

LEFT

Survey Article

LEFT

Gao, Hu, Yin, Ruan, Pu, and Wan 1

LEFT

LEFT

RIGHT

COMPUTATIONAL LINGUISTICS

Just Accepted MS.

https://doi.org/10.1162/coli_a_00561

© 2025 Association for Computational Linguistics Published under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license

LLM-based NLG Evaluation: Current Status and Challenges

LEFT

LEFT

RIGHT

COMPUTATIONAL LINGUISTICS

Just Accepted MS.

https://doi.org/10.1162/coli_a_00561

© 2025 Association for Computational Linguistics Published under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license

Mingqi Gao*

Peking University

gaomingqi@pku.edu.cn

LEFT

LEFT

RIGHT

COMPUTATIONAL LINGUISTICS

Just Accepted MS.

https://doi.org/10.1162/coli_a_00561

© 2025 Association for Computational Linguistics Published under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license

Xunjian Yin

Peking University

xjyin@pku.edu.cn

LEFT

LEFT

RIGHT

COMPUTATIONAL LINGUISTICS

Just Accepted MS.

https://doi.org/10.1162/coli_a_00561

© 2025 Association for Computational Linguistics Published under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license

Xiao Pu

Peking University

puxiao@stu.pku.edu.cn

LEFT

Evaluating natural language generation (NLG) is a vital but challenging problem in natural language processing. Traditional evaluation metrics mainly capturing content (e.g. n-gram) overlap between system outputs and references are far from satisfactory, and large language models (LLMs) such as ChatGPT have demonstrated great potential in NLG evaluation in recent years. Various automatic evaluation methods based on LLMs have been proposed, including metrics derived from LLMs, prompting LLMs, fine-tuning LLMs, and human-LLM collaborative evaluation. In this survey, we first give a taxonomy of LLM-based NLG evaluation methods, and discuss their pros and cons, respectively. Lastly, we discuss several open problems in this area and point out future research directions.

LEFT

Contents

LEFT

1	Introduction	1
2	LLM-derived Metrics	3
3	Prompting LLMs	5
4	Fine-tuning LLMs	11
5	Human-LLM Collaborative Evaluation	15
6	Conclusions and Future Trends	17

LEFT

1. Introduction

LEFT

The evaluation of natural language generation (NLG) is an important but challenging issue. The lack of a single standard answer and the presence of multiple quality criteria make evaluating NLG more challenging than other NLP tasks. For example, in news summarization, a good summary should capture the key information from the source

LEFT

* Equal contribution.

LEFT

LEFT

Survey Article

LEFT

Gao, Hu, Yin, Ruan, Pu, and Wan 2

LEFT

document, remain faithful to the source document, and be expressed in logically coherent and fluent language, but there is not a single "correct" way to achieve this. The inherent difficulty of NLG evaluation means that human evaluation is always needed and regarded as the gold standard. However, due to the high cost and time-consuming nature of human evaluation, automatic evaluation metrics remain indispensable and play a crucial role in model development. Over the past two decades, many automatic evaluation metrics such as BLEU (Papineni et al. 2002) and BARTScore (Yuan, Neubig, and Liu 2021) have been developed, but none have been fully satisfactory. Some studies (Sai et al. 2021; He et al. 2023a) have highlighted their deficiencies in robustness, such as insensitivities, biases, or even loopholes when evaluating challenging texts.

Recently, large language models (LLMs) have emerged and demonstrated unprecedented capacities in following instructions, understanding content, and generating text, which inspires researchers to utilize LLMs for NLG evaluation. Although this is a research direction that only emerged in 2023, the past year has seen an enormous amount of relevant work. While there have been surveys on automatic evaluation metrics or human evaluation practices in NLG evaluation (Celikyilmaz, Clark, and Gao 2020; Härmäläinen and Al-Najjar 2021; Zhou et al. 2022; Sai, Mohankumar, and Khapra 2023; Gehrmann, Clark, and Sellam 2023; Zhou, Ringeval, and Portet 2023), none of them addresses LLM-based evaluation approach, and a comprehensive survey of this area is urgently needed.

The survey will mainly focus on research on LLM-based approaches for NLG evaluation, which involves language models with over one billion parameters. Moreover, it mainly considers the typical scope of NLG tasks where both input and output are natural languages including machine translation, text summarization, story generation, dialogue response generation, data-to-text, text simplification, paraphrase generation, grammatical error correction, and creative writing. Broader areas like evaluation of LLMs are not included (Zhuang et al. 2023; Chang et al. 2024) because this work focuses on LLMs used for evaluation, rather than the evaluation of LLMs' capabilities. We search the literature on Google Scholar with an end date of June 2024 with keywords. Since this is a new direction that emerged in 2023, a considerable number of arXiv preprints are included in addition to papers published in *ACL venues or other related venues. About 100 pieces of work will be included. To maintain focus, this paper neither discusses datasets and benchmarks in NLG evaluation (Gehrmann et al. 2021; Kim et al. 2024a) nor analyzes evaluation metrics statistically (Ni'mah et al. 2023; Xiao et al. 2023).

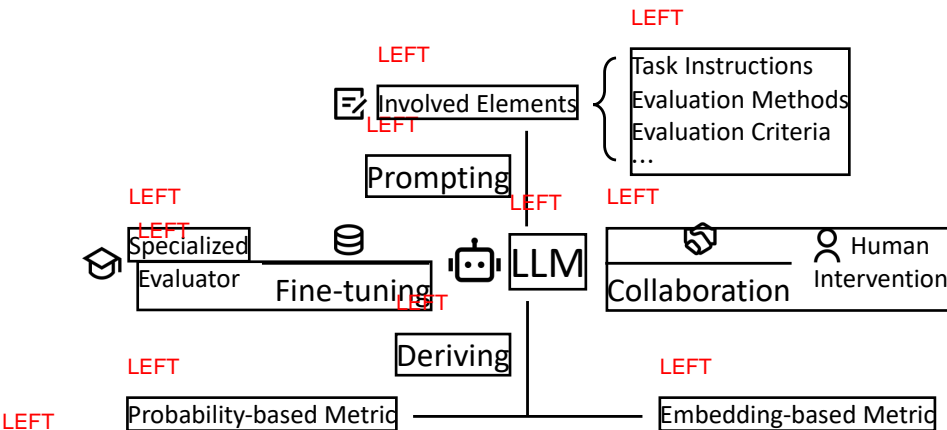


Figure 1

Schematic representation of our proposed four categories of LLM-based NLG evaluation.

LEFT

Downloaded from http://direct.mit.edu/coli/article-pdf/doi/10.1162/coli_a_00561/2513169/coli_a_00561.pdf by guest on 18 April 2025

LEFT

Survey Article

LEFT

Gao, Hu, Yin, Ruan, Pu, and Wan 3

LEFT

As shown in Figure 1, we categorize related studies into four categories according to how LLMs are utilized for NLG evaluation:

LEFT

- **LLM-derived Metrics (§ 2):** developing or deriving evaluation metrics from embeddings or generation probabilities of LLMs.
- **Prompting LLMs (§ 3):** directly inquiring of existing LLMs via specific prompts and processes designed for evaluation.
- **Fine-tuning LLMs (§ 4):** using labeled evaluation data to fine-tune existing LLMs and improving their NLG evaluation capabilities.
- **Human-LLM Collaborative Evaluation (§ 5):** leveraging distinctive strengths of both human evaluators and LLMs to achieve robust and nuanced evaluations through human-LLM collaboration.

LEFT

LLMs have driven NLG evaluation toward a more human-centered direction, and the four categories we propose reflect this evolution: LLM-derived metrics are a continuation of traditional evaluation metrics and can only handle coarse-grained evaluation; prompting and fine-tuning methods enable users to express flexible evaluation requirements in natural language; collaborative evaluation takes it a step further, making it possible for humans and LLMs to leverage their strength respectively. We will review each type of evaluation method and discuss the pros and cons, respectively. Last but not least, we will provide our suggestions and conclusions, and discuss future directions in this area (§ 6).

It is worth stating that since LLM-based evaluation has shown unprecedented generality across NLG tasks, we do not summarize the literature for each task separately. Nevertheless, we will draw a list documenting all the approaches in this survey, indicating which NLG tasks each approach has been experimented on.

