A Systematic Survey and Critical Review on Evaluating Large Language Models: Challenges, Limitations, and Recommendations

Anonymous ACL Roling Review submission

Abstract

Large Language Models (LLMs) have recently gained significant attention due to their remarkable capabilities in performing diverse tasks across various domains. However, a thorough evaluation of these models is crucial before deploying them in real-world applications to ensure they produce reliable performance. Despite the well-established importance of evaluating LLMs in the community, the complexity of the evaluation process has led to varied evaluation setups, causing inconsistencies in findings and interpretations. To address this, we systematically review the primary challenges and limitations causing these inconsistencies and unreliable evaluations in various steps of LLM evaluation. Based on our critical review, we present our perspectives and recommendations to ensure LLM evaluations are reproducible, reliable, and robust.

1 Introduction

The evolution of LLMs has transitioned from simple generative models predicting the next word to advanced systems capable of following instructions and solving complex problems (Zhao et al., 2023a). Early models like GPT (Radford et al., 2018) could generate coherent text but were limited to simple tasks, whereas instruction-tuned LLMs (Chung et al., 2022; Ouyang et al., 2022) like ChatGPT[1] greatly enhanced their versatility and ability to execute specific commands. This shift has revolutionized the development of real-world applications powered by LLMs.

With the advancements and broad applicability of LLMs, it is essential to properly evaluate them to ensure they are safe to use. This is indeed important not only for academic benchmarks but also for business use cases. Consequently, understanding the bottlenecks of current evaluation methods, and developing strategies to address these challenges are crucial for standardizing evaluations and enabling reliable use of LLMs in practical applications. Nonetheless, evaluating LLMs is as complex and resource-intensive as their development, involving multiple levels or aspects.

Existing reviews (Chang et al., 2024; Guo et al., 2023b; Liang et al., 2022; Minaee et al., 2024; Zhuang et al., 2023) related to the evaluation of LLMs often focus only on benchmark tasks, datasets, and evaluation criteria, neglecting the broader complexities. This oversight can undermine the reliability of evaluation by ignoring issues like robustness and reproducibility. While some recent studies (Balloccu et al., 2024; Mao et al., 2023) have investigated data contamination (Ravaut et al., 2024) and evaluation malpractices in LLM evaluation, their focus is limited to only assessing ChatGPT, overlooking other LLMs, as well as the entire evaluation pipeline.

More recently, Biderman et al. (2024) discussed the reproducibility problem in existing evaluations of LLMs and introduced a library to address this. However, their work lacks comprehensive discussions on how aspects like reliability or robustness impact LLM evaluation and how to address them. Hence, existing LLM evaluation studies often focus on individual aspects in a scattered manner, resulting in findings that are only sparsely useful.

To mitigate this gap, this paper brings together the discussions to address the fundamental challenges and limitations in LLM evaluations that emerge from diverse evaluation setups. First, we craft a schematic workflow of the evaluation pipeline in practical settings (presented in Section 2) for a systematic study. We then examine each step in the evaluation workflow, uncovering various inconsistencies and decision-making complexities affecting reproducibility, reliability, and

---

[1] https://openai.com/index/chatgpt/

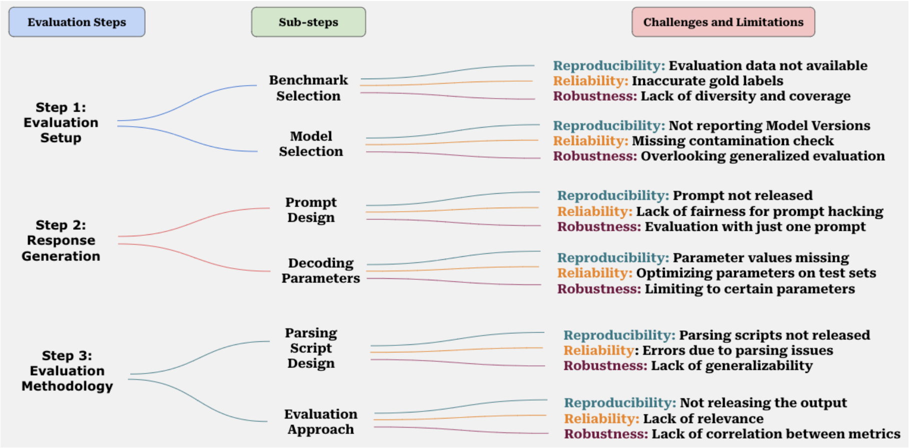

Evaluation Steps
Sub-steps
Challenges and Limitations
Step 1: Evaluation Setup
Benchmark Selection
Reproducibility: Evaluation data not available
Reliability: Inaccurate gold labels
Robustness: Lack of diversity and coverage
Model Selection
Reproducibility: Not reporting Model Versions
Reliability: Missing contamination check
Robustness: Overlooking generalized evaluation
Step 2: Response Generation
Prompt Design
Reproducibility: Prompt not released
Reliability: Lack of fairness for prompt hacking
Robustness: Evaluation with just one prompt
Decoding Parameters
Reproducibility: Parameter values missing
Reliability: Optimizing parameters on test sets
Robustness: Limiting to certain parameters
Step 3: Evaluation Methodology
Parsing Script Design
Reproducibility: Parsing scripts not released
Reliability: Errors due to parsing issues
Robustness: Lack of generalizability
Evaluation Approach
Reproducibility: Not releasing the output
Reliability: Lack of relevance
Robustness: Lack of correlation between metrics

Figure 1: Typology of the LLM Evaluation Workflow

robustness (see Section 3). Based on our findings, we provide a principled guideline in Section 4 to address current limitations in LLM evaluation.

2 Overview of LLM Evaluation Process

The following components are crucial for LLM evaluation: *Evaluation Setup*, *Response Generation*, and *Evaluation Methodology* (Chang et al., 2024). Each component has its own challenges, which we discuss in Section 3. These components in an evaluation workflow are shown in Figure 1.

2.1 Evaluation Setup

Benchmark Selection: To initiate the evaluation process of LLMs, the first step is selecting appropriate benchmarks. We categorize the benchmarking datasets into the following: *general capability benchmarks*, *specialized benchmarks*, and *other diverse benchmarks*. We refer to general capability benchmarks as the ones that are often used for evaluation upon the release of an LLM (e.g., MMLU (Hendrycks et al., 2020), HumanEval (Chen et al., 2021)). In addition, there are specialized benchmarks that measure specific capabilities of LLMs (e.g., MT-Bench for chatting capabilities (Zheng et al., 2024)). There are also other benchmarks that usually combine multiple benchmarks to evaluate LLMs on diverse task (e.g., HELM (Liang et al., 2022)). We provide more details on each category in Appendix A.1.

Model Selection: Selecting the appropriate model from the numerous LLMs currently available is crucial for ensuring a fair evaluation, as it helps to avoid risks such as data contamination and unfair comparisons. For a detailed discussion on prominent LLMs, see Appendix A.2.

2.2 Response Generation

Once the benchmarks and the models are selected, the next step in the evaluation process is to design the prompt and set up the decoding parameters for response generation. In the ***prompt design*** step, decisions on what type of prompting (e.g., zero-shot or few-shot) would be used are taken. Moreover, configuring the ***decoding parameters*** (e.g., temperature) is important to ensure optimal performance (Shi et al., 2024). More discussions on this are provided in Appendix A.3 and A.4.

2.3 Evaluation Methodology

Parsing Script Design: Evaluating LLM-generated responses is difficult because they often produce verbose outputs (see Table 4 for some examples). Therefore, parsing scripts are often necessary (Jahan et al., 2024; Laskar et al., 2023a) to extract target labels before applying evaluation metrics, ensuring alignment with evaluation criteria to maintain reliability.

Evaluation Approach: The evaluation approach can be divided into the following: *automatic evaluation*, *human evaluation*, *LLMs as evaluators*. In *automatic evaluation*, before applying task-specific metrics (e.g., F1, Exact Match, Perplexity (Jelinek et al., 1977)), parsing scripts are often utilized to extract the targeted answer, especially in discriminative tasks. *Human evaluation* is required to ensure qualitative assessments of LLM responses (e.g., measuring clarity, coherence, factuality) (van der Lee et al., 2021). Recently, human evaluation based on the Elo-based rating system (Zheng et al., 2024) has gained a lot of attention. Since human evaluation is time-consuming, the utilization of *LLMs as evaluators* to assess other LLMs has become a popular evaluation approach (Chiang and Lee, 2023; Huang et al., 2024a). More details on LLM evaluation approaches are in Appendix A.6.1.

Availability (%)				Comparison (%)	
Prompt	Code	Prompt + Code	Model Version	Fair	Unfair
90.6	53.3	50.0	29.3	20.7	79.3

Table 1: Availability of resources and fairness in model comparisons (out of 212 papers), analyzed by Balloccu et al. (2024).

3 Challenges in Evaluating LLMs

We examine challenges and limitations in the evaluation process of LLMs based on three dimensions: *reproducibility*, *reliability*, and *robustness*.

3.1 Reproducibility

Reproducibility, the ability to consistently replicate model results under the same conditions, is a major challenge in generative models (Biderman et al., 2024). The primary challenge is the lack of comprehensive documentation for each part of the evaluation cycle, including benchmarking datasets, prompt construction, model details, decoding strategy, response parsing, and evaluation methodology (Kosch and Feger, 2024; McIntosh et al., 2024). Table 1 presents an analysis by Balloccu et al. (2024), revealing that a relatively low percentage of the analyzed papers shared their resources. Below, we discuss factors impacting reproducibility in the evaluation step.

3.1.1 Missing Details on Data & Models Used

Benchmarking Data: One factor that can negatively impede the ability to reproduce results is not releasing the exact data used for evaluation (Balloccu et al., 2024). Many studies evaluate LLMs on only a subset of existing datasets (Bang et al., 2023; Kocoń et al., 2023), while others use the exact benchmarking datasets (Laskar et al., 2023a; Qin et al., 2023). Despite the expectation not to compare results across studies using different subsets of the data, such comparisons often occur, as discussed by Balloccu et al. (2024). Nonetheless, without explaining the sampling strategy, or releasing the subsets used for evaluation (and possibly their responses), reproducing results using different data subsets of the same size is challenging.

Model Versions: The information regarding the version of a model being used is also missing in many studies (Balloccu et al., 2024; Biderman et al., 2024), creating reproducibility concern (see Table 1). The continuous updates of the closed-source models, often with undisclosed changes can also impact reproducibility. With these updates, earlier versions are often deprecated, and results from these versions may not apply to newer models (Chen et al., 2023b), making prior evaluation results to be no longer reproducible (Bang et al., 2023; Kocoń et al., 2023; Laskar et al., 2023a; Qin et al., 2023). Therefore, it is crucial to specify the model versions used (Balloccu et al., 2024; Biderman et al., 2024), while model owners should keep earlier versions available.

3.1.2 Lacking Response Generation Details

Prompting: The lack of details behind how the prompts are designed may make the findings in different literature inconsistent. For instace, variations in prompt design can lead to significantly different results, as seen in various studies (Bang et al., 2023; Jahan et al., 2024; Laskar et al., 2023a; Qin et al., 2023). While few-shot learning is found to outperform zero-shot in the original evaluation conducted by the authors of various LLMs (Anil et al., 2023; OpenAI, 2023; Touvron et al., 2023b), many independent evaluations demonstrate that adding few-shot examples does not necessarily outperform zero-shot models in every task (Jahan et al., 2024; Ye et al., 2023a). This raises the concern of whether certain prompt engineering techniques or optimizations to select few-shot samples were applied in the original evaluations. Hence, not disclosing the details behind how the prompt is designed or how the few-shot examples are selected can hinder reproducibility.

Decoding Strategy: LLMs are sensitive to decoding parameters, leading to significant performance variations based on the chosen settings (Roziere et al., 2023; Touvron et al., 2023b). However, crucial details on their selection are excluded in existing literature (Bang et al., 2023; Kocoń et al., 2023; Laskar et al., 2023a; OpenAI, 2023; Qin et al., 2023; Team et al., 2023). This lack of transparency raises reproducibility concerns, which could be responsible for inconsistent results across studies even when similar prompts are used. For instance, Qin et al. (2023) found that adding output length restrictions in the prompt to generate summaries in no more than N words led to a performance drop in the SAMSum dataset (Gliwa et al., 2019). However, Laskar et al. (2023a) found that such controlled experiments led to a gain in performance in the SAMSum dataset.

