognized emblem of the Ferrari brand, which is known for its association with luxury sports cars and motorsports.

2. Analysis of Incorrect Response:

The response identifies the brand as 'Red Bull' with the country of origin as 'Austria'. However, the logo in the query image is not the Red Bull logo. The logo is instead the Ferrari logo, which is entirely different both in design and branding elements.

3. Correct Interpretation:

Given the horse emblem and the initials 'SF', the brand in question is Ferrari. Ferrari was founded in Italy. Thus, the correct answer should have been 'Ferrari' for the brand name and 'Italy' for the country of origin.

4. Formatting Issues in the Response:

The brand name provided in the response contains a space ('Red Bull'), which violates the instruction to remove all spaces and hyphens from the brand name. This further indicates inconsistencies in following the task instructions.

5. Overall Evaluation:

The assistant failed to accurately identify the brand in the image, leading to both an incorrect brand name and country of origin. Additionally, they did not adhere to the format requirement of removing spaces from the brand name. Given these observations, the response is deemed incorrect."

Judge:

{ "explanation": "The generated critique provides a thorough analysis of the assistant's errors. It starts with explaining the content of the image, identifies the mistake in misidentifying the logo, and addresses both the factual inaccuracy and formatting issue (the retention of the space in 'Red Bull'). It also gives a clear summary of the assistant's failures, explicitly noting the deviation from instructions. Compared to the reference critique, the generated critique is slightly more detailed in its step-by-step analysis, such as mentioning the 'SF' initials and emphasizing the formatting rule violation. Both critiques are factually accurate and clearly point out the failure to correctly identify Ferrari and its Italian origin. **However, the generated critique contains more granular stepwise reasoning, making it marginally better and more instructive while still remaining concise and on-point.** Therefore, according to the rules provided, the score given should be slightly higher than 8.",
"score": "8"}

Figure 11: An example from the brand logo recognition and elaboration task, where the critique is generated by Qwen2.5-vl-32b. The model produces a detailed, step-by-step reasoning critique. When scored by the judge model, it explicitly explains its preference for lengthier, somewhat redundant reasoning—highlighted in bold red—demonstrating a bias toward richer textual justifications.

Task:stackoverflow_debug_QA

Given the screenshot code snippet, please answer 1. Which programming language is the code written in? 2. Whether the code can be executed correctly? The answer should be Yes or No. 3. If the code execution is unsuccessful, please find out line number of code snippet that first leads to the syntax or format error. If the code is correct, the answer is N/A. The line number is labeled before each line of code with red color

Demonstration example(s) of the task without concrete image, providing the expected output format:
Example 1:
Example Response:
[PLEASE OUTPUT YOUR REASONING]
Answer: {'Programming type': 'python', 'Programming execution': 'No', 'Error line': '3'}
Answer the new question below. The last part of your response should be of the following format:
"Answer: <YOUR ANSWER>" (without angle brackets) where YOUR ANSWER is your answer, following the same output format of the demonstration example(s). For your answer, do not output additional contents that violate the specified format. Think step by step before answering.

Question:

I'm trying to urlencode an array in python with urllib.urlencode. The problem is, I have to encode an array.

The result needs to be:

```
 criterias%5B%5D=member&criterias%5B%5D=issue
#unquoted: criterias=[]=member&criterias=[]=issue
```

But the result I get is:

```
criterias=%5B%27member%27%2C%27issue%27%5D
#unquoted: criterias=['member','issue']
```

I have tried several things, but I can't seem to get the right result.

```
1 import urllib
2 criterias = ['member', 'issue']
3 params = {
4     'criterias[]': criterias,
5 }
6 print urllib.urlencode(params)
```

If I use cgi.parse_qs to decode a correct query string, I get this as result:

```
{'criterias[]': ['member', 'issue']}
```

But if I encode that result, I get a wrong result back. Is there a way to produce the expected result?

Response:

The programming language used in the code snippet is Python. The code is attempting to encode a dictionary into a URL query string using the 'urllib' library, which is not a standard Python library. The code contains several syntax errors, such as incorrect use of the '+' operator, missing quotes, and incorrect function calls. The line number of the first error is 1, where the 'import' statement is incorrectly written as 'import urllib' instead of 'import urllib.parse' (assuming 'urllib' is a typo and the intended library is 'urllib.parse').

Answer: {'Programming type': 'Python', 'Programming execution': 'No', 'Error line': '1'}

claude-3.7-sonnet Critique:

```
{
"correct": "Error",
"critique": "The assistant's response contains several critical errors in its analysis. The code shown in the image DOES use the correct import statement 'import urllib', which is a valid Python library. The issue is not with the import statement itself but with how the user is trying to encode dictionary values to achieve a specific URL format. The code shown would execute without syntax errors - it's functionally correct Python code. The assistant incorrectly identified line 1 as having an error and misunderstood the user's actual problem, which is about getting the correct URL encoding format rather than fixing syntax errors. The code uses urllib.urlencode() which is valid (though in Python 3 this would be urllib.parse.urlencode()), but the user's issue is with the formatting of the resulting encoded string, not with syntax errors. The correct answer should indicate the code is syntactically correct Python code that would execute without errors, even if it doesn't produce the user's desired output format."
}
```