LEFT

2. LLM-derived Metrics

LEFT

LLM-derived metrics can be viewed as a continuation of early model-based NLG evaluation metrics such as BERTScore and BARTScore, replacing traditional pre-trained language models with stronger LLMs. Such works can be categorized into two main types: embedding-based metrics (Es et al. 2023) and probability-based metrics. The latter can be further divided into two categories based on different ways of using probabilities: directly converting the probabilities into scores (Fu et al. 2023a; Varshney et al. 2023) and leveraging the variation in probabilities under changed conditions (Jia et al. 2023; Xie et al. 2023).

LEFT

2.1 Embedding-based Metrics

LEFT

The embedding-based methods, like BERTScore, generally utilize representations of language models and thus compute the semantic similarity between the reference and the target text to evaluate, with different possible ways of implementation. However, unlike traditional embedding-based evaluation metrics that require references, many LLM-based embedding evaluation metrics do not. This is because their application scenarios and implementation methods differ from those of traditional metrics. For example, when Es et al. (2023) evaluate the answer relevance of Retrieval Augmented

LEFT

Survey Article

LEFT

Gao, Hu, Yin, Ruan, Pu, and Wan 4

LEFT

Generation, given the original question q and the answer Y to be evaluated, they first prompt the LLM to generate n possible questions q_i for Y . Then, the relevance of Y is represented by the average similarity between q_i and q , denoted as $\sum_{i=1}^n \text{sim}(q_i, q)$, where $\text{sim}(q_i, q)$ refers to the cosine similarity of the embeddings of q_i and q . The embedding is generated by OpenAI *text-embedding-ada-002*, which can efficiently convert text into a 1536-dimensional vector, capturing semantic information and ensuring that similar texts are positioned close to each other in the vector space. Furthermore, Sheng et al. (2024) developed a more sophisticated method based on embeddings from the open-source decoder-only LLM, utilizing Principal Component Analysis to adapt it for both pointwise scoring and pairwise comparison.

LEFT

2.2 Probability-based Metrics

LEFT

To better utilize the knowledge inherent in language models, probability-based methods like BARTScore formulate text generation evaluation as conditional probability comparison, positing that the better the quality of the target text, the higher the likelihood that models should be able to generate it. Recently, GPTScore (Fu et al. 2023a) has established tailored evaluation templates for each aspect to effectively guide multiple LLMs for NLG evaluation, including GPT3 (Brown et al. 2020), OPT (Zhang et al. 2022), and FLAN (Chung et al. 2022). The core idea of GPTScore is that a good generative language model is more likely to assign higher probabilities to high-quality text generated in response to a given instruction and context. Specifically, given a generative large language model θ , context information X (such as a source document), output text $Y = \{y_1, y_2, \dots, y_m\}$ containing m tokens to be evaluated, and instruction I that specifies the requirement for the LLMs to generate text that can flexibly correspond to different evaluation aspects (e.g., *generating a factually consistent summary* for the aspect of consistency), GPTScore is defined as:

$$\text{GPTScore}(X, Y, I, \theta) = \frac{1}{m} \sum_{i=1}^m \log P(y_i | y_{<i}, X, I, \theta)$$

LEFT

Similarly, Murugadoss et al. (2024) score the task output Y to be evaluated by its perplexity under the corresponding large language model θ , given only the task context X . They believe this approach is unbiased by prompts, which transparently measures alignment with model training data. Furthermore, such methods have also been applied to the hallucination detection of the LLM-generated text (Varshney et al. 2023) with three different attempts for calculating the probability score.

On the other hand, some works leverage the variation in probabilities under changed conditions as the evaluation metric. FFLM (Jia et al. 2023) proposes to evaluate the faithfulness of the target text by calculating a combination of probability changes based on the intuition that the generation probability of a given text segment increases when more consistent information is provided, and vice versa. Similarly, DELTAScore (Xie et al. 2023) measures the quality of different story aspects according to the likelihood difference between pre- and post-perturbation states with LLMs including GPT-3.5 (text-davinci-003) that provide logits. They believe that the sensitivity to specific perturbations indicates the quality of related aspects, and their experiments demonstrate the effectiveness of their approach.

LEFT

LEFT

Survey Article

LEFT

Gao, Hu, Yin, Ruan, Pu, and Wan 5

LEFT

2.3 Pros and Cons

LEFT

Traditional NLG evaluation approaches always fall short due to their surface-form similarity when the target text and reference convey the same meaning but use different expressions. In contrast, LLM-derived metrics offer a remedy for the limitation and demonstrate stronger correlations with human judgments benefiting from the evolving modeling techniques. However, the flaws within LLMs can lead to some issues, as introduced in the following:

Robustness. Some research has investigated the robustness of LLM-derived metrics and found that they lack robustness in different attack scenarios. Specifically, [He et al. \(2023b\)](#) develops a set of stress tests to assess the robustness of various model-based metrics on some common NLG tasks. They show a catalogue of the blind spots and potential errors identified that are not detected by different metrics.

Efficiency. Compared to traditional metrics, LLM-derived evaluation methods are more time-consuming and require more computational resources, especially when adopting LLMs with quite large parameter scales. To address this, [Eddine et al. \(2022\)](#) propose an approach to learning a lightweight version of LLM-derived metrics, and some fast LLM inference and serving tools like popular vLLM ([Kwon et al. 2023](#)) have been launched. vLLM improves memory utilization during inference through the PagedAttention algorithm, as well as the optimized memory management and batching strategies, thereby increasing LLMs' generation throughput. However, closed-source LLMs often do not make their parameters, representations, or logits public and available, thus making it impossible to apply LLM-derived methods to them.

Fairness. [Sun et al. \(2022\)](#) assess the social bias across various metrics for NLG evaluation on six sensitive attributes: race, gender, religion, physical appearance, age, and socioeconomic status. Their findings reveal that model-based metrics carry noticeably more social bias than traditional metrics. Relevant biases can be categorized into two types: intrinsic bias encoded within pre-trained language models and extrinsic bias injected during the computation of similarity. Therefore, current LLM-derived methods may have similar issues.

LEFT

3. Prompting LLMs

LEFT

The remarkable generation abilities of LLMs have expanded the possibilities for NLG evaluation. For a long time, human evaluation has been viewed as the gold standard for NLG evaluation. Recently, some studies claim that LLMs are on par with crowd-sourcing annotators in several tasks ([Törnberg 2023](#); [Gilardi, Alizadeh, and Kubli 2023](#); [Ostyakova et al. 2023](#); [Cegin, Simko, and Brusilovsky 2023](#)). This raises questions about whether LLMs could replace human evaluators. Studies in this area often involve feeding LLMs with detailed prompts that include both instructions and the text to be evaluated, with LLMs producing the evaluation outcomes. An example of prompting LLMs is shown in [Figure 2](#). From this example, we can see that such a prompt is quite similar to the guidelines given to human evaluators. The main differences between this prompting method for LLMs and LLM-derived metrics are twofold: (1) LLM-derived metrics generally do not involve highly human-like prompts that require the LLM to perform an evaluation. (2) The evaluation results from prompting LLMs are typically generated directly by the LLM, whereas LLM-derived metrics require further transformation from embeddings and probabilities. We will describe existing works according to the five elements that they mainly focus on:

LEFT

LEFT

Survey Article

LEFT

Gao, Hu, Yin, Ruan, Pu, and Wan 7

LEFT

LEFT

RIGHT

Prompt

Please act as an impartial and helpful evaluator for natural language generation (NLG). You will be given a news article and a summary written for it. Your task is to evaluate the quality of the summary strictly based on the given evaluation criterion. Please make sure you read and understand these instructions carefully.

Evaluation Criterion:

Consistency: the factual alignment between the summary and the summarized source. A factually consistent summary contains only statements that are entailed by the source document. Annotators were also asked to penalize summaries that contained hallucinated facts.

News Article:

Paul Merson has restarted his row with Andros Townsend after the Tottenham midfielder was brought on with only seven minutes remaining in his team's 0-0 draw with Burnley on Sunday. 'Just been watching the game, did you miss the coach? #RubberDub #7minutes,' Merson ...

Summary:

Paul Merson was brought on with only seven minutes remaining in his team's 0-0 draw with Burnley. Andros Townsend scored the Tottenham midfielder in the 89th minute ...

Evaluation Form:

Begin the evaluation by providing your analysis concisely and accurately, and then on the next line, start with "Rating:" followed by your rating on a Likert scale from 1 to 5 (higher means better).

Response

The summary contains several inconsistencies with the facts presented in the article. Firstly, it states that "Paul Merson was brought on with only seven minutes remaining in his team's 0-0 draw with Burnley," which is incorrect as it was Andros Townsend who was brought on in that context, not ...
Rating: 2

LEFT

LEFT

Figure 2

An example of prompting LLMs to evaluate the aspect of consistency of the summary. There are role and interaction, task instructions, evaluation criteria, input content, and evaluation methods in the prompt, as well as the evaluation results, including the rating and explanation generated by LLMs.