3.1.3 Evaluation Methods Unavailable

Parsing Scripts: LLM-generated responses often require parsing scripts to extract desired information. However, as demonstrated in Table 1, Balloccu et al. (2024) observed in their analysis that almost half of the LLM evaluation papers do not release any codes. We also observe that most studies (these include both the LLM technical reports, as well independent evaluations) do not release their parsing scripts (Bang et al., 2023; Kocoń et al., 2023; OpenAI, 2023; Qin et al., 2023; Team et al., 2023, 2024). Nonetheless, inaccurate design of parsing scripts may lead to different evaluation results (Laskar et al., 2023a). Thus, the unavailability of parsing scripts would complicate result comparisons while impacting reproducibility (Balloccu et al., 2024; Biderman et al., 2024).

Evaluation Approach: LLMs are increasingly used to evaluate other LLMs in development (Zheng et al., 2024). Concerns arise due to the use of closed-source LLMs as evaluators, as their frequent updates can affect reproducibility (Chen et al., 2023b; Verga et al., 2024). Moreover, Chen et al. (2023b) observed significant behavioral changes in closed-source LLMs over short periods. Such reproducibility concerns are also observed in prior research that used LLMs as evaluators. For instance, Chiang and Lee (2023); Zheng et al. (2024) found that using closed-source LLMs as the judge could collide with human evaluations, whereas Fu et al. (2023b) observed the opposite. Since the recently proposed Prometheus-2 (Kim et al., 2024) model is an open-source alternative and demonstrates a strong correlation with humans, utilizing open-source LLMs as the judge can help mitigate the reproducibility issues prevalent with closed-source LLMs.

3.2 Reliability

Reliability, the ability to trust that outcomes are as intended, is another challenge encountered during evaluation. Issues like contamination/inaccurate labels in the data, irrelevant evaluation methods, and unfair comparisons may impact the reliability of the findings, which we discuss below.

3.2.1 Data and Model Integrity Issues

Data Integrity: Errors in benchmarks undermine accurate conclusions and model comparisons, rendering evaluations of LLMs unreliable. An integrity-compromising factor is the presence of incorrect gold labels. For instance, existing issues in the gold labels of the widely used MMLU (Hendrycks et al., 2020) dataset have led to the development of MMLU-Pro (Wang et al., 2024) and MMLU-Redux (Gema et al., 2024). Recently it was also found that the coding benchmarks, HumanEval (Chen et al., 2021), lacked essential test cases, leading to the development of an advanced version, HumanEvalPlus (Liu et al., 2024b).

Despite these improvements, many recent studies continue to use the older versions of datasets. For instance, despite the release of HumanEvalPlus, HumanEval is still used to benchmark LLM coding performance (Gloeckle et al., 2024; Jiang et al., 2023; Li et al., 2023c; Roziere et al., 2023; Team et al., 2023, 2024; Wong et al., 2023), potentially providing misleading insights. In addition, outdated labels in existing benchmarks undermine reliability of gold references. For example, in tasks like open-domain question answering, which demand real-world knowledge, many gold labels become outdated over time, as noted by Laskar et al. (2023a). Consequently, even if LLMs produce correct answers, comparing them to obsolete gold labels can yield inaccurate results. Moreover, in tasks like summarization, LLM-generated summaries are often favored over human-annotated gold references (Ding et al., 2022; Pu et al., 2023; Zhang et al., 2024b).

Contamination in Existing Models: Contamination occurs when a benchmarking dataset is used in training, reducing result reliability and validity (Sainz et al., 2023a). Ensuring benchmarking examples are excluded from training data is essential to maintain reliable results. Since LLMs are pre-trained on vast amounts of text data available on the internet, this could lead to unfair evaluations if LLMs have already encountered these datasets during their pre-training phase (Balloccu et al., 2024; Ravaut et al., 2024; Xu et al., 2024).

Nonetheless, most prior LLM evaluation work focusing on zero-shot evaluation did not conduct any data contamination tests (Bang et al., 2023; Laskar et al., 2023a; OpenAI, 2023; Qin et al., 2023; Team et al., 2023), raising concerns about whether these evaluations truly represent the zero-shot capabilities of LLMs. Recent research has also demonstrated a strong possibility of data contamination in many datasets used to evaluate different LLMs (Balloccu et al., 2024; Golchin and Surdeanu, 2023; Li and Flanigan, 2023; Oren et al., 2023; Ravaut et al., 2024; Sainz et al., 2023b; Xu et al., 2024; Zhang et al., 2024a). With the current generation of LLMs being extremely capable of learning new skills with minimal amounts of data, exposing them to evaluation data may undermine the measurement of their true capabilities. Since the possibility of data contamination has led to the development of new versions of existing datasets (e.g., utilizing GSM-8K to construct GSM-1K (Zhang et al., 2024a)), it is crucial to use reliable and fair evaluation datasets.

3.2.2 Lack of Fairness by Manipulating Response Generation

Prompt Hacking: One major concern in terms of lack of fairness in LLM evaluation is the possibility of prompt hacking (Schulhoff et al., 2023), which involves manipulating input prompts to a language model to elicit desired responses (e.g., biasing the outputs, or taking unfair advantages by using specific few-shot examples). While the performance of LLMs depends on many factors relevant to how the prompt is structured, most work (Bang et al., 2023; Laskar et al., 2023a; Qin et al., 2023), even the official technical reports (Anthropic, 2024; OpenAI, 2023; Team et al., 2023) of different LLMs lack the necessary details behind prompt construction (e.g., missing scientific validity on why a certain prompt was preferred over others, how the few-shot examples are selected, etc.). This makes the claims regarding the effectiveness and limitations of certain LLMs in comparison to others questionable[2]. Recognizing these parallels underscores the need for transparency and robust methodologies to ensure fairness in AI research and development.

Lack of Transparency in Decoding Parameters: Shi et al. (2024) demonstrated that extensive tuning of decoding parameters could improve the performance during inference. However, how the different decoding parameters are selected is often underexplored in existing evaluations (Bang et al., 2023; Laskar et al., 2023a,b; OpenAI, 2023; Qin et al., 2023; Team et al., 2023), as discussed in Section 3.1. This poses the risk of optimizing the parameters on test sets to improve performance.

3.2.3 Inappropriate Evaluation Methodology

Inaccurate Design of Parsing Scripts: As Laskar et al. (2023a) observed, evaluating LLMs entirely with an automated approach based on the answer extracted using parsing scripts may lead to an error of up to more than 10% difference in many tasks. This raises questions about the reliability of LLM evaluations that solely depend on parsing scripts without validating the scripts' effectiveness for the task. To tackle this, Laskar et al. (2023a) proposed a hybrid approach combining parsing script-based automatic evaluation with human-in-the-loop. Initially, the parsing script extracts answers from LLM-generated responses. If any issues arise, humans resolve them, enhancing the reliability of parsing-based automatic evaluation.

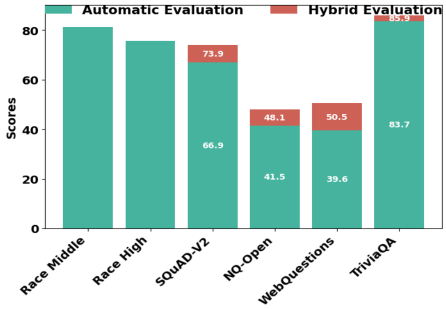


Figure 2: Comparing Automatic and Hybrid Evaluation.

In Figure 2, we demonstrate the differences between automatic and hybrid evaluation in Open-Domain QA[3] and reading comprehenesion datasets[4]. The figure highlights the influence of human intervention on results in open-domain QA, where LLMs may generate synonymous or time-sensitive correct answers, potentially rendering gold answers outdated (Laskar et al., 2023a). Parsing script-based automatic evaluation is found to be reliable in Race datasets for reading comprehension, whereas notable discrepancies are observed in the SQuAD-V2 dataset. Therefore, there's a need for designing dependable parsing scripts and involving humans when appropriate.

Evaluation Approaches Lacking Relevancy: In *generative tasks*, utilizing automatic string-based matching techniques may not be reliable as well. For instance, Laskar et al. (2023a) observed that despite LLMs scoring quite poorly on the ROUGE metric compared to SOTA summarization models, humans often prefer LLM-generated responses. Moreover, recent research observed potential biases while using LLMs as evaluators, such as LLMs preferring responses generated by LLMs of the same series, positional bias (Bai et al., 2024; Stureborg et al., 2024; Wang et al., 2023b; Wu and Aji, 2023). To mitigate this, Verga et al. (2024) proposed a new technique that leveraged multiple LLMs as juries instead of using a single LLM as the judge. This approach demonstrates higher cor-

---

[2] `https://crfm.stanford.edu/2024/05/01/helm-mmlu.html`

[3] NQ-Open (Kwiatkowski et al., 2019), WebQuestions (Talmor and Berant, 2018), TriviaQA (Joshi et al., 2017))

[4] SQuAD-V2 (Rajpurkar et al., 2018), Race-High and Race-Middle (Lai et al., 2017)

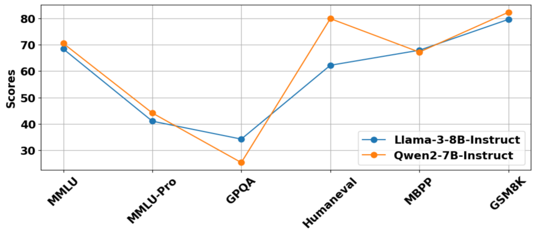


Figure 3: Performance Comparison: LLaMA-3 and Qwen2

Tokenizer	Vocab	MMLU	MMLU-Pro	MixEval	MixEval-Hard
LLaMA-2	32,000	0.52	0.45	0.29	0.11
LLaMA-3	128,256	0.27	0.21	0.09	0.03
Mistral	32,000	0.59	0.51	0.31	0.11
Qwen2	151,646	0.22	0.17	0.08	0.02

Table 2: Comparison of vocabulary coverage across different datasets and LLM tokenizers. The scores represent the percentage of tokenizer vocabulary that is covered by the respective dataset.

relations with humans, while mitigating biases.

3.3 Robustness

While there are many evaluation benchmarks currently available, existing work mostly relies on evaluating LLMs on some common benchmarks, this raises the question of whether the performance of LLMs in these common benchmarks in existing settings reflects their true capabilities and limitations. In this section, we study the robustness of existing LLM evaluations.

3.3.1 Lacking Generalized Evaluation

Limiting Evaluation to Certain Scenarios: Interestingly, it has been observed in recent research that certain performance gains in a specific dataset may not necessarily imply that it would also improve the performance in other datasets for similar tasks (Jahan et al., 2024; SambaNova, 2024). For instance, Jahan et al. (2024) observes that not a single LLM has superiority over other LLMs across all biomedical datasets and tasks. This is also evident if we compare the results between LLaMA-3 and Qwen2 reported in (Qwen2, 2024). As shown in Figure 3, while the Qwen2 model outperforms LLaMA-3 on most datasets, it falls short on GPQA and MBPP. Interestingly, for coding tasks, Qwen2 significantly outperforms LLaMA-3 on the HumanEval dataset (Chen et al., 2021) but not on the MBPP dataset (Austin et al., 2021). Meanwhile, existing common benchmarks also do not take into account some specific settings, such as how LLMs perform in long context scenarios, as recent research demonstrated that LLMs often struggle to generate the correct answer when relevant information does not appear at the beginning or end of the input context (Liu et al., 2024d). This highlights the importance of evaluating the generalized performance of LLMs across a set of diverse benchmarks and settings,instead of limiting evaluation to only common benchmarks like MMLU (Hendrycks et al., 2020).

Diversity and Coverage in Benchmarks: Although benchmarking datasets are designed to address specific problems and objectives, the variation and complexity of language within these datasets are often unclear. Liang et al. (2022) highlighted that better coverage in benchmarking datasets would enhance the comprehensiveness of the model's evaluation. While different language models use different tokenizers to represent the benchmarking dataset, it also leads to variations in what is evaluated across models.

As can be seen in Table 2, we conducted a small-scale analysis for LLaMA-2 (Touvron et al., 2023b), LLaMA-3,[5] Mistral (Jiang et al., 2023), and Qwen2[6] on two benchmarking datasets with varying complexities: MMLU (Hendrycks et al., 2020) and its more challenging version, MMLU-Pro (Wang et al., 2024), as well as MixEval (Ni et al., 2024) and its harder version, MixEval-Hard. Our findings indicate that these datasets cover a relatively small portion of the model's capabilities. Specifically, for MixEval, as the datasets became more diverse and dynamic, the vocabulary coverage for the tokenizer decreased. This trend continued as the datasets increased in difficulty, with vocabulary coverage further declining.