Judge:

```
{
"explanation": "The generated critique does a strong job highlighting the main mistakes in the assistant's response, such as misidentifying the import error and misunderstanding the actual issue with the code (which is functional, but does not produce the desired output, not a syntax error). It explicitly and accurately points out that the code would execute without syntax errors and correctly clarifies the distinction between functionality and desired formatting. Compared to the reference critique, the generated critique is slightly more focused and concise, directly addressing where the assistant went wrong and why. However, both critiques broadly cover the necessary points. The generated critique could have slightly improved by explicitly mentioning that the user's question is about the output format, not code execution errors, but overall, it provides a clearer and more direct rebuttal to the assistant's mistakes.",
"score": 9
}
```

Reference Critique:

The response correctly identifies the programming language as Python and notes that the code cannot be executed. However, it provides an incorrect analysis of the errors in the code. The claim that 'urllib' is not a standard Python library is incorrect, as 'urllib' is part of Python's standard library. Additionally, it raises an issue with the 'import' statement on line 1, which is valid and does not cause an error. The correct reason for the unsuccessful execution lies in line 4, where the 'urlencode' function in 'urllib' does not handle lists as expected. The assistant misses this and incorrectly identifies line 1 as the first point of failure. To improve, the response needs to better analyze the purpose and behavior of the 'urllib.urlencode' function and determine why it fails to produce the expected output in line 4. Lastly, the phrasing could be refined for clarity.

Figure 12: An example from the StackOverflow debug QA task, where the critique is generated by claude-3.7-sonnet. The critique provides a complete analysis and exactly points out the original response's error. Compared with the reference critique, the judge model accurately evaluates that Claude-3.7-sonnet, as a well-known pioneer model of coding, outperforms the reference critique.



Figure 13: An example from the GUI agent application task, where the critique is generated by Genimi-2.5-pro. It can accurately tell the correctness of the model's response, and the textual critique score is slightly below the anchored reference critique score (e.g., 8), where the judge model provides a reasonable explanation (marked in red).

Task: face_keypoint_detection

This task involves analyzing images to detect and locate 5 specific facial landmarks: the **left eye center**, **right eye center**, **nose tip**, **left mouth corner**, and **right mouth corner**. The goal is to accurately identify the positions of these key facial features by providing the normalized coordinates of these keypoints within each image. Each keypoint is [x, y], where x and y are the coordinates normalized based on the width and height and the picture. The output should be a sequence of 5 keypoint coordinates.



Demonstration example(s) of the task:
Example 1:<image> Example Response:
[PLEASE OUTPUT YOUR REASONING]
Answer:
[[0.38, 0.52], [0.62, 0.51],
[0.52, 0.62], [0.38, 0.7],
[0.6, 0.68]]
Answer the new question below. The last part of your response should be of the following format:

"Answer: <YOUR ANSWER>" (without angle brackets) where YOUR ANSWER is your answer, following the same task logic and output format of the demonstration example(s). For your answer, do not output additional contents that violate the specified format. Think step by step before answering.

Question:



Response A by internvl2_5_4b:

I'm unable to analyze or provide coordinates for facial landmarks in images. However, I can guide you on how to do it using tools like OpenCV in Python. If you need help with that, feel free to ask!

(response_quality: 2)

Response B by phi4_multimodal:

To solve this task, we need to identify the positions of the five specific facial landmarks: the left eye center, right eye center, nose tip, left mouth corner, and right mouth corner. We will use the normalized coordinates [x, y] for each keypoint, where x and y are the coordinates normalized based on the width and height of the image.
1. **Left Eye Center**: This is typically located near the midpoint of the left eye.
2. **Right Eye Center**: This is typically located near the midpoint of the right eye.
3. **Nose Tip**: This is typically located at the tip of the nose.
4. **Left Mouth Corner**: This is typically located at the corner of the mouth on the left side.
5. **Right Mouth Corner**: This is typically located at the corner of the mouth on the right side.
Given the image, we can estimate the positions of these keypoints. Since the image is not provided, I will use a hypothetical example to demonstrate the process.
Assuming the following normalized coordinates for the keypoints:
- Left Eye Center: [0.35, 0.5]
- Right Eye Center: [0.65, 0.5]
- Nose Tip: [0.5, 0.55]
- Left Mouth Corner: [0.35, 0.7]
- Right Mouth Corner: [0.65, 0.7]
Answer: [[0.35, 0.5], [0.65, 0.5], [0.5, 0.55], [0.35, 0.7], [0.65, 0.7]]
(response_quality: 6)

pixtral-large (24-11) Critique:

```
{  
  "choice": "B",
```

"explanation": "Response A fails to address the task requirements and instead offers an unrelated solution (using tools like OpenCV), which is not aligned with the prompt. On the other hand, Response B correctly understands the task, explains the reasoning step by step, and provides a structured and logical answer in the required format.

Therefore, Response B is the better and more relevant solution to the given problem.

```
}
```

Figure 14: An example of comparative critique from the face keypoint detection task, where the critique is generated by Pixtral-large(24-11). The original responses are scored for their response quality scores by the annotator model (GPT-4o). As a (low, medium) pairwise comparison, it is easy to distinguish the better one with a high-performance model.