LEFT

evaluation. Chiang and Lee (2023a) have conducted relevant studies early, using a Likert scale from 1 to 5 to evaluate story generation and adversarial attacks with InstructGPT (Ouyang et al. 2022) and ChatGPT¹, showing that the evaluation results of LLMs are consistent with expert human evaluators. Kocmi and Federmann (2023b) discover GPT-3.5 and GPT-4 achieve the state-of-the-art accuracy of evaluating translation quality compared to human labels with a rating scale from 1 to 5 or 0 to 100, outperforming all the results from the metric shard task of WMT22 (Freitag et al. 2022). Furthermore, Wang et al. (2023a) experiment on five datasets across summarization, story generation, and data-to-text, and ChatGPT with similar rating scales achieves the state-of-the-art or comparative correlations with human judgments in most settings, compared with prior metrics. Similar conclusions are also observed in open-domain dialogue response generation (Lin and Chen 2023). Besides English, Mendonça et al. (2023) show that

LEFT

¹ <https://openai.com/blog/chatgpt/>

LEFT

LEFT

Survey Article

Gao, Hu, Yin, Ruan, Pu, and Wan 8

LEFT

ChatGPT with simple rating prompts is a strong evaluator for multilingual dialogue evaluation, surpassing prior metrics based on encoders.

Comparison. Different from absolute scoring, comparison refers to choosing the better of the two. Luo, Xie, and Ananiadou (2023); Gao et al. (2023) use ChatGPT to compare the factual consistency of two summaries. AuPEL (Wang et al. 2023d) evaluate personalized text generation from three aspects in the form of comparison with the PaLM 2 family (Anil et al. 2023). According to Liusie, Manakul, and Gales (2023), pairwise comparison is better than scoring when medium-sized LLMs (e.g. FlanT5 (Chung et al. 2022) and Llama2 (Touvron et al. 2023)) are adopted as evaluators.

Ranking. Ranking can be viewed as an extended form of comparison. In comparison, only two examples are involved at a time, whereas in ranking, the order of more than two examples needs to be decided at once. Ji et al. (2023) use ChatGPT to rank five model-generated responses across several use cases at once, indicating the ranking preferences of ChatGPT align with those of humans to some degree. Similarly, GPTRank is a method to rank summaries in a list-wise manner (Liu et al. 2023c). Moreover, Liu et al. (2023b) compare different evaluation methods in LLM-based summarization including scoring, comparison, and ranking, showing that the optimal evaluation method for each backbone LLM may vary.

Boolean QA. Boolean QA requires LLMs to answer "Yes" or "No" to a question. It is adopted more in scenarios where human annotations are binary, such as grammaticality (Hu et al. 2023), faithfulness of summaries and statements (Luo, Xie, and Ananiadou 2023; Gao et al. 2023; Es et al. 2023; Hu et al. 2023), factuality of generated text (Fu et al. 2023b; Guan et al. 2023; Manakul, Liusie, and Gales 2023), and answerability of generated questions (Wang, Funakoshi, and Okumura 2023).

Error Analysis. Error Analysis refers to the evaluation of a text by looking for errors that occur in the text according to a set of predefined error categories. Multidimensional Quality Metrics (MQM) (Jain et al. 2023) is an error analysis framework prevalent in machine translation evaluation. According to MQM, Lu et al. (2023); Kocmi and Federmann (2023a) use ChatGPT or GPT-4 to automatically detect translation quality error spans. BOOOOKSCORE (Chang et al. 2023), an LLM-based evaluation metric, assesses the coherence of book summaries by identifying eight types of errors.

3.2 Task Instructions

In human evaluation, task instruction usually comes in the form of a task description or evaluation steps. They can also exist at the same time. The task description states the annotation in a more general way, and the evaluation steps, which can be considered as Chain-of-Thought, explicitly describe what to do at each step. In the context of prompting LLMs for NLG evaluation, we discuss three broad categories of influences: various templates of prompts (Leiter et al. 2023; Kim et al. 2023a; Kotonya et al. 2023; He, Zhang, and Roth 2023), in-context examples (Jain et al. 2023; Kotonya et al. 2023; Hasanbeig et al. 2023), and whether LLMs are required to provide analyses or explanations (Chiang and Lee 2023b; Naismith, Mulcaire, and Burstein 2023).

Form and requirements. Several studies from an Eval4NLP 2023 shared task (Leiter et al. 2023) have explored task instructions in various settings. For example, Kim et al. (2023a) conduct experiments on different templates and lengths of task descriptions and evaluation steps, finding that providing clear and straightforward instructions, akin to those explained to humans, is more effective. Kotonya et al. (2023) generate task instructions with LLMs or improve existing task instructions with LLMs. Moreover, Leiter and Eger (2024) conduct a larger-scale prompt exploration for the evaluation

LEFT

LEFT

Survey Article

LEFT

Gao, Hu, Yin, Ruan, Pu, and Wan 10

LEFT

3.4 Evaluation Criteria

LEFT

The evaluation targeting specific aspects is used in numerous studies of human evaluation for NLG, such as text summarization, story generation, dialogue, and text simplification. Evaluation criteria, i.e., the definitions of aspects are key in this context. Most evaluation criteria in LLM-based evaluation are directly derived from human evaluation. However, a few studies have attempted to let LLMs generate or improve evaluation criteria. Liu et al. (2023e) use a few human-rated examples as seeds to let LLMs draft some candidate evaluation criteria, and then further filter them based on the performance of LLMs using these criteria on a validation set, to obtain the final evaluation criteria. Kim et al. (2023c) designed an LLM-based interactive evaluation system, which involves using LLMs to review the evaluation criteria provided by users, including eliminating ambiguities in criteria, merging criteria with overlapping meanings, and decomposing overly broad criteria. Additionally, Ye et al. (2023a) propose a hierarchical aspect classification system with 12 subcategories, demonstrating that under the proposed fine-grained aspect definitions, human evaluation and LLM-based evaluation are highly correlated. Besides, the chain-of-aspects approach improves LLMs' ability to evaluate on a specific aspect by having LLMs score on some related aspects before generating the final score (Gong and Mao 2023).

LEFT

3.5 Role and Interaction

LEFT

We include in this section the evaluation strategies that either use the same LLMs in different ways or involve different LLMs (Bai et al. 2023; Li, Patel, and Du 2023; Cohen et al. 2023). The former can be further divided into chain-style (Yuan et al. 2024; Fu et al. 2023b; Hu et al. 2023) and network-style interactions (Chan et al. 2023; Zhang et al. 2023b; Saha et al. 2023; Wu et al. 2023).

Chain-style interaction. Inspired by human evaluators, Yuan et al. (2024) have LLMs score a batch of examples to be evaluated each time. Specifically, the evaluation process is divided into three stages: analysis, ranking, and scoring. Similar to QA-based evaluation metrics (Durmus, He, and Diab 2020), Fu et al. (2023b) assess the faithfulness of summaries in two stages: treating LLMs as question generators to generate a question from the summary; then having LLMs answer the question using the source document. Differently, when Hu et al. (2023) use GPT-4 to evaluate the faithfulness of summaries, it first asks GPT-4 to extract event units from the summary, then verifies whether these event units meet the requirements, and finally judges whether the event units are faithful to the source document.

LEFT

Network-style interaction. Unlike chain-style interactions, network-style interactions involve the dispersion and aggregation of information. In network-style interactions, LLMs on the same layer play similar roles. ChatEval (Chan et al. 2023) is a framework for evaluating content through debates among multiple LLMs, with three communication strategies designed among the three types of LLMs: One-By-One, Simultaneous-Talk, and Simultaneous-Talk-with-Summarizer. Zhang et al. (2023b) find that under certain conditions, widening and deepening the network of LLMs can better align its evaluation with human judgments. Saha et al. (2023) propose a branch-solve-merge strategy, assigning LLMs the roles of decomposing problems, solving them, and aggregating answers, thereby improving the accuracy and reliability of evaluations. Wu et al. (2023) assume that different people such as politicians and the general public have different concerns about the quality of news summaries, use LLMs to play different roles in evaluation accordingly, and aggregate the results finally.