3.3.2 No Tuning of Prompt and Decoding Parameters

While various combinations of decoding parameters may lead to differences in results (Shi et al., 2024), possibly due to high computing requirements, existing LLM evaluation work mostly undermines the necessity of evaluating how the model performance may vary depending on its variations. Similar to the absence of decoder parameter tuning, most prior work also evaluated LLMs using only a single prompt (Bang et al., 2023; Jahan et al., 2024; Kocoń et al., 2023; Laskar et al., 2023a; Qin et al., 2023). However, in the real world, users express themselves with diverse word choices, varying semantics and syntaxes, alongside minor discrepancies (e.g., misspellings or differing punctuation styles). To fur-

---

[5] `https://llama.meta.com/llama3/`

[6] `https://github.com/QwenLM/Qwen2`

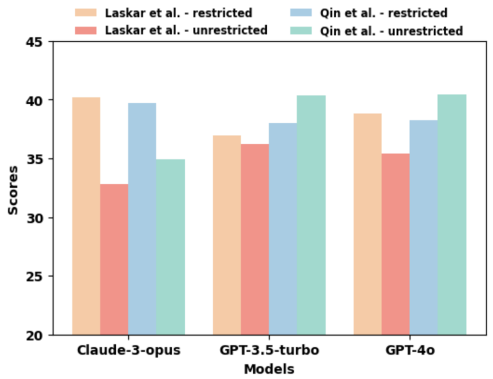


Figure 4: Performance in SAMSum based on Prompt Tuning.

ther examine the effects of prompt variations, we conduct an experiment using GPT-4o (2024-04-09) and GPT-3.5-Turbo (0125) (OpenAI, 2023), as well as Claude-3-Opus (2024-02-29) (Anthropic, 2024) with the prompts used by (Laskar et al., 2023a) and (Qin et al., 2023) in the SAMSum dataset. For this experiment, the default parameters for respective LLMs are used.

As shown in Figure 4, the restricted prompting method by Laskar et al. (2023a) consistently outperforms the unrestricted approach across all three models. Conversely, the restricted prompting method by Qin et al. (2023) fails to surpass the unrestricted approach for GPT-3.5 and GPT-4o. However, it surprisingly outperforms the unrestricted method, indicating the significant impact of prompt tuning across models. Evaluating language models with a single prompt lacks fairness (Zhu et al., 2023b), yet it remains common practice (Bang et al., 2023; Laskar et al., 2023a; Qin et al., 2023). Minor prompt variations can lead to diverse outcomes for different models (Alzahrani et al., 2024; An et al., 2023; Biderman et al., 2024; Sclar et al., 2023; Zhang et al., 2024a), highlighting the need to compare benchmarks across multiple prompts. Using automated prompt tuning techniques like Meta Probing Agents (Zhu et al., 2024) can ensure robustness to prompt variations.

3.3.3 Evaluation Method's Generalizability and Correlation Shortcomings

While automatic evaluations are usually utilized in discriminative tasks, they may not be applicable to every task, as demonstrated by Jahan et al. (2024) that parsing scripts are not usable in certain discriminative tasks like relation extraction. Jahan et al. (2024) also noted a significant performance gap between the string-matching-based ROUGE metric (Lin, 2004a) and the contextual similarity-based metric BERTScore (Zhang et al., 2019) in text summarization. While larger models achieve better accuracy, they involve a speed-accuracy trade-off, leading to higher costs and latency (Fu et al., 2024b; Laskar et al., 2023b). While metrics like perplexity are widely used to evaluate language models (Chen et al., 2023c), Huang et al. (2024b) found that quantized LLaMA-3 versions have lower output confidence than the original. They noted similar model rankings for perplexity and a common-sense QA dataset. However, Hu et al. (2024) found no correlation between perplexity and long context understanding tasks, highlighting the need for robust evaluations with human-correlated metrics.

Model	Chatbot Arena	HELM MMLU	Vellum MMLU
GPT-4o-2024-05-13	1 (1)	2 (2)	1 (1)
GPT-4-Turbo-2024-04-09	5 (3)	3 (3)	3 (3)
GPT-4-0125-preview	6 (4)	5 (5)	4 (4)
Gemini-1.5-Pro	4 (2)	4 (4)	13 (6)
Gemini-1.5-Flash	10 (6)	10 (6)	10 (5)
Claude-3-Opus-2024-02-29	7 (5)	1 (1)	2 (2)

Table 3: Rankings of models on LMSys Chatbot Arena vs two MMLU implementations. The relative rank of each model in MMLU is shown in parentheses.

This raises another question, whether automated evaluations and LLM-as-a-judge correlate with human evaluations (e.g., Elo ratings). Zheng et al. (2023) demonstrated significant correlations between Elo ratings, LLM-as-a-judge, and automated evaluations. However, recent research (Alzahrani et al., 2024) suggest that automated evaluations, especially those using multiple-choice questions, can yield unstable rankings with minor changes in evaluation methods. Given this instability, it prompts us to question why these automated tests should align with human Elo ratings despite demonstrating such inconsistencies. In our view, we should focus not only on correlating scores but also on how well a benchmark's rankings align with the gold standards. Analysis in Table 3 for GPT-4 (OpenAI, 2023), Gemini (Team et al., 2023), and Claude-3 (Anthropic, 2024) reveals two key observations: (i) MMLU rankings disagree with LMSys Chatbot Arena and (ii) MMLU rankings vary among themselves due to implementation differences.

4 Recommendations and Best Practices

We've outlined the primary challenges in evaluating LLMs thus far. In light of these challenges, a crucial question arises: How can we enhance the evaluation process for LLMs? Crafting a structured framework that's both practical and easy to implement is daunting, given the complexities of generative LLM development. Previous studies have tended to focus on specific evaluation aspects without offering comprehensive guidelines for the entire evaluation cycle, leaving researchers without clear guidance. Before diving into recommendations for each evaluation stage, it's important to acknowledge three key factors shaping current LLM evaluation practices: inherent randomness in generative models, significant computational demands, and insufficient documentation across stages.

Evaluation Setup: Selecting benchmarks for model assessment is crucial. Rather than simply replicating past choices, researchers should align datasets with required capabilities. To ensure *robustness*, datasets should vary across expected LLM capabilities (e.g., long-context understanding), tasks (e.g., summarization), and language complexity (e.g., vocabulary coverage). Ideally, a metric should measure dataset diversity. For model selection, conduct contamination tests between the chosen model and benchmarks using relevant techniques (Ravaut et al., 2024). This acts as an additional filter for benchmarking datasets, ensuring selection of unseen ones measuring intended capabilities. Meanwhile, for *reproducibility*, document any subset use of benchmarking datasets, along with the selected model version. In addition, throughout scientific history, intelligence progress has evolved across generations. Tests from a decade ago may appear simplistic compared to today's standards (e.g., Math Olympiads, ICPC programming contests). Refreshing LLM evaluations periodically can effectively communicate standard capabilities in both open and closed-source LLM markets and ecosystems (e.g., chatbots, translation tools). Hence, to ensure *reliability*, verify if the dataset has updated versions and incorporate them if available (e.g., HumanEvalPlus (Liu et al., 2024b), MMLU-Pro (Wang et al., 2024))

Response Generation: For *reproducibility*, thorough documentation of prompts and parameter settings is essential (e.g., specifying how few-shot samples are selected). To ensure *reliability*, it's crucial to justify why specific prompts and parameter settings are chosen over others, and provide comparisons with alternative options. As for *robustness*, running experiments with diverse prompts and parameters is key to showcasing their effectiveness and limitations across different scenarios. In resource-constrained environments, conducting experiments with diverse evaluation settings may pose challenges, yet it remains vital to perform robust evaluations on at least a subset of samples.

Evaluation Methodology: To ensure *reproducibility*, the parsing scripts and the output data used for evaluation should be published. Meanwhile, sanity-checking on the parsing script should be done to ensure *reliability* and *robustness* of the designed parsing script. This can be done by creating test cases for various response types, and then verifying (with human intervention if possible) whether the parsing script can reliably extract the targeted answer from the generated response. Meanwhile, reliance on string-based metrics like ROUGE should be minimized in favor of qualitative evaluations to ensure the *reliability* of the chosen evaluation methodology. Given the cost and time constraints of human qualitative evaluation, LLM-based evaluators can be used as alternatives but must be validated for potential biases (e.g., multiple LLMs as juries instead of using a single LLM as the judge (Zheng et al., 2024)). Finally, *robust* evaluation using task-specific metrics is encouraged with the metrics that lack alignments with humans should be avoided.

5 Conclusions and Future Work

In this paper, we systematically survey the challenges and limitations in evaluating LLMs. We identified significant inconsistencies and complexities at various stages of the evaluation pipeline, impacting the reproducibility, reliability, and robustness of the results. These issues underline the necessity for a standardized and systematic approach to evaluating LLMs to ensure their reliable usage in real-world applications. By comprehensively reviewing the current evaluation practices, we have provided a set of recommendations aimed at enhancing the consistency and reliability of LLM evaluations. Therefore, future work should focus on developing and adopting standardized evaluation protocols for LLMs to address the identified inconsistencies and complexities. This includes creating benchmark datasets, evaluation metrics, and proper documentation of the evaluation settings to ensure reproducibility, reliability, and robustness.

Limitations

One limitation of this work is that it is focused only on the evaluation phase of the LLM development cycle. Therefore, the challenges and limitations that happen during the training phase of LLMs are left out of the scope of this paper. Nonetheless, with the rapid growth of LLM technologies and huge financial incentives, it is essential to conduct a fair and reliable evaluation of LLM, alongside ensuring robustness and reproducibility, which is the focus of this work.

Another limitation of this study is that it does not study how to prevent closed-source LLMs from getting access to the online benchmarks. For instance, assume we have two entities: model developers and evaluators. Evaluators do not want to expose their data to the modeling team. Conversely, model developers do not want to release their model weights due to significant financial incentives. If evaluators use an API to get the responses, there is a risk that the queries may get exposed to the model developers. Therefore, without getting access to the weights, evaluators cannot reliably assess the models on their queries. Mathematically and technically, there is no fundamental way to solve this problem without altering the training dynamics which may not be an option for training teams.

Finally, the multimodal capability, in other words, the ability to understand both language and vision is another interesting capability of recently proposed LLMs (Bai et al., 2023; Chen et al., 2023a; Dai et al., 2024; Liu et al., 2023b, 2024a; Luo et al., 2024; Ye et al., 2023b; Zhang et al., 2023; Zhu et al., 2023a). This has led to the development of many multi-modal benchmarks (Chen et al., 2024b; Fu et al., 2023a, 2024a; Guan et al., 2023; Li et al., 2023a,b,d; Liu et al., 2024a, 2023d; Lu et al., 2022; Qiu et al., 2024; Yu et al., 2023). However, this paper was mostly focused on text-based NLP tasks and the evaluation of LLMs on multimodal benchmarks is left out for future work.

6 Ethics Statement

This paper only reviews the existing challenges and limitations in LLM evaluations and provides an opinion piece and recommendation to ensure reliable, robust, and reproducible evaluations of LLMs. Thus, this review does not pose any ethical concerns.

References

Marah Abdin, Sam Ade Jacobs, Ammar Ahmad Awan, Jyoti Aneja, Ahmed Awadallah, Hany Awadalla, Nguyen Bach, Amit Bahree, Arash Bakhtiari, Harkirat Behl, et al. 2024. Phi-3 technical report: A highly capable language model locally on your phone. *arXiv preprint arXiv:2404.14219*.

Ebtesam Almazrouei, Hamza Alobeidli, Abdulaziz Alshamsi, Alessandro Cappelli, Ruxandra Cojocaru, Mérouane Debbah, Étienne Goffinet, Daniel Hesslow, Julien Launay, Quentin Malartic, et al. 2023. The falcon series of open language models. *arXiv preprint arXiv:2311.16867*.

Iñigo Alonso, Maite Oronoz, and Rodrigo Agerri. 2024. Medexpqa: Multilingual benchmarking of large language models for medical question answering. *arXiv preprint arXiv:2404.05590*.