LEFT

LEFT

Survey Article

LEFT

Gao, Hu, Yin, Ruan, Pu, and Wan 11

LEFT

Different LLMs. Different from having the same LLM play different roles, some research has used different LLMs (such as GPT-4 and Claude) in their studies. The use of a single LLM as evaluator may introduce bias, resulting in unfair evaluation results. In light of this, [Bai et al. \(2023\)](#) design a decentralized Peer-examination method, using different LLMs as evaluators and then aggregating the results. Further, [Li, Patel, and Du \(2023\)](#) let different LLMs serve as evaluators in pairwise comparisons and then have them go through a round of discussion to reach the final result. Additionally, [Cohen et al. \(2023\)](#) evaluate the factuality of texts through the interaction of two LLMs, where the LLM that generated the text acts as the examinee and the other LLM as the examiner.

LEFT

3.6 Pros and Cons

LEFT

The benefits of prompting LLMs for NLG evaluation are exciting. First, for the first time, people can express evaluation criteria and evaluation methods in natural language within the prompts given to LLMs, providing great flexibility. Where previously people needed to design specific evaluation metrics for different NLG tasks or even different aspects of a single task, now they only need to modify the prompts for LLMs. Secondly, surprisingly, LLMs have the ability to generate explanations while assessing texts, making this approach somewhat interpretable. Furthermore, in many NLG tasks, prompting LLMs for evaluation has achieved state-of-the-art correlations with human judgments.

However, as many studies have pointed out, this type of approach still has many limitations. [Wang et al. \(2023b\)](#) note that when using ChatGPT and GPT-4 for pairwise comparisons, the order of the two texts can affect the evaluation results, which is known as position bias. To alleviate this issue, [Li et al. \(2023c\)](#) propose a strategy of splitting, aligning, and then merging the two texts to be evaluated into the prompt. Also, LLM evaluators tend to favor longer, more verbose responses ([Zheng et al. 2023](#)) and responses generated by themselves ([Liu et al. 2023a](#)). [Wu and Aji \(2023\)](#) show that compared to answers that are too short or grammatically incorrect, answers with factual errors are considered better by LLMs. [Liu et al. \(2023d\)](#) demonstrate through adversarial meta-evaluation that LLMs without references are not suitable for evaluating dialogue responses in closed-ended scenarios: they tend to score highly on responses that conflict with the facts in the dialogue history. [Zhang et al. \(2023a\)](#) also present the robustness issues of LLMs in dialogue evaluation through adversarial perturbations. [Shen et al. \(2023\)](#) indicate that LLM evaluators have a lower correlation with human assessments when scoring high-quality summaries. In addition, [Hada et al. \(2023\)](#) state that LLM-based evaluators have a bias towards high scores, especially in non-Latin languages like Chinese and Japanese. [Bavaresco et al. \(2024\)](#) find that the performance of LLM-based evaluators exhibits significant variance depending on the dataset, evaluation criteria, and whether the evaluated texts are human-generated. Beyond these shortcomings of performance, both ChatGPT and GPT-4 are proprietary models, and their opacity could lead to irreproducible evaluation results.

LEFT

4. Fine-tuning LLMs

LEFT

As mentioned above, despite the exciting performance of prompting LLMs like ChatGPT and GPT-4 for NLG evaluation, several shortcomings in practice are inevitable, such as high costs, possibly irreproducible results, and potential biases in LLMs. In response, recent research has shifted towards fine-tuning smaller, open-source LLMs specifically for evaluation purposes, aiming to achieve performance close to GPT-4 in NLG evaluation. Representative works of this type include PandaLM ([Wang et al.](#)

LEFT

LEFT

Survey Article

LEFT

Gao, Hu, Yin, Ruan, Pu, and Wan 12

LEFT				
Method	Data Construction			Foundation LLM
	Instruction Source	Annotator	Scale	LEFT
PandaLM	Alpaca 52K	GPT-3.5	500K	LLaMA 7B
Prometheus	GPT-4 Construction	GPT-4	100K	LLaMA-2-Chat 7B & 13B
Prometheus 2	FEEDBACK COLLECTION	GPT-4	200K	Mistral-7B Mistral-8x7B
Shepherd	Community Critique Data	Human	1317	LLaMA 7B
TIGERScore	100 & 9 NLP Tasks Data 23 Distinctive Text Generation Datasets	GPT-4	48K	LLaMA-2 7B & 13B
INSTRUCTSCORE	GPT-4 Construction	GPT-4	40K	LLaMA 7B
AUTO-J	Real-world User Queries from Preference Datasets	GPT-4	4396	LLaMA-2-Chat 13B
CritiqueLLM	AlignBench & ChatGPT Augmentation	GPT-4	9332	ChatGLM-2 6B, 12B & 66B
JudgeLM	GPT4All-LAION, ShareGPT Alpaca-GPT4 & Dolly-15K	GPT-4	100K	Vicuna 7B, 13B & 33B
Themis	NLG-Eval with 58 NLG Evaluation Datasets	Human & GPT-4	67K	Llama-3-8B
Self-Taught	Screened WildChat	Llama-3-70B	20K	Llama-3-70B
CompassJudger-1	Sampling from existing datasets	Mixture	900K	Qwen-2.5 1.5B, 7B, 14B & 32B

LEFT

Table 2

Comparison of the different key components among the representative methods of fine-tuning LLMs (Part 1).

LEFT

2023e), Prometheus (Kim et al. 2023b), Prometheus 2 (Kim et al. 2024b), Shepherd (Wang et al. 2023c), TIGERScore (Jiang et al. 2023), INSTRUCTSCORE (Xu et al. 2023), Auto-J (Li et al. 2023a), CritiqueLLM (Ke et al. 2023), JudgeLM (Zhu, Wang, and Wang 2023), Themis (Hu et al. 2024), CompassJudger-1 (Cao et al. 2024) and Self-Taught (Wang et al. 2024). Their main ideas are similar, involving the elaborate construction of high-quality evaluation data, followed by fine-tuning open-source foundation LLMs with specific methods. Nevertheless, there are certain discrepancies in the designs across different works, such as the usage of references and evaluation criteria. We have summarized the key different components of these methods in Table 2 and Table 3 for comparison, which will be elaborated in the following sections.

LEFT

4.1 Data Construction

LEFT

Diverse data with high-quality annotations is crucial for the fine-tuning of evaluation models, which mainly involves task scenarios, inputs, target texts to evaluate, and evaluation results. Early NLG evaluation research primarily focused on conventional NLG tasks, such as summarization and dialogue generation. Thus, the task scenarios, inputs, and target texts refer to the corresponding NLP task, source inputs of the task, and outputs generated by specialized systems based on task requirements, respectively.

LEFT

Survey Article

LEFT

Gao, Hu, Yin, Ruan, Pu, and Wan 14

LEFT

consistent training data. Self-Taught uses the evaluation results from the model to fine-tune itself (Llama-3-70B), considering it already possesses strong capabilities. During the construction, these studies basically all design detailed prompts or guidance and apply heuristic filtering strategies and post-processing methods to mitigate noise. Overall, despite the possible higher quality of human annotation, the corresponding drawback is the difficulty in constructing large-scale datasets, which in turn may hinder adequate model training, while using LLMs for construction is the opposite situation.

LEFT

4.2 Evaluation Method

LEFT

As with prompting LLMs, the evaluation methods adopted in these works are highly diversified, involving different evaluation criteria, result modes, and usages of the reference. Given that current instruction-response scenarios encompass different types of tasks, it is unsuitable to specify unified evaluation criteria as in traditional NLG tasks. However, some works still do it this way, while some other methods let LLM annotators adaptively and implicitly reflect the required criteria in their evaluations, like PandaLM, TIGERScore, and AUTO-J. In particular, AUTO-J has meticulously crafted 332 evaluation criteria, matched to different tasks. Furthermore, Prometheus and Themis explicitly incorporate evaluation criteria into the evaluation instructions, and CompassJuder can work either with or without evaluation criteria, enabling flexible evaluation based on various customized criteria.

More details about the evaluation methods are shown in Table 3. All the works require models to provide detailed information, such as reasons for their evaluation results. And the MQM mode can achieve more informative error analysis, offering stronger interpretability. Moreover, some works do not necessarily require references and then have greater value in practice. And a more optimal method is to concurrently support both reference-based and reference-free evaluations as JudgeLM and CritiqueLLM.