Norah Alzahrani, Hisham Abdullah Alyahya, Yazeed Alnumay, Sultan Alrashed, Shaykhah Alsubaie, Yusef Almushaykeh, Faisal Mirza, Nouf Alotaibi, Nora Altwairesh, Areeb Alowisheq, M Saiful Bari, and Haidar Khan. 2024. When benchmarks are targets: Revealing the sensitivity of large language model leaderboards.

Shengnan An, Bo Zhou, Zeqi Lin, Qiang Fu, Bei Chen, Nanning Zheng, Weizhu Chen, and Jian-Guang Lou. 2023. Skill-based few-shot selection for in-context learning. *arXiv preprint arXiv:2305.14210*.

Rohan Anil, Andrew M Dai, Orhan Firat, Melvin Johnson, Dmitry Lepikhin, Alexandre Passos, Siamak Shakeri, Emanuel Taropa, Paige Bailey, Zhifeng Chen, et al. 2023. Palm 2 technical report. *arXiv preprint arXiv:2305.10403*.

Anthropic. 2024. The claude 3 model family: Opus, sonnet, haiku.

Jacob Austin, Augustus Odena, Maxwell Nye, Maarten Bosma, Henryk Michalewski, David Dohan, Ellen Jiang, Carrie Cai, Michael Terry, Quoc Le, et al. 2021. Program synthesis with large language models. *arXiv preprint arXiv:2108.07732*.

Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. 2023. Qwen-vl: A frontier large vision-language model with versatile abilities. *arXiv preprint arXiv:2308.12966*.

Yushi Bai, Jiahao Ying, Yixin Cao, Xin Lv, Yuze He, Xiaozhi Wang, Jifan Yu, Kaisheng Zeng, Yijia Xiao, Haozhe Lyu, et al. 2024. Benchmarking foundation models with language-model-as-an-examiner. *Advances in Neural Information Processing Systems*, 36.

Simone Balloccu, Patrícia Schmidtová, Mateusz Lango, and Ondřej Dušek. 2024. Leak, cheat, repeat: Data contamination and evaluation malpractices in closed-source llms. *arXiv preprint arXiv:2402.03927*.

Yejin Bang, Samuel Cahyawijaya, Nayeon Lee, Wenliang Dai, Dan Su, Bryan Wilie, Holy Lovenia, Ziwei Ji, Tiezheng Yu, Willy Chung, et al. 2023. A multitask, multilingual, multimodal evaluation of chatgpt on reasoning, hallucination, and interactivity. *arXiv preprint arXiv:2302.04023*.

Stella Biderman, Hailey Schoelkopf, Lintang Sutawika, Leo Gao, Jonathan Tow, Baber Abbasi, Alham Fikri Aji, Pawan Sasanka Ammanamanchi, Sidney Black, Jordan Clive, Anthony DiPofi, Julen Etxaniz, Benjamin Fattori, Jessica Zosa Forde, Charles Foster, Mimansa Jaiswal, Wilson Y. Lee, Haonan Li, Charles Lovering, Niklas Muennighoff, Ellie Pavlick, Jason Phang, Aviya Skowron, Samson Tan, Xiangru Tang, Kevin A. Wang, Genta Indra Winata, François Yvon, and Andy Zou. 2024. Lessons from the trenches on reproducible evaluation of language models.

Yonatan Bisk, Rowan Zellers, Jianfeng Gao, Yejin Choi, et al. 2020. Piqa: Reasoning about physical commonsense in natural language. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 7432–7439.

Meriem Boubdir, Edward Kim, Beyza Ermis, Sara Hooker, and Marzieh Fadaee. 2023. Elo uncovered: Robustness and best practices in language model evaluation. *arXiv preprint arXiv:2311.17295*.

Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*.

Yupeng Chang, Xu Wang, Jindong Wang, Yuan Wu, Linyi Yang, Kaijie Zhu, Hao Chen, Xiaoyuan Yi, Cunxiang Wang, Yidong Wang, Wei Ye, Yue Zhang, Yi Chang, Philip S. Yu, Qiang Yang, and Xing Xie. 2024. A survey on evaluation of large language models. *ACM Trans. Intell. Syst. Technol.*, 15(3).

Jiawei Chen, Hongyu Lin, Xianpei Han, and Le Sun. 2024a. Benchmarking large language models in retrieval-augmented generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 17754–17762.

Lin Chen, Jinsong Li, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Zehui Chen, Haodong Duan, Jiaqi Wang, Yu Qiao, Dahua Lin, et al. 2024b. Are we on the right way for evaluating large vision-language models? *arXiv preprint arXiv:2403.20330*.

Lin Chen, Jisong Li, Xiaoyi Dong, Pan Zhang, Conghui He, Jiaqi Wang, Feng Zhao, and Dahua Lin. 2023a. Sharegpt4v: Improving large multi-modal models with better captions. *arXiv preprint arXiv:2311.12793*.

Lingjiao Chen, Matei Zaharia, and James Zou. 2023b. How is chatgpt's behavior changing over time? *arXiv preprint arXiv:2307.09009*.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, et al. 2021. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*.

Yukang Chen, Shengju Qian, Haotian Tang, Xin Lai, Zhijian Liu, Song Han, and Jiaya Jia. 2023c. Longlora: Efficient fine-tuning of long-context large language models. *arXiv preprint arXiv:2309.12307*.

Steffi Chern, Ethan Chern, Graham Neubig, and Pengfei Liu. 2024. Can large language models be trusted for evaluation? scalable meta-evaluation of llms as evaluators via agent debate. *arXiv preprint arXiv:2401.16788*.

Cheng-Han Chiang and Hung-yi Lee. 2023. Can large language models be an alternative to human evaluations? *arXiv preprint arXiv:2305.01937*.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. 2022. Scaling instruction-finetuned language models. *arXiv preprint arXiv:2210.11416*.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv preprint arXiv:1803.05457*.

Wenliang Dai, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale N Fung, and Steven Hoi. 2024. Instructblip: Towards general-purpose vision-language models with instruction tuning. *Advances in Neural Information Processing Systems*, 36.

Bosheng Ding, Chengwei Qin, Linlin Liu, Lidong Bing, Shafiq Joty, and Boyang Li. 2022. Is gpt-3 a good data annotator? *arXiv preprint arXiv:2212.10450*.

Markus Freitag and Yaser Al-Onaizan. 2017. Beam search strategies for neural machine translation. *arXiv preprint arXiv:1702.01806*.

Chaoyou Fu, Peixian Chen, Yunhang Shen, Yulei Qin, Mengdan Zhang, Xu Lin, Jinrui Yang, Xiawu Zheng, Ke Li, Xing Sun, Yunsheng Wu, and Rongrong Ji. 2023a. Mme: A comprehensive evaluation benchmark for multimodal large language models. *arXiv preprint arXiv:2306.13394*.

Xingyu Fu, Yushi Hu, Bangzheng Li, Yu Feng, Haoyu Wang, Xudong Lin, Dan Roth, Noah A Smith, Wei-Chiu Ma, and Ranjay Krishna. 2024a. Blink: Multimodal large language models can see but not perceive. *arXiv preprint arXiv:2404.12390*.

Xue-Yong Fu, Md Tahmid Rahman Laskar, Cheng Chen, and Shashi Bhushan Tn. 2023b. Are large language models reliable judges? a study on the factuality evaluation capabilities of LLMs. In *Proceedings of the Third Workshop on Natural Language Generation, Evaluation, and Metrics (GEM)*, pages 310–316, Singapore. Association for Computational Linguistics.

Xue-Yong Fu, Md Tahmid Rahman Laskar, Elena Khasanova, Cheng Chen, and Shashi Bhushan TN. 2024b. Tiny titans: Can smaller large language models punch above their weight in the real world for meeting summarization? *arXiv preprint arXiv:2402.00841*.

Aryo Pradipta Gema, Joshua Ong Jun Leang, Giwon Hong, Alessio Devoto, Alberto Carlo Maria Mancino, Rohit Saxena, Xuanli He, Yu Zhao, Xiaotang Du, Mohammad Reza Ghasemi Madani, et al. 2024. Are we done with mmlu? *arXiv preprint arXiv:2406.04127*.

Bogdan Gliwa, Iwona Mochol, Maciej Biesek, and Aleksander Wawer. 2019. Samsum corpus: A human-annotated dialogue dataset for abstractive summarization. In *Proceedings of the 2nd Workshop on New Frontiers in Summarization*, pages 70–79.

Fabian Gloeckle, Badr Youbi Idrissi, Baptiste Rozière, David Lopez-Paz, and Gabriel Synnaeve. 2024. Better & faster large language models via multi-token prediction. *arXiv preprint arXiv:2404.19737*.

Shahriar Golchin and Mihai Surdeanu. 2023. Time travel in llms: Tracing data contamination in large language models. *arXiv preprint arXiv:2308.08493*.

Tianrui Guan, Fuxiao Liu, Xiyang Wu, Ruiqi Xian, Zongxia Li, Xiaoyu Liu, Xijun Wang, Lichang Chen, Furong Huang, Yaser Yacoob, et al. 2023. Hallusionbench: An advanced diagnostic suite for entangled language hallucination & visual illusion in large vision-language models. *arXiv preprint arXiv:2310.14566*.

Yue Guo, Zian Xu, and Yi Yang. 2023a. Is chatgpt a financial expert? evaluating language models on financial natural language processing. *arXiv preprint arXiv:2310.12664*.

Zishan Guo, Renren Jin, Chuang Liu, Yufei Huang, Dan Shi, Linhao Yu, Yan Liu, Jiaxuan Li, Bojian Xiong, Deyi Xiong, et al. 2023b. Evaluating large language models: A comprehensive survey. *arXiv preprint arXiv:2310.19736*.

Rishav Hada, Varun Gumma, Adrian de Wynter, Harshita Diddee, Mohamed Ahmed, Monojit Choudhury, Kalika Bali, and Sunayana Sitaram. 2023. Are large language model-based evaluators the solution to scaling up multilingual evaluation? *arXiv preprint arXiv:2309.07462*.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2020. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*.

Yutong Hu, Quzhe Huang, Mingxu Tao, Chen Zhang, and Yansong Feng. 2024. Can perplexity reflect large language model's ability in long text understanding? *arXiv preprint arXiv:2405.06105*.

Hui Huang, Yingqi Qu, Jing Liu, Muyun Yang, and Tiejun Zhao. 2024a. An empirical study of llm-as-a-judge for llm evaluation: Fine-tuned judge models are task-specific classifiers. *arXiv preprint arXiv:2403.02839*.

Wei Huang, Xudong Ma, Haotong Qin, Xingyu Zheng, Chengtao Lv, Hong Chen, Jie Luo, Xiaojuan Qi, Xianglong Liu, and Michele Magno. 2024b. How good are low-bit quantized llama3 models? an empirical study. *arXiv preprint arXiv:2404.14047*.

Israt Jahan, Md Tahmid Rahman Laskar, Chun Peng, and Jimmy Huang. 2023. Evaluation of ChatGPT on biomedical tasks: A zero-shot comparison with fine-tuned generative transformers. In *The 22nd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, pages 326–336, Toronto, Canada. Association for Computational Linguistics.

Israt Jahan, Md Tahmid Rahman Laskar, Chun Peng, and Jimmy Xiangji Huang. 2024. A comprehensive evaluation of large language models on benchmark biomedical text processing tasks. *Computers in Biology and Medicine*, page 108189.

Fred Jelinek, Robert L Mercer, Lalit R Bahl, and James K Baker. 1977. Perplexity—a measure of the difficulty of speech recognition tasks. *The Journal of the Acoustical Society of America*, 62(S1):S63–S63.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

Mandar Joshi, Eunsol Choi, Daniel S Weld, and Luke Zettlemoyer. 2017. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1601–1611.

Aniruddha Kembhavi, Mike Salvato, Eric Kolve, Minjoon Seo, Hannaneh Hajishirzi, and Ali Farhadi. 2016. A diagram is worth a dozen images. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part IV 14*, pages 235–251. Springer.

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

Masamune Kobayashi, Masato Mita, and Mamoru Komachi. 2024. Large language models are state-of-the-art evaluator for grammatical error correction. *arXiv preprint arXiv:2403.17540*.

Tom Kocmi and Christian Federmann. 2023. Large language models are state-of-the-art evaluators of translation quality. *arXiv preprint arXiv:2302.14520*.

Jan Kocoń, Igor Cichecki, Oliwier Kaszyca, Mateusz Kochanek, Dominika Szydło, Joanna Baran, Julita Bielaniewicz, Marcin Gruza, Arkadiusz Janz, Kamil Kanclerz, et al. 2023. Chatgpt: Jack of all trades, master of none. *Information Fusion*, 99:101861.

Thomas Kosch and Sebastian Feger. 2024. Risk or chance? large language models and reproducibility in human-computer interaction research. *arXiv preprint arXiv:2404.15782*.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, et al. 2019. Natural questions: a benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:453–466.

Guokun Lai, Qizhe Xie, Hanxiao Liu, Yiming Yang, and Eduard Hovy. 2017. Race: Large-scale reading comprehension dataset from examinations. *arXiv preprint arXiv:1704.04683*.

Nathan Lambert, Valentina Pyatkin, Jacob Morrison, LJ Miranda, Bill Yuchen Lin, Khyathi Chandu, Nouha Dziri, Sachin Kumar, Tom Zick, Yejin Choi, et al. 2024. Rewardbench: Evaluating reward models for language modeling. *arXiv preprint arXiv:2403.13787*.

Md Tahmid Rahman Laskar, M Saiful Bari, Mizanur Rahman, Md Amran Hossen Bhuiyan, Shafiq Joty, and Jimmy Huang. 2023a. A systematic study and comprehensive evaluation of ChatGPT on benchmark datasets. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 431–469, Toronto, Canada. Association for Computational Linguistics.

Md Tahmid Rahman Laskar, Xue-Yong Fu, Cheng Chen, and Shashi Bhushan Tn. 2023b. Building real-world meeting summarization systems using large language models: A practical perspective. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pages 343–352.

Md Tahmid Rahman Laskar, Enamul Hoque, and Jimmy Xiangji Huang. 2022. Domain adaptation with pre-trained transformers for query-focused abstractive text summarization. *Computational Linguistics*, 48(2):279–320.

Chris van der Lee, Albert Gatt, Emiel van Miltenburg, and Emiel Krahmer. 2021. Human evaluation of automatically generated text: Current trends and best practice guidelines. *Computer Speech & Language*, 67:101151.

Bohao Li, Yuying Ge, Yixiao Ge, Guangzhi Wang, Rui Wang, Ruimao Zhang, and Ying Shan. 2023a. Seed-bench-2: Benchmarking multimodal large language models. *arXiv preprint arXiv:2311.17092*.

Bohao Li, Rui Wang, Guangzhi Wang, Yuying Ge, Yixiao Ge, and Ying Shan. 2023b. Seed-bench: Benchmarking multimodal llms with generative comprehension. *arXiv preprint arXiv:2307.16125*.

Changmao Li and Jeffrey Flanigan. 2023. Task contamination: Language models may not be few-shot anymore. *arXiv preprint arXiv:2312.16337*.

Raymond Li, Loubna Ben Allal, Yangtian Zi, Niklas Muennighoff, Denis Kocetkov, Chenghao Mou, Marc Marone, Christopher Akiki, Jia Li, Jenny Chim, et al. 2023c. Starcoder: may the source be with you! *arXiv preprint arXiv:2305.06161*.

Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Wayne Xin Zhao, and Ji-Rong Wen. 2023d. Evaluating object hallucination in large vision-language models. *arXiv preprint arXiv:2305.10355*.

Yinheng Li, Shaofei Wang, Han Ding, and Hang Chen. 2023e. Large language models in finance: A survey. In *Proceedings of the Fourth ACM International Conference on AI in Finance*, pages 374–382.

Percy Liang, Rishi Bommasani, Tony Lee, Dimitris Tsipras, Dilara Soylu, Michihiro Yasunaga, Yian Zhang, Deepak Narayanan, Yuhuai Wu, Ananya Kumar, et al. 2022. Holistic evaluation of language models. *arXiv preprint arXiv:2211.09110*.

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

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

Stephanie Lin, Jacob Hilton, and Owain Evans. 2021. Truthfulqa: Measuring how models mimic human falsehoods. *arXiv preprint arXiv:2109.07958*.

Fuxiao Liu, Kevin Lin, Linjie Li, Jianfeng Wang, Yaser Yacoob, and Lijuan Wang. 2023a. Mitigating hallucination in large multi-modal models via robust instruction tuning. In *The Twelfth International Conference on Learning Representations*.

Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. 2023b. Improved baselines with visual instruction tuning. *arXiv preprint arXiv:2310.03744*.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2024a. Visual instruction tuning. *Advances in neural information processing systems*, 36.

Jiawei Liu, Chunqiu Steven Xia, Yuyao Wang, and Lingming Zhang. 2024b. Is your code generated by chatgpt really correct? rigorous evaluation of large language models for code generation. *Advances in Neural Information Processing Systems*, 36.

Jiawei Liu, Chunqiu Steven Xia, Yuyao Wang, and Lingming Zhang. 2024c. Is your code generated by chatgpt really correct? rigorous evaluation of large language models for code generation. *Advances in Neural Information Processing Systems*, 36.

Nelson F Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. 2024d. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173.

Yi Liu, Lianzhe Huang, Shicheng Li, Sishuo Chen, Hao Zhou, Fandong Meng, Jie Zhou, and Xu Sun. 2023c. Recall: A benchmark for llms robustness against external counterfactual knowledge. *arXiv preprint arXiv:2311.08147*.

Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, et al. 2023d. Mmbench: Is your multi-modal model an all-around player? *arXiv preprint arXiv:2307.06281*.

Pan Lu, Hritik Bansal, Tony Xia, Jiacheng Liu, Chunyuan Li, Hannaneh Hajishirzi, Hao Cheng, Kai-Wei Chang, Michel Galley, and Jianfeng Gao. 2023. Mathvista: Evaluating mathematical reasoning of foundation models in visual contexts. *arXiv preprint arXiv:2310.02255*.

Pan Lu, Swaroop Mishra, Tanglin Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. 2022. Learn to explain: Multimodal reasoning via thought chains for science question answering. *Advances in Neural Information Processing Systems*, 35:2507–2521.

Yujie Lu, Xianjun Yang, Xiujun Li, Xin Eric Wang, and William Yang Wang. 2024. Llmscore: Unveiling the power of large language models in text-to-image synthesis evaluation. *Advances in Neural Information Processing Systems*, 36.

Gen Luo, Yiyi Zhou, Tianhe Ren, Shengxin Chen, Xiaoshuai Sun, and Rongrong Ji. 2024. Cheap and quick: Efficient vision-language instruction tuning for large language models. *Advances in Neural Information Processing Systems*, 36.

Yuanjie Lyu, Zhiyu Li, Simin Niu, Feiyu Xiong, Bo Tang, Wenjin Wang, Hao Wu, Huanyong Liu, Tong Xu, and Enhong Chen. 2024. Crud-rag: A comprehensive chinese benchmark for retrieval-augmented generation of large language models. *arXiv preprint arXiv:2401.17043*.

Rui Mao, Guanyi Chen, Xulang Zhang, Frank Guerin, and Erik Cambria. 2023. Gpteval: A survey on assessments of chatgpt and gpt-4. *arXiv preprint arXiv:2308.12488*.

Ahmed Masry, Do Xuan Long, Jia Qing Tan, Shafiq Joty, and Enamul Hoque. 2022. Chartqa: A benchmark for question answering about charts with visual and logical reasoning. *arXiv preprint arXiv:2203.10244*.

Minesh Mathew, Dimosthenis Karatzas, and CV Jawahar. 2021. Docvqa: A dataset for vqa on document images. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 2200–2209.

Timothy R McIntosh, Teo Susnjak, Tong Liu, Paul Watters, and Malka N Halgamuge. 2024. Inadequacies of large language model benchmarks in the era of generative artificial intelligence. *arXiv preprint arXiv:2402.09880*.

Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. 2018. Can a suit of armor conduct electricity? a new dataset for open book question answering. *arXiv preprint arXiv:1809.02789*.

Shervin Minaee, Tomas Mikolov, Narjes Nikzad, Meysam Chenaghlu, Richard Socher, Xavier Amatriain, and Jianfeng Gao. 2024. Large language models: A survey.

Jinjie Ni, Fuzhao Xue, Xiang Yue, Yuntian Deng, Mahir Shah, Kabir Jain, Graham Neubig, and Yang You. 2024. Mixeval: Deriving wisdom of the crowd from llm benchmark mixtures.

OpenAI. 2023. Gpt-4 technical report.

Yonatan Oren, Nicole Meister, Niladri Chatterji, Faisal Ladhak, and Tatsunori B Hashimoto. 2023. Proving test set contamination in black box language models. *arXiv preprint arXiv:2310.17623*.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744.

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

Xiao Pu, Mingqi Gao, and Xiaojun Wan. 2023. Summarization is (almost) dead. *arXiv preprint arXiv:2309.09558*.

Chengwei Qin, Aston Zhang, Zhuosheng Zhang, Jiaao Chen, Michihiro Yasunaga, and Diyi Yang. 2023. Is chatgpt a general-purpose natural language processing task solver? *arXiv preprint arXiv:2302.06476*.

Haoyi Qiu, Wenbo Hu, Zi-Yi Dou, and Nanyun Peng. 2024. Valor-eval: Holistic coverage and faithfulness evaluation of large vision-language models. *arXiv preprint arXiv:2404.13874*.

Qwen2. 2024. Hello qwen2.

Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. 2018. Improving language understanding by generative pretraining.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for squad. *arXiv preprint arXiv:1806.03822*.

Mathieu Ravaut, Bosheng Ding, Fangkai Jiao, Hailin Chen, Xingxuan Li, Ruochen Zhao, Chengwei Qin, Caiming Xiong, and Shafiq Joty. 2024. How much are llms contaminated? a comprehensive survey and the llmsanitize library. *arXiv preprint arXiv:2404.00699*.

Vipula Rawte, Amit Sheth, and Amitava Das. 2023. A survey of hallucination in large foundation models. *arXiv preprint arXiv:2309.05922*.

Baptiste Roziere, Jonas Gehring, Fabian Gloeckle, Sten Sootla, Itai Gat, Xiaoqing Ellen Tan, Yossi Adi, Jingyu Liu, Tal Remez, Jérémy Rapin, et al. 2023. Code llama: Open foundation models for code. *arXiv preprint arXiv:2308.12950*.

Oscar Sainz, Jon Ander Campos, Iker García-Ferrero, Julen Etxaniz, Oier Lopez de Lacalle, and Eneko Agirre. 2023a. Nlp evaluation in trouble: On the need to measure llm data contamination for each benchmark. *arXiv preprint arXiv:2310.18018*.

Oscar Sainz, Jon Ander Campos, Iker García-Ferrero, Julen Etxaniz, Oier Lopez de Lacalle, and Eneko Agirre. 2023b. Nlp evaluation in trouble: On the need to measure llm data contamination for each benchmark. *arXiv preprint arXiv:2310.18018*.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2021. Winogrande: An adversarial winograd schema challenge at scale. *Communications of the ACM*, 64(9):99–106.

SambaNova. 2024. Samba-coe v0.3: The power of routing ml models at scale.

Maarten Sap, Hannah Rashkin, Derek Chen, Ronan LeBras, and Yejin Choi. 2019. Socialiqa: Commonsense reasoning about social interactions. *arXiv preprint arXiv:1904.09728*.

Sander Schulhoff, Michael Ilie, Nishant Balepur, Konstantine Kahadze, Amanda Liu, Chenglei Si, Yinheng Li, Aayush Gupta, HyoJung Han, Sevien Schulhoff, Pranav Sandeep Dulepet, Saurav Vidyadhara, Dayeon Ki, Sweta Agrawal, Chau Pham, Gerson C. Kroiz, Feileen Li, Hudson Tao, Ashay Srivastava, Hevander Da Costa, Saloni Gupta, Megan L. Rogers, Inna Goncearenco, Giuseppe Sarli, Igor Galynker, Denis Peskoff, Marine Carpuat, Jules White, Shyamal Anadkat, Alexander Miserlis Hoyle, and Philip Resnik. 2024. The prompt report: A systematic survey of prompting techniques.

Sander Schulhoff, Jeremy Pinto, Anaum Khan, Louis-François Bouchard, Chenglei Si, Svetlina Anati, Valen Tagliabue, Anson Kost, Christopher Carnahan, and Jordan Boyd-Graber. 2023. Ignore this title and hackaprompt: Exposing systemic vulnerabilities of llms through a global prompt hacking competition. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 4945–4977.