LEFT

4.3 Fine-tuning Implementation

LEFT

The fine-tuning process is implemented by different studies on their selected open-source foundation LLMs, like LLaMA, and respective constructed data, with some targeted settings. Specifically, Prometheus maintains balanced data distributions during fine-tuning, including the length and label. JudgeLM eliminates potential biases by randomly swapping sample pairs to be compared and randomly removing references. INSTRUCTSCORE utilizes GPT-4 to provide error annotations for the intermediate outputs of the fine-tuned model for further supervised reinforcement. And based on some preliminary experiments and manual analysis, TIGERScore determines appropriate ratios of different types of data during fine-tuning, which are claimed to be crucial by them. Moreover, CritiqueLLM implements separately, with and without references, and explores the effects of data and model scale. Themis employs additional rating-guided preference optimization after the fine-tuning process. Specifically, Self-Taught utilizes the evaluation results of the fine-tuned model itself for self-iterative optimization, leading to surprising improvements. Compared to the vanilla fine-tuning setting, these methods have improved the efficiency of model training and the robustness of evaluations.

LEFT

LEFT

Survey Article

LEFT

Gao, Hu, Yin, Ruan, Pu, and Wan 15

LEFT

4.4 Pros and Cons

LEFT

The shortcomings of prompting LLMs for NLG evaluation can be significantly alleviated by the customized construction of training data and specifically fine-tuned LLMs. For instance, most models in Table 2 have less than 14B parameters, facilitating low-cost inference in practice and good reproducibility, with performance comparable to GPT4. And specific measures can be adopted to prevent certain biases found in GPT4 during different stages, such as randomly changing the order of training pairs for position bias. Furthermore, this type of approach allows for continuous iteration and improvement of the model to address potential deficiencies or emerging issues discovered in future applications.

However, some inherent biases associated with GPT4 may still persist, like self-biases, as the data construction of most methods employs GPT4 for critical evaluation annotation. On the other hand, many studies have chosen open-source foundation LLMs spanning three generations of the Llama series. With the recent rapid updates and improvements of open-source LLMs, it is intuitive that employing a more powerful foundation LLM should lead to better evaluation performance of the fine-tuned model. However, this means repetitive fine-tuning processes and computational expenses from scratch since directly migrating existing fine-tuned models to the new foundation LLM is challenging.

Additionally, although many existing methods aspire to more flexible and comprehensive evaluation through fine-tuning, demanding excessive evaluation settings may ultimately lead to poor performance or failure in model training, as AUTO-J and CritiqueLLM found difficult on criteria and references, respectively. However, there are some disagreements here since Prometheus, JudgeLM, and CompassJuderger show different results, indicating such a seemingly straightforward fine-tuning process is actually quite complex. Moreover, considering the different evaluation settings in existing works, conducting a horizontal comparison among them is challenging. These issues require further exploration in future research.

LEFT

5. Human-LLM Collaborative Evaluation

LEFT

Human evaluation remains the gold standard for NLG due to its ability to capture nuanced aspects of quality. However, it is expensive, time-consuming, and prone to subjective biases (van der Lee et al. 2021; Deriu et al. 2021; Li et al. 2023b). While LLMs offer a promising avenue for automated evaluation, their reliability and correlation with human judgment are still areas of active development (Li et al. 2023c; Liu et al. 2023d). Human-LLM collaborative evaluation seeks to leverage the strengths of both: the nuanced judgment of humans and the scalability and efficiency of LLMs. This chapter explores emerging paradigms in this collaborative space, focusing on how humans and LLMs can work together to improve the accuracy, efficiency, and trustworthiness of NLG evaluation. This includes collaborative approaches like: traditional evaluation tasks such as scoring and explaining (Zhang, Ren, and de Rijke 2021; Li et al. 2023b); general evaluation tasks such as testing and debugging (Ribeiro and Lundberg 2022); auditing NLG models to ensure fairness (Rastogi et al. 2023); aligning LLM-assisted evaluation of LLM outputs with human preferences (Shankar et al. 2024); addressing the intricate challenge of scalable oversight (Amodei et al. 2016; Saunders et al. 2022).

LEFT

Survey Article

LEFT

Gao, Hu, Yin, Ruan, Pu, and Wan 17

LEFT

strengths in schematization and hypothesis testing, while LLMs assist in identifying a wide range of failure modes. This collaborative approach uncovered both previously known and under-reported issues.

Evaluating LLMs on complex tasks can be challenging even for humans (Chen et al. 2021; Nakano et al. 2021; Li et al. 2022; Menick et al. 2022). The concept of scalable oversight (Amodei et al. 2016) suggests using AI to assist in evaluation. Saunders et al. (2022) explore using LLM-generated critiques to help humans identify flaws in model outputs, demonstrating that this form of assistance improves human performance. What's more, Pozdniakov et al. (2024) focus on designing conversational user interfaces, which helps educators to use LLMs to evaluate assignments of students.

LEFT

5.3 Pros and Cons

LEFT

Human-LLM collaborative evaluation offers a compelling balance between the accuracy of human judgment and the efficiency of automated methods. Key advantages include: (1) **Efficiency and Cost-Effectiveness**: LLMs can significantly reduce the time and resources required for evaluation. (2) **Complementary Strengths**: Humans excel at nuanced judgment and critical thinking, while LLMs excel at processing large amounts of data and generating diverse outputs. (3) **Improved Accuracy**: Combining human and LLM strengths can lead to more accurate and reliable evaluations than either approach alone.

However, challenges remain: (1) **Prompt Sensitivity**: LLM evaluation results can be sensitive to the phrasing of prompts, requiring careful prompt engineering (Li et al. 2023b; Rastogi et al. 2023). (2) **Confidence Calibration**: LLMs' ability to accurately assess their own confidence is still limited, making it difficult to know when to trust their judgments. (3) **Need for Human Oversight**: While reduced, human supervision is still necessary, limiting the potential for full automation. (4) **Explainability**: Ensuring the collaborative process is transparent and understandable can be challenging.

LEFT

6. Conclusions and Future Trends

LEFT

6.1 Comparison with traditional evaluation metrics

LEFT

Traditional evaluation metrics are criticized for their poor correlation with human judgments (Stent, Marge, and Singhai 2005), uninterpretable evaluation results (Zhang, Vogel, and Waibel 2004), and inability to adapt to specific evaluation criteria (Wiseman, Shieber, and Rush 2017), which are being greatly mitigated by LLM-based evaluation. However, the higher cost, the requirements for computing resources, and the issues of reproducibility may be the downside.

LEFT

6.2 Comparison between different types of LLM-based NLG evaluation.

LEFT

We compare different types of LLM-based evaluation according to flexibility and reproducibility due to the difficulty of comparing the effectiveness of different types of methods in various scenarios.

LEFT

Flexibility. Human-LLM Collaborative Evaluation > Prompting LLMs > Fine-tuning LLMs > LLM-derived Metrics. Human-LLM Collaborative Evaluation involves human annotators, which provides the highest flexibility. LLM-derived Metrics are typically designed to evaluate specific aspects, such as text similarity, and do not fully allow

LEFT

LEFT

Survey Article

LEFT

Gao, Hu, Yin, Ruan, Pu, and Wan 18

Method	Parameter	Coherence	Consistency	Fluency	Relevance	Overall
Traditional Metrics						
BERTScore	555M	0.285	0.151	0.186	0.302	0.231
BARTScore	400M	0.474	0.266	0.258	0.318	0.329
LLM-derived Metrics						
GPTScore (Fl5)	11B	0.456	0.438	0.424	0.343	0.415
GPTScore (OPT)	66B	0.359	0.453	0.380	0.337	0.382
GPTScore (GPT-3)	175B	0.434	0.449	0.403	0.381	0.417
GPTScore (Phi-4)	14B	0.319	0.436	0.386	0.154	0.324
GPTScore (Llama-3.1)	70B	0.415	0.478	0.437	0.288	0.405
GPTScore (Qwen-2.5)	72B	0.447	0.486	0.437	0.376	0.436
Prompting LLMs						
G-Eval (GPT-3.5)	7B	0.440	0.386	0.424	0.385	0.409
G-Eval (GPT-4)	14B	0.582	0.507	0.455	0.548	0.523
Phi-4	14B	0.479	0.454	0.421	0.452	0.451
Llama-3.1	70B	0.510	0.387	0.317	0.494	0.427
Qwen-2.5	72B	0.515	0.509	0.435	0.528	0.497
Fine-tuning LLMs						
INSTRUCTSCORE	7B	0.528	0.232	0.260	0.211	0.258
Prometheus 2	7B	0.403	0.318	0.269	0.356	0.336
Themis	3B	0.566	0.600	0.571	0.474	0.553
TIGERScore	13B	0.381	0.427	0.363	0.366	0.384
CompassJudger-1	32B	0.494	0.424	0.318	0.410	0.411
Human-LLM Collaborative Evaluation						
InteractEval (GPT-3.5 1st)	7B	0.583	0.630	0.734	0.614	0.640
InteractEval (GPT-3.5 2nd)	7B	0.590	0.614	0.726	0.623	0.638
InteractEval (GPT-4 1st)	7B	0.649	0.799	0.783	0.626	0.714
InteractEval (GPT-4 2nd)	7B	0.660	0.781	0.816	0.642	0.725

Table 4
Performance of different types of LLM-based NLG evaluation approaches on SummEval, where some results are from Fu et al. (2023a), Hu et al. (2024) and Chu, Kim, and Yi (2025).

LEFT

evaluation criteria to be expressed in natural language, making them the least flexible. When comparing Prompting LLMs and Fine-tuning LLMs, the former, which uses proprietary models, generally performs better at following instructions compared to smaller open-source models.

LEFT

Reproducibility. LLM-derived Metrics \approx Prompting LLMs $>$ Fine-tuning LLMs $>$ Human-LLM Collaborative Evaluation. Human-LLM Collaborative Evaluation requires human annotators, and the recruitment and training of these annotators pose greater challenges to reproducibility. LLM-derived Metrics and Prompting LLMs do not modify the existing models, and therefore have better reproducibility than Fine-tuning LLMs. However, they may still become non-reproducible if proprietary models are deprecated.

LEFT

Survey Article

LEFT

Gao, Hu, Yin, Ruan, Pu, and Wan 19

LEFT

Performance. Human-LLM Collaborative Evaluation > Fine-tuning LLMs \approx Prompting LLMs > LLM-derived Metrics. We compare the performance of different LLM-based evaluation approaches on the most commonly used NLG evaluation benchmark on summarization, SummEval (Fabbri et al. 2021), as shown in Table 4. When using the same LLMs, LLM-derived metrics perform worse than directly prompting LLMs for evaluation, and the latter is more convenient. Moreover, among methods of fine-tuning LLMs, only models focused on NLG evaluation scenarios, such as Themis, outperform prompting-based methods, including that using GPT-4. Other studies either use relatively outdated foundation LLMs or lack training on specific evaluation aspects like those in SummEval, leading to relatively weaker performance. Furthermore, Human-LLM Collaborative Evaluation enhances the LLM evaluation by incorporating checklists elaborated with human expert insights and LLM knowledge, resulting in the strongest performance.

LEFT

Cost. LLM-derived Metrics \approx Prompting open-source LLMs < Fine-tuning LLMs \approx Prompting proprietary LLMs < Human-LLM Collaborative Evaluation. When using the same open-source LLM, the inference costs of LLM-derived metrics, prompting LLM, and fine-tuning LLM methods are the same, while fine-tuning LLM incurs additional training costs. When prompting proprietary LLMs, the cost is high and mainly concentrated in API calls during evaluation, making it difficult to directly compare with the training cost required for fine-tuning LLM. Moreover, human-LLM collaborative evaluation requires the involvement of human experts for each task, making it the most expensive approach.

LEFT

6.3 Future Directions

LEFT

Unified benchmarks for LLM-based NLG evaluation approaches. As mentioned above, each of the studies that fine-tuned LLMs to construct specialized evaluation models uses different settings and data during testing, making them incomparable. In the research on prompting LLMs for NLG evaluation, there are some publicly available human judgments on the same NLG task, such as SummEval for summarization. However, the existing human judgments have many problems. Firstly, most of the existing data only involve one type of NLG task and a single human evaluation method (e.g., scoring), making it difficult to evaluate LLMs' performance on different tasks, as well as using different evaluation methods on the same task. Secondly, many of the texts in these human judgments are generated by outdated models (such as Pointer Network) and do not include texts generated by more advanced LLMs. Lastly, many human evaluation datasets are too small in scale. There is an urgent need for large-scale, high-quality human evaluation data covering various NLG tasks and evaluation methods as a benchmark.

LEFT

NLG evaluation for low-resource languages and new task scenarios. Almost all existing research focuses on English data. However, it is doubtful whether LLMs have similar levels of NLG evaluation capability for texts in other languages, especially low-resource languages. As (Zhang et al. 2023a) points out, we should be more cautious about using LLMs to evaluate texts in non-Latin languages. We believe that the lack of evaluation capability of LLM-based evaluators on low-resource languages may be due to the insufficient presence of these languages in the pretraining corpus. Therefore, further fine-tuning on certain low-resource languages may be a potential strategy to address this issue, and Hada et al. (2024) have already shown promising preliminary results. Additionally, existing research mainly focuses on more traditional NLG tasks such

LEFT

LEFT

Survey Article

LEFT

Gao, Hu, Yin, Ruan, Pu, and Wan 20

LEFT

as translation, summarization, and dialogue. However, there are many new scenarios in reality with different requirements and evaluation criteria. For example, using LLMs to automatically evaluate scientific reviews could be valuable in identifying and flagging content that is unfaithful or unclear, alerting reviewers to potential issues. Research on low-resource languages and new task scenarios will provide a more comprehensive understanding of LLMs' evaluation capabilities.

Diverse forms of human-LLM collaborative NLG evaluation. According to the literature reviewed above, there is little research on collaborative evaluation between humans and LLMs. Neither humans nor LLMs are perfect, and each has its strengths. Since the ultimate goal of NLG research is to evaluate text quality more accurately and efficiently, we believe that collaboration between humans and LLMs can achieve better results than pure human evaluation or automatic evaluation. In the collaboration between humans and LLMs, technologies in the field of human-computer interaction may bring new implementation methods to the collaboration. In addition, what roles humans and LLMs should play in the evaluation and how they can better complement each other are still worth researching.

LEFT

LEFT

Survey Article

LEFT

Gao, Hu, Yin, Ruan, Pu, and Wan 21

LEFT

LEFT
Acknowledgments

This work was supported by Beijing Science and Technology Program (Z231100007423011) and Key Laboratory of Science, Technology and Standard in Press Industry (Key Laboratory of Intelligent Press Media Technology). We appreciate the anonymous reviewers for their helpful comments. Xiaojun Wan is the corresponding author.

LEFT

LEFT

LEFT
References

Amodei, Dario, Chris Olah, Jacob Steinhardt, Paul F. Christiano, John Schulman, and Dan Mané. 2016. Concrete problems in AI safety. *CoRR*, abs/1606.06565.

Anil, Rohan, Andrew M. Dai, Orhan Firat, Melvin Johnson, Dmitry Lepikhin, Alexandre Passos, Siamak Shakeri, Emanuel Taropa, Paige Bailey, Zhifeng Chen, Eric Chu, Jonathan H. Clark, Laurent El Shafey, Yanping Huang, Kathy Meier-Hellstern, Gaurav Mishra, Erica Moreira, Mark Omernick, Kevin Robinson, Sebastian Ruder, Yi Tay, Kefan Xiao, Yuanzhong Xu, Yujing Zhang, Gustavo Hernandez Abrego, Junwhan Ahn, Jacob Austin, Paul Barham, Jan Botha, James Bradbury, Siddhartha Brahma, Kevin Brooks, Michele Catasta, Yong Cheng, Colin Cherry, Christopher A. Choquette-Choo, Aakanksha Chowdhery, Clément Crepy, Shachi Dave, Mostafa Dehghani, Sunipa Dev, Jacob Devlin, Mark Díaz, Nan Du, Ethan Dyer, Vlad Feinberg, Fangxiaoyu Feng, Vlad Fienber, Markus Freitag, Xavier Garcia, Sebastian Gehrmann, Lucas Gonzalez, Guy Gur-Ari, Steven Hand, Hadi Hashemi, Le Hou, Joshua Howland, Andrea Hu, Jeffrey Hui, Jeremy Hurwitz, Michael Isard, Abe Ittycheriah, Matthew Jagielski, Wenhao Jia, Kathleen Kenealy, Maxim Krikun, Sneha Kudugunta, Chang Lan, Katherine Lee, Benjamin Lee, Eric Li, Music Li, Wei Li, YaGuang Li, Jian Li, Hyeontaek Lim, Hanzhao Lin, Zhongtao Liu, Frederick Liu, Marcello Maggioni, Aroma Mahendru, Joshua Maynez, Vedant Misra, Maysam Moussaleem, Zachary Nado, John Nham, Eric Ni, Andrew Nystrom, Alicia Parrish, Marie Pellat, Martin Polacek, Alex Polozov, Reiner Pope, Siyuan Qiao, Emily Reif, Bryan Richter, Parker Riley, Alex Castro Ros, Aurko Roy, Brennan Saeta, Rajkumar Samuel, Renee Shelby, Ambrose Slone, Daniel Smilkov, David R. So, Daniel Sohn, Simon Tokumine, Dasha Valter, Vijay Vasudevan, Kiran Vodrahalli, Xuezhi Wang, Pidong Wang, Zirui Wang, Tao Wang, John Wieting, Yuhuai Wu, Kelvin Xu, Yunhan Xu, Linting Xue, Pengcheng Yin, Jiahui Yu, Qiao Zhang, Steven Zheng, Ce Zheng, Weikang Zhou,

Denny Zhou, Slav Petrov, and Yonghui Wu. 2023. Palm 2 technical report. *Computing Research Repository*, arxiv:2305.10403.