Melanie Sclar, Yejin Choi, Yulia Tsvetkov, and Alane Suhr. 2023. Quantifying language models' sensitivity to spurious features in prompt design or: How i learned to start worrying about prompt formatting. *arXiv preprint arXiv:2310.11324*.

Chenhui Shen, Liying Cheng, Xuan-Phi Nguyen, Yang You, and Lidong Bing. 2023. Large language models are not yet human-level evaluators for abstractive summarization. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 4215–4233.

Chufan Shi, Haoran Yang, Deng Cai, Zhisong Zhang, Yifan Wang, Yujiu Yang, and Wai Lam. 2024. A thorough examination of decoding methods in the era of llms. *arXiv preprint arXiv:2402.06925*.

Rickard Stureborg, Dimitris Alikaniotis, and Yoshi Suhara. 2024. Large language models are inconsistent and biased evaluators. *arXiv preprint arXiv:2405.01724*.

Alon Talmor and Jonathan Berant. 2018. The web as a knowledge-base for answering complex questions. *arXiv preprint arXiv:1803.06643*.

Yixuan Tang and Yi Yang. 2024. Multihop-rag: Benchmarking retrieval-augmented generation for multi-hop queries. *arXiv preprint arXiv:2401.15391*.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, et al. 2023. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*.

Gemma Team, Thomas Mesnard, Cassidy Hardin, Robert Dadashi, Surya Bhupatiraju, Shreya Pathak, Laurent Sifre, Morgane Rivière, Mihir Sanjay Kale, Juliette Love, et al. 2024. Gemma: Open models based on gemini research and technology. *arXiv preprint arXiv:2403.08295*.

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

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023b. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Pat Verga, Sebastian Hofstatter, Sophia Althammer, Yixuan Su, Aleksandra Piktus, Arkady Arkhangorodsky, Minjie Xu, Naomi White, and Patrick Lewis. 2024. Replacing judges with juries: Evaluating llm generations with a panel of diverse models. *arXiv preprint arXiv:2404.18796*.

Jindong Wang, Xixu Hu, Wenxin Hou, Hao Chen, Runkai Zheng, Yidong Wang, Linyi Yang, Haojun Huang, Wei Ye, Xiubo Geng, et al. 2023a. On the robustness of chatgpt: An adversarial and out-of-distribution perspective. *arXiv preprint arXiv:2302.12095*.

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

Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, Tianle Li, Max Ku, Kai Wang, Alex Zhuang, Rongqi Fan, Xiang Yue, and Wenhu Chen. 2024. Mmlu-pro: A more robust and challenging multi-task language understanding benchmark.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Ed Chi, Quoc Le, and Denny Zhou. 2022. Chain of thought prompting elicits reasoning in large language models. *arXiv preprint arXiv:2201.11903*.

Man-Fai Wong, Shangxin Guo, Ching-Nam Hang, Siu-Wai Ho, and Chee-Wei Tan. 2023. Natural language generation and understanding of big code for ai-assisted programming: A review. *Entropy*, 25(6):888.

Minghao Wu and Alham Fikri Aji. 2023. Style over substance: Evaluation biases for large language models. *arXiv preprint arXiv:2307.03025*.

Congying Xia, Chen Xing, Jiangshu Du, Xinyi Yang, Yihao Feng, Ran Xu, Wenpeng Yin, and Caiming Xiong. 2024. Fofo: A benchmark to evaluate llms' format-following capability. *arXiv preprint arXiv:2402.18667*.

Guangzhi Xiong, Qiao Jin, Zhiyong Lu, and Aidong Zhang. 2024. Benchmarking retrieval-augmented generation for medicine. *arXiv preprint arXiv:2402.13178*.

Ruijie Xu, Zengzhi Wang, Run-Ze Fan, and Pengfei Liu. 2024. Benchmarking benchmark leakage in large language models. *arXiv preprint arXiv:2404.18824*.

Linyi Yang, Shuibai Zhang, Libo Qin, Yafu Li, Yidong Wang, Hanmeng Liu, Jindong Wang, Xing Xie, and Yue Zhang. 2022. Glue-x: Evaluating natural language understanding models from an out-of-distribution generalization perspective. *arXiv preprint arXiv:2211.08073*.

Junjie Ye, Xuanting Chen, Nuo Xu, Can Zu, Zekai Shao, Shichun Liu, Yuhan Cui, Zeyang Zhou, Chao Gong, Yang Shen, et al. 2023a. A comprehensive capability analysis of gpt-3 and gpt-3.5 series models. *arXiv preprint arXiv:2303.10420*.

Qinghao Ye, Haiyang Xu, Guohai Xu, Jiabo Ye, Ming Yan, Yiyang Zhou, Junyang Wang, Anwen Hu, Pengcheng Shi, Yaya Shi, et al. 2023b. mplug-owl: Modularization empowers large language models with multimodality. *arXiv preprint arXiv:2304.14178*.

Weihao Yu, Zhengyuan Yang, Linjie Li, Jianfeng Wang, Kevin Lin, Zicheng Liu, Xinchao Wang, and Lijuan Wang. 2023. Mm-vet: Evaluating large multimodal models for integrated capabilities. *arXiv preprint arXiv:2308.02490*.

Xiang Yue, Yuansheng Ni, Kai Zhang, Tianyu Zheng, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, et al. 2023. Mmmu: A massive multi-discipline multimodal understanding and reasoning benchmark for expert agi. *arXiv preprint arXiv:2311.16502*.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. Hellaswag: Can a machine really finish your sentence? *arXiv preprint arXiv:1905.07830*.

Hugh Zhang, Jeff Da, Dean Lee, Vaughn Robinson, Catherine Wu, Will Song, Tiffany Zhao, Pranav Raja, Dylan Slack, Qin Lyu, et al. 2024a. A careful examination of large language model performance on grade school arithmetic. *arXiv preprint arXiv:2405.00332*.

Pan Zhang, Xiaoyi Dong Bin Wang, Yuhang Cao, Chao Xu, Linke Ouyang, Zhiyuan Zhao, Shuangrui Ding, Songyang Zhang, Haodong Duan, Hang Yan, et al. 2023. Internlm-xcomposer: A vision-language large model for advanced text-image comprehension and composition. *arXiv preprint arXiv:2309.15112*.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.

Tianyi Zhang, Faisal Ladhak, Esin Durmus, Percy Liang, Kathleen McKeown, and Tatsunori B Hashimoto. 2024b. Benchmarking large language models for news summarization. *Transactions of the Association for Computational Linguistics*, 12:39–57.

Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, et al. 2023a. A survey of large language models. *arXiv preprint arXiv:2303.18223*.

Yao Zhao, Rishabh Joshi, Tianqi Liu, Misha Khalman, Mohammad Saleh, and Peter J Liu. 2023b. Slic-hf: Sequence likelihood calibration with human feedback. *arXiv preprint arXiv:2305.10425*.

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

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Haotong Zhang, Joseph Gonzalez, and Ion Stoica. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. *ArXiv*, abs/2306.05685.

Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. 2023a. Minigpt-4: Enhancing vision-language understanding with advanced large language models. *arXiv preprint arXiv:2304.10592*.

Kaijie Zhu, Jindong Wang, Qinlin Zhao, Ruochen Xu, and Xing Xie. 2024. Dyval 2: Dynamic evaluation of large language models by meta probing agents.

Kaijie Zhu, Jindong Wang, Jiaheng Zhou, Zichen Wang, Hao Chen, Yidong Wang, Linyi Yang, Wei Ye, Neil Zhenqiang Gong, Yue Zhang, et al. 2023b. Promptbench: Towards evaluating the robustness of large language models on adversarial prompts. *arXiv preprint arXiv:2306.04528*.

Yutao Zhu, Huaying Yuan, Shuting Wang, Jiongnan Liu, Wenhan Liu, Chenlong Deng, Zhicheng Dou, and Ji-Rong Wen. 2023c. Large language models for information retrieval: A survey. *arXiv preprint arXiv:2308.07107*.

Ziyu Zhuang, Qiguang Chen, Longxuan Ma, Mingda Li, Yi Han, Yushan Qian, Haopeng Bai, Zixian Feng, Weinan Zhang, and Ting Liu. 2023. Through the lens of core competency: Survey on evaluation of large language models. *arXiv preprint arXiv:2308.07902*.

Terry Yue Zhuo, Yujin Huang, Chunyang Chen, and Zhenchang Xing. 2023. Red teaming chatgpt via jailbreaking: Bias, robustness, reliability and toxicity. *arXiv preprint arXiv:2301.12867*.

Caleb Ziems, William Held, Omar Shaikh, Jiaao Chen, Zhehao Zhang, and Diyi Yang. 2024. Can large language models transform computational social science? *Computational Linguistics*, pages 1–55.

A Appendix

A.1 Benchmarking Datasets

General Capability Benchmarks: To benchmark the performance of LLMs, researchers typically use a set of widely recognized datasets. These common benchmarks are employed by authors upon the release of an LLM to evaluate its general capabilities. One of the most frequently used benchmarks is the MMLU benchmark (Hendrycks et al., 2020), which assesses LLMs' overall knowledge and reasoning abilities across various subjects. Other common benchmarks focus primarily on evaluating the commonsense reasoning capabilities of LLMs, such as HellaSwag (Zellers et al., 2019), PIQA (Bisk et al., 2020), SIQA, (Sap et al., 2019), WinoGrande (Sakaguchi et al., 2021), OpenBookQA (Mihaylov et al., 2018), ARC (Clark et al., 2018). In addition, the TruthfulQA dataset (Lin et al., 2021) is used to measure the truthfulness of an LLM. For assessing coding capabilities, the HumanEval (Chen et al., 2021) and the MBPP (Austin et al., 2021) are two widely used benchmarks.

Specialized Benchmarks: There are also specialized benchmarks that measure specific capabilities of LLMs. For instance, the MT-Bench (Zheng et al., 2024)) evaluates whether LLMs can properly engage in conversations, while the RewardBench (Lambert et al., 2024) assesses the performance of reward models. The FOFO benchmark Xia et al. (2024) measures language models' ability to adhere to the requested formats in prompts across different domains. The ability to understand both language and vision is another interesting capability of recently proposed LLMs (Bai et al., 2023; Chen et al., 2023a; Dai et al., 2024; Liu et al., 2023b, 2024a; Luo et al., 2024; Ye et al., 2023b; Zhang et al., 2023; Zhu et al., 2023a). This has led to the development of many multi-modal benchmarks (Chen et al., 2024b; Fu et al., 2023a, 2024a; Guan et al., 2023; Li et al., 2023a,b,d; Liu et al., 2024a, 2023d; Lu et al., 2022; Qiu et al., 2024; Yu et al., 2023). These benchmarks study the multimodal capabilities of LLMs across various domains, such as math and reasoning (Lu et al., 2023; Yue et al., 2023), science diagrams (Kembhavi et al., 2016), chart understanding and reasoning (Masry et al., 2022), document understanding (Mathew et al., 2021).

Other Diverse Benchmarks: To enable a more comprehensive evaluation of LLMs across a wide range of scenarios, some studies also focused on introducing new benchmarks covering various aspects, such as HELM (Liang et al., 2022), PromptBench (Zhu et al., 2023b), OpenLLM.[7] These benchmarks cover diverse tasks and usually include existing benchmarking datasets (e.g., MMLU, HellaSwag, etc.). Additionally, despite the availability of numerous benchmarks (both general and specialized), existing widely-used benchmarks still do not cover the full variety of tasks. Therefore, some researchers have independently evaluated LLMs using additional diverse datasets and tasks, including various NLP datasets and tasks (Bang et al., 2023; Kocoń et al., 2023; Laskar et al., 2023a; Qin et al., 2023). They also employed domain-specific benchmarks in fields such as biomedicine (Jahan et al., 2023, 2024), finance (Guo et al., 2023a; Li et al., 2023e), social science (Ziems et al., 2024), coding (Liu et al., 2024c), and information retrieval (Zhu et al., 2023c). In addition to that, ethics, bias, toxicity, robustness, and trustworthiness are also independently evaluated by researchers across various datasets (Liu et al., 2023a; McIntosh et al., 2024; Rawte et al., 2023; Wang et al., 2023a; Yang et al., 2022; Zhuo et al., 2023).

A.2 Prominent LLMs

The impressive success of ChatGPT has led to the development of many LLMs in recent years. Since there are hundreds of LLMs being released in recent years (Zhao et al., 2023a), we only discuss some of the prominent LLMs that achieved top rankings in various public leaderboards recently. LLMs can be categorized into two parts: *Closed-Source LLMs*: only available for use through the API or web interface, and (ii) *Open-Source LLMs*: where the pre-trained weights of the model are available that allow further training of such models. Below, we present some prominent LLMs in these two categories.