Ashktorab, Zahra, Michael Desmond, Qian Pan, James M. Johnson, Martin Santillan Cooper, Elizabeth M. Daly, Rahul Nair, Tejaswini Pedapati, Swapnaja Achintalwar, and Werner Geyer. 2024. Aligning human and llm judgments: Insights from evalassist on task-specific evaluations and ai-assisted assessment strategy preferences.

Bai, Yushi, Jiahao Ying, Yixin Cao, Xin Lv, Yuze He, Xiaozhi Wang, Jifan Yu, Kaisheng Zeng, Yijia Xiao, Haozhe Lyu, Jiayin Zhang, Juanzi Li, and Lei Hou. 2023. Benchmarking foundation models with language-model-as-an-examiner. *CoRR*, abs/2306.04181.

Bavaresco, Anna, Raffaella Bernardi, Leonardo Bertolazzi, Desmond Elliott, Raquel Fernández, Albert Gatt, Esam Ghaleb, Mario Giulianelli, Michael Hanna, Alexander Koller, André F. T. Martins, Philipp Mondorf, Vera Neplenbroek, Sandro Pezzelle, Barbara Plank, David Schlangen, Alessandro Suglia, Aditya K. Surikuchi, Ece Takmaz, and Alberto Testoni. 2024. Llm instead of human judges? A large scale empirical study across 20 NLP evaluation tasks. *CoRR*, abs/2406.18403.

Blodgett, Su Lin, Solon Barocas, Hal Daumé III, and Hanna M. Wallach. 2020. Language (technology) is power: A critical survey of "bias" in NLP. In *ACL*.

Brown, Tom B., Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *NeurIPS*.

Cao, Maosong, Alexander Lam, Haodong Duan, Hongwei Liu, Songyang Zhang,

LEFT

LEFT

Survey Article

LEFT
 LEFT
 and Kai Chen. 2024. Compassjudge-1:
 All-in-one judge model helps model
 evaluation and evolution. *arXiv preprint*
arXiv:2410.16256.
 LEFT
 LEFT
 Cegin, Jan, Jakub Simko, and Peter
 Brusilovsky. 2023. ChatGPT to replace
 crowdsourcing of paraphrases for intent
 classification: Higher diversity and
 comparable model robustness. In
 LEFT
 LEFT
 LEFT
Proceedings of the 2023 Conference on
Empirical Methods in Natural Language
Processing, pages 1889–1905, Association
 for Computational Linguistics, Singapore.
 LEFT
 Celikyilmaz, Asli, Elizabeth Clark, and
 Jianfeng Gao. 2020. Evaluation of text
 generation: A survey. *CoRR*,
 abs/2006.14799.
 LEFT
 LEFT
 Chan, Chi-Min, Weize Chen, Yusheng Su,
 Jianxuan Yu, Wei Xue, Shanghang Zhang,
 Jie Fu, and Zhiyuan Liu. 2023. Chateval:
 Towards better llm-based evaluators
 through multi-agent debate. *CoRR*,
 abs/2308.07201.
 LEFT
 LEFT
 Chang, Yapei, Kyle Lo, Tanya Goyal, and
 Mohit Iyyer. 2023. Boookscore: A
 systematic exploration of book-length
 summarization in the era of llms. *CoRR*,
 abs/2310.00785.
 LEFT
 LEFT
 Chang, Yupeng, 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):39:1–39:45.
 LEFT
 LEFT
 Chen, Mark, Jerry Tworek, Heewoo Jun,
 Qiming Yuan, Henrique Pondé
 de Oliveira Pinto, Jared Kaplan, Harrison
 Edwards, Yuri Burda, Nicholas Joseph,
 LEFT
 Greg Brockman, Alex Ray, Raul Puri,
 Gretchen Krueger, Michael Petrov, Heidy
 Khlaaf, Girish Sastry, Pamela Mishkin,
 Brooke Chan, Scott Gray, Nick Ryder,
 Mikhail Pavlov, Alethea Power, Lukasz
 Kaiser, Mohammad Bavarian, Clemens
 Winter, Philippe Tillet, Felipe Petroski
 Such, Dave Cummings, Matthias Plappert,
 Fotios Chantzis, Elizabeth Barnes, Ariel
 Herbert-Voss, William Hebgen Guss, Alex
 Nichol, Alex Paino, Nikolas Tezak, Jie
 Tang, Igor Babuschkin, Suchir Balaji,
 Shantanu Jain, William Saunders,
 Christopher Hesse, Andrew N. Carr, Jan
 Leike, Joshua Achiam, Vedant Misra, Evan
 Morikawa, Alec Radford, Matthew
 Knight, Miles Brundage, Mira Murati,
 Katie Mayer, Peter Welinder, Bob McGrew,
 Dario Amodei, Sam McCandlish, Ilya

LEFT

Gao, Hu, Yin, Ruan, Pu, and Wan 22

LEFT
 LEFT
 Sutskever, and Wojciech Zaremba. 2021.
 Evaluating large language models trained
 on code. *CoRR*, abs/2107.03374.
 LEFT
 LEFT
 Chiang, David Cheng-Han and Hung-yi Lee.
 2023a. Can large language models be an
 alternative to human evaluations? In *ACL*
 LEFT
 LEFT
 Chiang, David Cheng-Han and Hung-yi Lee.
 2023b. A closer look into using large
 language models for automatic evaluation.
 In *EMNLP (Findings)*.
 LEFT
 LEFT
 Chu, SeongYeub, JongWoo Kim, and
 MunYong Yi. 2025. Think together and
 work better: Combining humans' and
 llms' think-aloud outcomes for effective
 text evaluation.
 LEFT
 LEFT
 Chung, Hyung Won, Le Hou, Shayne
 Longpre, Barret Zoph, Yi Tay, William
 Fedus, Eric Li, Xuezhi Wang, Mostafa
 Dehghani, Siddhartha Brahma, Albert
 Webson, Shixiang Shane Gu, Zhuyun Dai,
 Mirac Suzgun, Xinyun Chen, Aakanksha
 Chowdhery, Sharan Narang, Gaurav
 Mishra, Adams Yu, Vincent Y. Zhao,
 Yanping Huang, Andrew M. Dai,
 Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff
 Dean, Jacob Devlin, Adam Roberts, Denny
 Zhou, Quoc V. Le, and Jason Wei. 2022.
 Scaling instruction-finetuned language
 models. *CoRR*, abs/2210.11416.
 LEFT
 LEFT
 Cohen, Roi, May Hamri, Mor Geva, and
 Amir Globerson. 2023. LM vs LM:
 detecting factual errors via cross
 examination. In *EMNLP*.
 LEFT
 LEFT
 Deriu, Jan, Alvaro Rodrigo, Arantxa Otegi,
 Guillermo Echegoyen, Sophie Rosset,
 Eneko Agirre, and Mark Cieliebak. 2021.
 Survey on evaluation methods for
 dialogue systems. *Artif. Intell. Rev.*,
 54(1):755–810.
 LEFT
 LEFT
 Durmus, Esin, He He, and Mona T. Diab.
 2020. FEQA: A question answering
 evaluation framework for faithfulness
 assessment in abstractive summarization.
 In *ACL*.
 LEFT
 LEFT
 Eddine, Moussa Kamal, Guokan Shang,
 Antoine J.-P. Tixier, and Michalis
 Vazirgiannis. 2022. Frugalscore: Learning
 cheaper, lighter and faster evaluation
 metrics for automatic text generation. In
ACL (1).
 LEFT
 LEFT
 Es, Shahul, Jithin James, Luis Espinosa Anke,
 and Steven Schockaert. 2023. RAGAS:
 automated evaluation of retrieval
 augmented generation. *CoRR*,
 abs/2309.15217.
 LEFT
 LEFT
 Fabbri, Alexander R, Wojciech Kryscinski,
 Bryan McCann, Caiming Xiong, Richard
 Socher, and Dragomir Radev. 2021.