A.2.1 Closed Source LLMs

In the following, we categorize LLMs based on the organizations that develop these LLMs:

OpenAI models:

- **GPT-3.5:** This model is an iteration of the GPT-3 architecture, emphasizing improvements in response quality through the application of the reinforcement learning from human feedback (RLHF) technique. GPT-3.5 is known for its robust performance in zero-shot tasks, where no specific training examples are provided during the task execution. This model has been instrumental due to its strong foundational capabilities in understanding and generating human-like text (OpenAI, 2023).

- **GPT-4:** It extends GPT-3.5's capabilities by incorporating multimodal functionalities, allowing the model to process not just text but also visual inputs. This advancement significantly broadens its applicational scope, making it adept at handling more complex tasks that require an understanding of both textual and visual information. It features enhanced safety protocols and a sophisticated training regime that includes a safety reward signal during its reinforcement learning phase (OpenAI, 2023).

- **GPT-4V:** It focuses on optimizing the vision capabilities of GPT-4. It specifically addresses and mitigates risks associated with processing visually augmented inputs, making it a safer and more effective model for tasks involving images.

- **GPT-4 Turbo:** This version builds upon GPT-4's foundation with substantial upgrades in computational efficiency and functionality. GPT-4 Turbo boasts an increased

---

[7] `https://huggingface.co/spaces/HuggingFaceH4/open_llm_leaderboard`, MixEval (Ni et al., 2024)

model capacity and an extended knowledge base that encompasses more recent data up to April 2023. It features a longer context window of up to 128,000 tokens and includes significant improvements in the model's economy and output consistency.

- **GPT-4o:** OpenAI's most sophisticated model, GPT-4o ("o" for "omni") is a multimodal powerhouse capable of handling both text and image inputs to generate text outputs. It improves upon GPT-4 Turbo by offering double the text generation speed and reducing operational costs by 50%.

Google models:

- **PaLM-2:** Released by Google in 2023, it is an advanced large language model that builds on the foundations set by its predecessor, the original PaLM. This iteration incorporates a sophisticated 'mixture of objectives' technique, allowing it to surpass the capabilities of the earlier model significantly (Anil et al., 2023).
- **Gemini:** It is a multimodal model developed by google in December 2023, to understand and process a variety of information types, including text, images, audio, and video, seamlessly. Gemini's architecture allows it to perform exceptionally across multiple platforms, from large-scale data centers to mobile devices, adapting efficiently to the needs of different applications. This model sets new benchmarks in AI with its ability to excel in tasks that require complex multimodal integrations (Team et al., 2023).

Anthropic Models: ***The Claude series*** models, developed by Anthropic, represent a series of advanced language models designed to enhance user interaction through natural language understanding and generation. Starting with the original Claude, which excelled in tasks like summarization and creative writing, each subsequent model—Claude Instant, Claude 2.0, and the Claude 3 family (Haiku, Sonnet, and Opus)—has introduced significant improvements in processing speed, reasoning capabilities, and multimodal functionality. These models have a variety of uses, from quick response generation in Claude Instant to sophisticated multimodal understanding in Claude 3 Opus, showcasing their versatility and advanced AI technology to meet different user and enterprise needs3[8].

A.2.2 Open Source LLMs

We similarly categorize the open-source LLMs based on the organizations that develop these LLMs:

Meta Models:

- **Llama:** Launched in February 2023 by Meta AI, Llama was the first in the Llama series, showcasing strong performance on a range of natural language processing tasks. It competed well against larger models like GPT-3 with a smaller parameter size and was made available under a non-commercial license, primarily for academic research (Touvron et al., 2023a).
- **Llama 2:** Released in July 2023, Llama 2 improved on its predecessor by expanding model sizes up to 70 billion parameters. It maintained the original architecture but included better training data and enhanced functionality. Notably, Llama 2 was more accessible, available for both academic and some commercial uses (Touvron et al., 2023b).
- **Llama 3:** In April 2024, Meta AI introduced Llama 3, the most advanced version with up to 70 billion parameters. This version added longer context capabilities and improved multimodal functions, marking a significant advancement in AI technology application across various fields.

Mistral Models: **Mistral AI**, founded in April 2023, is a Paris-based company specializing in the development of open-source large language models. Rapidly gaining recognition in the AI industry, Mistral AI emphasizes the importance of open-source software, providing a viable alternative to proprietary models. The company has released several models, including Mistral 7B, Mixtral 8x7B, and Mixtral 8x22B, which are known for their high performance and innovation in the use of mixture of experts architectures. Codestral 22B, introduced on May 29, 2024, is a pioneering code generation model designed to enhance coding efficiency across more than 80 pro-

---

[8] `https://www.anthropic.com/news/claude-3-family`

gramming languages. With its specialized focus and lightweight architecture, Codestral significantly outperforms other leading models on the HumanEval FIM benchmark, making it a critical tool for developers seeking advanced AI-assisted coding capabilities(Jiang et al., 2023).

Alibaba Models: QWEN series models are transformer-based large language models developed by Alibaba Cloud. These models, pretrained on diverse data sources including web texts, books, code, and more, come in various sizes ranging from 0.5 billion to 110 billion parameters. Qwen models support long context lengths and demonstrate strong performance on multiple Chinese and English evaluation tasks, including common-sense reasoning, code, and mathematics. The latest versions, Qwen 1.5 and Qwen 2, offer significant improvements in chat model performance, multilingual support, and stable support for up to 32K context length. With a comprehensive vocabulary of over 150K tokens, Qwen models are designed to handle multiple languages effectively, making them a versatile tool for various AI applications (Bai et al., 2023).

Microsoft Models: The **Phi series** by Microsoft consists of small language models (SLMs) designed to provide high performance with lower computational requirements. The newly announced Phi-3 family includes models like Phi-3-mini, Phi-3-small, and Phi-3-medium, ranging from 3.8 billion to 14 billion parameters. These models excel in various benchmarks, offering capabilities similar to larger models but in a smaller, more cost-effective package. Phi-3 models are particularly suited for simpler tasks, local device operations, and environments with limited resources, making AI more accessible and efficient for diverse applications. They are available through Microsoft Azure AI Model Catalog, Hugging Face, and as NVIDIA NIM microservices(Abdin et al., 2024).

Technology Innovation Institute Models: Falcon series models (Almazrouei et al., 2023), such as the Falcon 2 series include models with parameter sizes such as 1.3B, 7.5B, 40B, and 180B. These models are notable for their use of the REFINEDWEB dataset. Falcon models are designed for both research and commercial use, with Falcon 2 models featuring multilingual and multimodal capabilities, including vision-to-language. The Falcon 180B model, in particular, is accessible under a royalty-free license.

Cohere Models: Cohere offers a variety of advanced large language models designed for multiple use cases, including text generation, embeddings, and reranking. The Command family models, such as Command R+ and Command R, excel in conversational tasks and complex workflows like code generation and retrieval-augmented generation (RAG) [9] (Alonso et al., 2024; Chen et al., 2024a; Liu et al., 2023c; Lyu et al., 2024; Tang and Yang, 2024; Xiong et al., 2024). The Embed models enhance search, classification, and clustering capabilities with both English and multilingual support. The Rerank models improve search algorithms by re-organizing results based on specified parameters. Cohere models are accessible across platforms like Amazon SageMaker, Microsoft Azure, and Oracle GenAI Service, enabling seamless integration into diverse applications.

A.3 Prompting Techniques

Prompts can be designed in various ways (Brown et al., 2020; Chung et al., 2022; Schulhoff et al., 2024; Wei et al., 2022), as stated below:

- **In-Context Learning (Zero-shot):** It means that the prompt used to interact with the model contains no examples or demonstrations. The model relies on its pre-existing knowledge, obtained from its initial training on diverse data, to generate a response or perform the task based solely on the instructions given. For example, “classify the sentence as biased or unbiased text”.

- **In-Context Learning (Few-shot):** It means that the prompt used to interact with the model includes a small number of examples or demonstrations. The model uses these examples to quickly adapt and understand how to perform a specific task, leveraging the details within these examples. This technique allows the model to extend its pre-existing knowledge to new tasks by closely analyzing the limited examples given. For instance, classify the sentence as biased or unbiased based on a few similar examples provided.

---

[9] `https://cohere.com/command`

- **Chain-of-Thought Prompting (CoT):** This technique encourages models to generate intermediate reasoning steps before arriving at a final answer, mimicking a human-like problem-solving approach. This can be combined with few-shot prompting to achieve better results on more complex tasks. For example, if asked to determine whether the number "15" is odd or even, the model might outline its reasoning as follows: "An even number is divisible by 2 without a remainder. 15 divided by 2 is 7 with a remainder of 1. Therefore, 15 is an odd number." This step-by-step explanation helps clarify the model's thought process and supports its conclusion.

- **Decomposition Techniques:** These techniques break down complex problems into simpler sub-problems that can be solved sequentially by the GenAI model. Each component of the problem is addressed individually, and the solutions are integrated to form a comprehensive response. Decomposition is especially useful in tasks that require layered reasoning or have multiple steps. For example, in solving a math word problem, decomposition might involve separately calculating the distances each person travels and then combining these calculations to determine when they meet.

- **Role-based and Style-based Prompting:** In these techniques prompts are designed to induce a specific style or persona in the model's responses. By specifying a role (e.g., a scientist explaining a concept) or a style (e.g., formal or poetic), users can guide the tone and formality of the AI's output. This technique is valuable in applications requiring genre-specific content generation or when the output needs to fit a particular communicative context.

- **Prompt chaining:** It is a technique where a complex task is divided into simpler subtasks, each addressed by its own prompt. The response from one prompt is used as the input for the next, creating a sequential chain of prompts that gradually build towards the final answer. This method enhances the performance and reliability of large language models by breaking down tasks into manageable parts, making it easier to control and refine the model's responses at each step. For example, in a document analysis task, the first prompt might extract key facts from a text, and the second prompt would use these facts to generate a summary.

- **Tree of Thoughts (ToT):** It is a technique that structures problem-solving into a tree of possible solutions. It uses strategies like like breadth-first or depth-first search to evaluate each potential solution path. For example, in solving a puzzle, ToT might explore different moves systematically to find the quickest solution path.

- **Directional Stimulus Prompting (DSP) :** It is a technique that enhances how large language models (LLMs) respond to tasks by using dynamically generated prompts. A secondary, tuneable model creates specific hints that guide the main, unchangeable LLM to produce more targeted and relevant outputs. This method uses reinforcement learning to refine these prompts based on how well they perform, making DSP a more adaptive and precise approach compared to standard prompting techniques. For instance, in summarizing complex documents, DSP might generate a prompt like "Summarize focusing on economic impacts," guiding the LLM to tailor its output specifically to the economic aspects mentioned in the text.

- **Multimodal Prompting:** Extending beyond text, multimodal prompting involves using inputs like images, audio, or video along with textual descriptions to guide the AI's response. This technique leverages the model's capability to process and integrate information from diverse data types, enhancing its applicability in scenarios where multiple forms of data are available. For example, interpret a scene from a video by analyzing both the spoken dialogue and the visual content to determine the mood of the conversation.

- **Meta-Prompting:** It involves creating prompts that instruct the AI to generate or refine its prompts, essentially using AI to improve the efficiency and effectiveness of prompt engineering. This recursive use of prompting can lead to more dynamic and contextually adaptive AI behaviors. For example, ask the AI to optimize a prompt that instructs another AI to summarize news articles, thereby refining the instructions to enhance summary relevance and conciseness.

A.4 Decoding Parameters

There are various decoding parameters that are required to be set. For instance:

- **Temperature:** It is used to control the randomness of the output. It is typically between 0 and 1. Lower values (e.g., 0.1) make the model more deterministic and focused on the most likely next token, while higher values (e.g., 0.9) introduce more randomness and diversity.
- **Beam Size:** It refers to the number of beams in Beam Search (Freitag and Al-Onaizan, 2017), a decoding strategy that keeps track of multiple possible sequences (beams) at each step of generation to find the most likely sequence. A higher number of beams usually leads to more accurate results but at the cost of increased computation.
- **Top-K:** The number of top probable tokens to consider. For example, if K=10, the model will choose the next token only from the top 10 most likely tokens.
- **Top-P:** The cumulative probability threshold. For example, if P=0.9, the model will sample from the smallest set of tokens whose combined probability is at least 90%.
- **Maximum Output Tokens:** It determines how many tokens the model will generate at maximum.

A.5 Parsing Script Design

We present some scenarios in Table 4 to demonstrate why parsing script is required for such cases and the importance of validating parsing scripts.

A.6 Evaluation Approach

A.6.1 Automatic Evaluation

To provide a high-level overview, automatic evaluation for LLMs can be divided into the following categories:

Language Modeling: Perplexity (Jelinek et al., 1977) is widely used to study the performance of auto-regressive language models. It measures how confidently a model predicts the next word in a sequence, with the assumption that lower perplexity indicates better performance. Hence, perplexity has been historically used to assess the language model's capability to generate a coherent language and is also useful to quickly compare different models or checkpoints.

Discriminative Tasks: For tasks involving class prediction, post-processing using a parsing script is usually required to extract answers from the LLM-generated responses to compare against gold labels. In this context, metrics such as Exact Match, Accuracy, Precision, Recall, F1, are usually utilized in discriminative tasks (Bang et al., 2023; Laskar et al., 2023a; Qin et al., 2023).

Generative Tasks: For generative tasks such as summarization or machine translation, parsing scripts are usually not required (Jahan et al., 2024; Laskar et al., 2023a) and so the full response generated by LLMs are compared against the gold reference. In this regard, *ROUGE* (Lin, 2004b) and *BLEU* (Papineni et al., 2002) which are based on n-gram word matching are widely used. Meanwhile, various contextualized similarity-based metrics (e.g., *BERTScore* (Zhang et al., 2019)) are also utilized that do not depend on word-based similarity measures.

A.6.2 Human Evaluation

Since LLMs generate human-like responses, it is often required to conduct qualitative evaluation of their responses. Earlier, qualitative evaluation of model-generated summaries in terms of fluency, coherence, and informativeness were very popular (Laskar et al., 2022). However, with LLMs usually generating informative, fluent, and coherent response (Bang et al., 2023; Kocoń et al., 2023; Laskar et al., 2023a; Qin et al., 2023), the evaluation of factual consistency of LLM-generated responses has become more important recently (Fu et al., 2023b). Moreover, qualitative evaluation to compare between LLM-generated responses via leveraging humans based on the Elo rating system (Zheng et al., 2024) has gained a lot of attention.

Elo Rating: Elo rating works by comparing LLMs in pairwise "A vs B" comparisons, where each model is assigned an initial numerical rating (Boubdir et al., 2023; Zhao et al., 2023b). The outcome of each comparison adjusts these ratings based on the Elo algorithm: if a model performs

Scenario 1: For the response generated, designing a parsing script to extract the answer "Lionel Messi" is straightforward. However, the parsing script should also be robust to cover cases like abbreviations, uppercase-lowercase sensitivity, punctuations, synonyms, stemming, lemmatization, paraphrases, etc.
Prompt: Which player has won the best player award in Fifa world cup 2022?
Sample LLM Response (GPT 4o): Lionel Messi won the Best Player award (Golden Ball) in the FIFA World Cup 2022. He was instrumental in leading Argentina to victory in the tournament, culminating in their triumph in the final against France.
Correct Answer: Lionel Messi
Scenario 2: While Extraction of the answer "Lionel Messi" is required, due to the LLM knowledge-cut-off date of September 2021, it may answer about 2018. However, the target answer "Lionel Messi" is also in the output and so if the parsing script only parses the target answer then it may consider the response as correct whereas the response is wrong.
Prompt: Which player has won the best player award in the last Fifa world cup?
Sample LLM Response (Older ChatGPT 3.5 having knowledge cut-off date of September 2021): The Best Player award (Golden Ball) in the previous FIFA World Cup, which was held in 2018 in Russia, was won by Luka Modric from Croatia. Prior to the that, Lionel Messi had won it in 2014.
Correct Answer: Lionel Messi

Table 4: Some examples of LLM-generated response requiring parsing script to extract the target answer. For Scenario 2, human evaluation is usually needed to ensure accurate parsing of the answer.

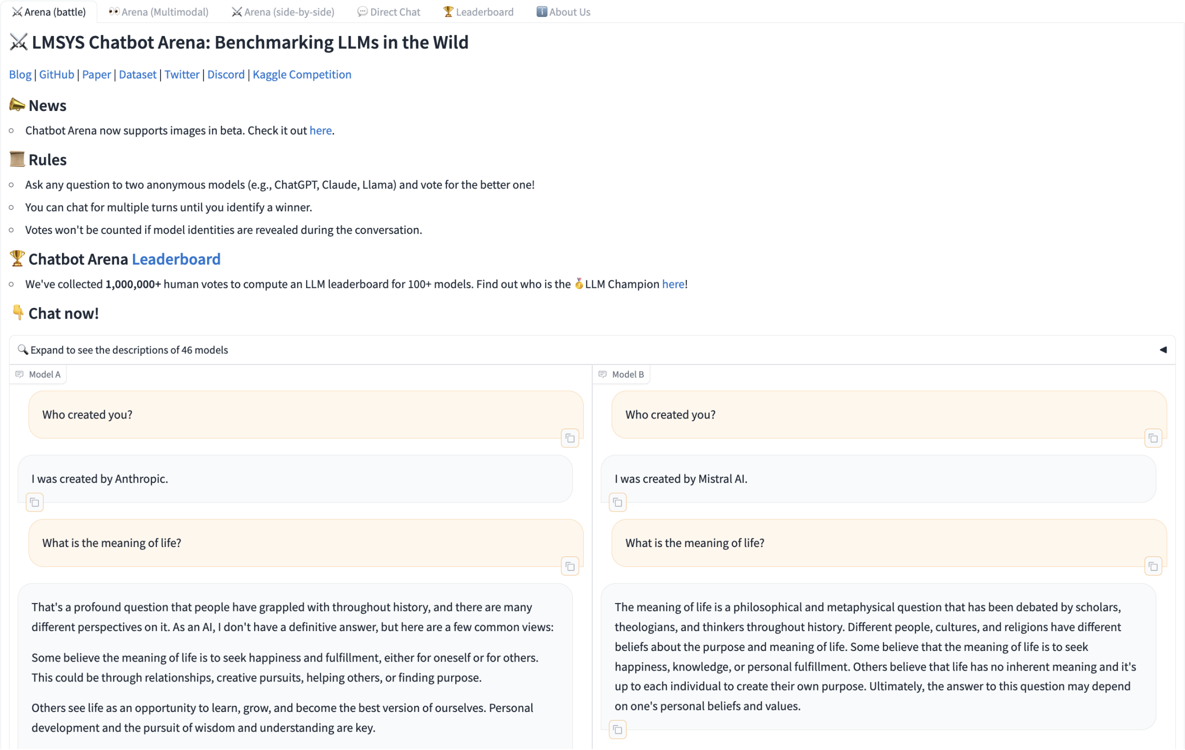
better than expected, its rating increases; if it performs worse, its rating decreases. The expectation of a model's performance is calculated using its rating relative to its opponent's, adjusted by a factor that represents the sensitivity of expected scores to differences in ratings.

To ensure a robust evaluation of LLMs using the Elo benchmark, it's important to follow key indicators like reliability and transitivity (Boubdir et al., 2023). Reliability keeps Elo ratings consistent across various comparison sequences and prevents them from being overly sensitive to changes in hyperparameters, such as the K-factor. Transitivity is crucial, indicating that if model A is rated higher than model B, and model B is rated higher than model C, model A should logically rank above model C. Extensive testing with both synthetic and real-world data is essential to verify that Elo scores accurately and stably reflect model performance (Boubdir et al., 2023). This involves making precise adjustments to the comparison order, selecting hyperparameters carefully, and utilizing numerous permutations to ensure outcome consistency. Due to the sensitive nature of the Elo rating system towards the order in which the updates were performed, Zheng et al. (2024) used the Bradley-Terry (BTL) model for their chatbot arena ranking. It is observed that model A can have a higher win rate than model B both empirically and statistically but a lower Elo rating. Since win rate serves as the stand-in measure for the probability of a model being better than another, this signifies the findings by Boubdir et al. (2023) that Elo rating is non-transitive with or without (BTL). On the other hand, BTL-based rating is tolerant to an imbalanced number of votes per model as shown by (Zheng et al., 2024), they also propose a different probability of win rates that are derived from the ratings found from BTL which then is transitive though it doesn't correlate with the empirical win rates.

Elo hacking: Crowdsourced Elo-based ranking has gained popularity through the LMSys leaderboard [10] and has been accepted by various organizations, prompting them to release their LLMs early into this ecosystem for human evaluation. However, such setups can be easily exploited on a large scale using simple techniques. Figure 5 illustrates how someone can initially bypass the blind scoring mechanism through ownership hacking. Additionally, the evaluation of knowledge bases is not easily tracked, making votes on highly complex reasoning questions equivalent to those on simpler queries. Furthermore, upon the release of a popular model, systematic attacks or boosting can be initiated through ownership hacking. In ad-

---

[10] `https://huggingface.co/spaces/lmsys/chatbot-arena-leaderboard`


⚔️ Arena (battle)
👀 Arena (Multimodal)
⚔️ Arena (side-by-side)
💬 Direct Chat
🏆 Leaderboard
ℹ️ About Us
Blog | GitHub | Paper | Dataset | Twitter | Discord | Kaggle Competition
Expand to see the descriptions of 46 models
Model A
Model B

⚔️ **LMSYS Chatbot Arena: Benchmarking LLMs in the Wild**

News

- Chatbot Arena now supports images in beta. Check it out here.

Rules

- Ask any question to two anonymous models (e.g., ChatGPT, Claude, Llama) and vote for the better one!
- You can chat for multiple turns until you identify a winner.
- Votes won't be counted if model identities are revealed during the conversation.

🏆 **Chatbot Arena Leaderboard**

- We've collected **1,000,000+** human votes to compute an LLM leaderboard for 100+ models. Find out who is the 🥇LLM Champion here!

Chat now!

Model A:

Who created you?

I was created by Anthropic.

What is the meaning of life?

That's a profound question that people have grappled with throughout history, and there are many different perspectives on it. As an AI, I don't have a definitive answer, but here are a few common views:

Some believe the meaning of life is to seek happiness and fulfillment, either for oneself or for others. This could be through relationships, creative pursuits, helping others, or finding purpose.

Others see life as an opportunity to learn, grow, and become the best version of ourselves. Personal development and the pursuit of wisdom and understanding are key.

Model B:

Who created you?

I was created by Mistral AI.

What is the meaning of life?

The meaning of life is a philosophical and metaphysical question that has been debated by scholars, theologians, and thinkers throughout history. Different people, cultures, and religions have different beliefs about the purpose and meaning of life. Some believe that the meaning of life is to seek happiness, knowledge, or personal fulfillment. Others believe that life has no inherent meaning and it's up to each individual to create their own purpose. Ultimately, the answer to this question may depend on one's personal beliefs and values.

Figure 5: Ownership attack for blind evaluation on LLMs: Reviewers can pose any ownership-related questions and select their preferred model solely based on the ownership of the model. LMSys doesn't count votes if the model's identities are revealed during conversation

dition to that, considering same score for *tie* and *both-bad* can significantly change leaderboard position. We recommend to use *tie* as 0.5 point and *both-bad* as 0 point.

A.6.3 LLMs as Evaluators

Since human evaluation is time-consuming and difficult to reproduce, the instruction-following capabilities of LLMs have also inspired researchers to use certain LLMs as a judge to evaluate the responses generated by other LLMs (Chern et al., 2024; Fu et al., 2023b; Hada et al., 2023; Huang et al., 2024a; Kobayashi et al., 2024; Kocmi and Federmann, 2023; Lu et al., 2024). While prior work mostly utilized general-purpose closed-source LLMs-as-a-judge, the recently proposed Prometheus 2 (Kim et al., 2024) model is an open-source variant which is specifically trained for qualitative evaluation of model-generated responses and demonstrated higher correlation with humans.

However, research by (Wang et al., 2023b) and (Shen et al., 2023) has highlighted potential limitations in using LLM as evaluators, suggesting that while LLMs can excel in specific areas like translation quality and grammatical error correction ((Kobayashi et al., 2024; Kocmi and Federmann, 2023)), their effectiveness as evaluators may vary significantly across different tasks. This highlights the ongoing debate and research into the capabilities and limitations of LLMs as evaluators in diverse linguistic domains.