LEFT

LEFT

Survey Article

LEFT

practice guidelines. *Comput. Speech Lang.*, 67:101151.

Leiter, Christoph and Steffen Eger. 2024. Prexme! large scale prompt exploration of open source llms for machine translation and summarization evaluation. In *EMNLP*, pages 11481–11506, Association for Computational Linguistics.

Leiter, Christoph, Juri Opitz, Daniel Deutsch, Yang Gao, Rotem Dror, and Steffen Eger. 2023. The eval4nlp 2023 shared task on prompting large language models as explainable metrics. *CoRR*, abs/2310.19792.

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

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

Li, Ruosen, Teerth Patel, and Xinya Du. 2023. PRD: peer rank and discussion improve large language model based evaluations. *CoRR*, abs/2307.02762.

Li, Yujia, David Choi, Junyoung Chung, Nate Kushman, Julian Schrittwieser, Remi Leblond, Tom Eccles, James Keeling, Felix Gimeno, Agustin Dal Lago, Thomas Hubert, Peter Choy, Cyprien de Masson d’Autume, Igor Babuschkin, Xinyun Chen, Po-Sen Huang, Johannes Welbl, Sven Gowal, Alexey Cherepanov, James Molloy, Daniel J. Mankowitz, Esme Sutherland Robson, Pushmeet Kohli, Nando de Freitas, Koray Kavukcuoglu, and Oriol Vinyals. 2022. Competition-level code generation with alphacode. *Science*, 378(6624):1092–1097.

Li, Zongjie, Chaozheng Wang, Pingchuan Ma, Daoyuan Wu, Shuai Wang, Cuiyun Gao, and Yang Liu. 2023c. Split and merge: Aligning position biases in large language model based evaluators. *CoRR*, abs/2310.01432.

Lin, Yen-Ting and Yun-Nung Chen. 2023. Llm-eval: Unified multi-dimensional automatic evaluation for open-domain conversations with large language models. In *NLP4ConvAI 2023*.

Liu, Yang, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. 2023a. G-eval: NLG evaluation using gpt-4 with better human alignment. In *EMNLP*.

Liu, Yixin, Alexander R. Fabbri, Jiawen Chen, Yilun Zhao, Simeng Han, Shafiq

LEFT

Gao, Hu, Yin, Ruan, Pu, and Wan 25

LEFT

Joty, Pengfei Liu, Dragomir Radev, Chien-Sheng Wu, and Arman Cohan. 2023b. Benchmarking generation and evaluation capabilities of large language models for instruction controllable summarization. *CoRR*, abs/2311.09184.

Liu, Yixin, Alexander R. Fabbri, Pengfei Liu, Dragomir Radev, and Arman Cohan. 2023c. On learning to summarize with large language models as references. *CoRR*, abs/2305.14239.

Liu, Yongkang, Shi Feng, Daling Wang, Yifei Zhang, and Hinrich Schütze. 2023d. Evaluate what you can’t evaluate: Unassessable generated responses quality. *CoRR*, abs/2305.14658.

Liu, Yuxuan, Tianchi Yang, Shaohan Huang, Zihan Zhang, Haizhen Huang, Furu Wei, Weiwei Deng, Feng Sun, and Qi Zhang. 2023e. Calibrating llm-based evaluator. *CoRR*, abs/2309.13308.

Liusie, Adian, Potsawee Manakul, and Mark J. F. Gales. 2023. Llm comparative assessment: Zero-shot nlq evaluation through pairwise comparisons using large language models. *Computing Research Repository*, arxiv:2307.07889.

Lu, Qingyu, Baopu Qiu, Liang Ding, Liping Xie, and Dacheng Tao. 2023. Error analysis prompting enables human-like translation evaluation in large language models: A case study on chatgpt. *CoRR*, abs/2303.13809.

Luo, Zheheng, Qianqian Xie, and Sophia Ananiadou. 2023. Chatgpt as a factual inconsistency evaluator for abstractive text summarization. *CoRR*, abs/2303.15621.

Manakul, Potsawee, Adian Liusie, and Mark J. F. Gales. 2023. Selfcheckgpt: Zero-resource black-box hallucination detection for generative large language models. In *EMNLP*.

Mendonça, John, Patrícia Pereira, João Paulo Carvalho, Alon Lavie, and Isabel Trancoso. 2023. Simple LLM prompting is state-of-the-art for robust and multilingual dialogue evaluation. In *Proceedings of The Eleventh Dialog System Technology Challenge*.

Menick, Jacob, Maja Trebacz, Vladimir Mikulik, John Aslanides, H. Francis Song, Martin J. Chadwick, Mia Glaese, Susannah Young, Lucy Campbell-Gillingham, Geoffrey Irving, and Nat McAleese. 2022. Teaching language models to support answers with verified quotes. *CoRR*, abs/2203.11147.

Murugadoss, Bhuvanashree, Christian Poeltz, Ian Drosos, Vu Le, Nick McKenna,

LEFT

LEFT

Survey Article

LEFT

Carina Suzana Negreanu, Chris Parnin, and Advait Sarkar. 2024. Evaluating the evaluator: Measuring llms' adherence to task evaluation instructions. *arXiv preprint arXiv:2408.08781*.

Naismith, Ben, Phoebe Mulcaire, and Jill Burstein. 2023. Automated evaluation of written discourse coherence using GPT-4. In *BEA@ACL*.

Nakano, Reiichiro, Jacob Hilton, Suchir Balaji, Jeff Wu, Long Ouyang, Christina Kim, Christopher Hesse, Shantanu Jain, Vineet Kosaraju, William Saunders, Xu Jiang, Karl Cobbe, Tyna Eloundou, Gretchen Krueger, Kevin Button, Matthew Knight, Benjamin Chess, and John Schulman. 2021. Webgpt: Browser-assisted question-answering with human feedback. *CoRR*, abs/2112.09332.

Nimah, Iftitahu, Meng Fang, Vlado Menkovski, and Mykola Pechenizkiy. 2023. NLG evaluation metrics beyond correlation analysis: An empirical metric preference checklist. In *ACL (1)*, pages 1240–1266, Association for Computational Linguistics.

Ostyakova, Lidiia, Veronika Smilga, Kseniia Petukhova, Maria Molchanova, and Daniel Kornev. 2023. ChatGPT vs. crowdsourcing vs. experts: Annotating open-domain conversations with speech functions. In *Proceedings of the 24th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 242–254, Association for Computational Linguistics, Prague, Czechia.

Ouyang, Long, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F. Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. In *NeurIPS*.

Papineni, Kishore, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *ACL*.

Pozdniakov, Stanislav, Jonathan Brazil, Solmaz Abdi, Aneesha Bakharia, Shazia Sadiq, Dragan Gašević, Paul Denny, and Hassan Khosravi. 2024. Large language models meet user interfaces: The case of provisioning feedback. *Computers and Education: Artificial Intelligence*, 7:100289.

Rastogi, Charvi, Marco Túlio Ribeiro, Nicholas King, Harsha Nori, and Saleema

LEFT

Gao, Hu, Yin, Ruan, Pu, and Wan 26

LEFT

Amershi. 2023. Supporting human-ai collaboration in auditing llms with llms. In *AIES*.

Ribeiro, Marco Túlio and Scott M. Lundberg. 2022. Adaptive testing and debugging of NLP models. In *ACL (1)*.

Saha, Swarnadeep, Omer Levy, Asli Celikyilmaz, Mohit Bansal, Jason Weston, and Xian Li. 2023. Branch-solve-merge improves large language model evaluation and generation. *CoRR*, abs/2310.15123.

Sai, Ananya B, Tanay Dixit, Dev Yashpal Sheth, Sreyas Mohan, and Mitesh M Khapra. 2021. Perturbation checklists for evaluating nlg evaluation metrics. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7219–7234.

Sai, Ananya B., Akash Kumar Mohankumar, and Mitesh M. Khapra. 2023. A survey of evaluation metrics used for NLG systems. *ACM Comput. Surv.*, 55(2):26:1–26:39.

Saunders, William, Catherine Yeh, Jeff Wu, Steven Bills, Long Ouyang, Jonathan Ward, and Jan Leike. 2022. Self-critiquing models for assisting human evaluators. *CoRR*, abs/2206.05802.

Shankar, Shreya, J.D. Zamfirescu-Pereira, Bjoern Hartmann, Aditya Parameswaran, and Ian Arawjo. 2024. Who validates the validators? aligning llm-assisted evaluation of llm outputs with human preferences. In *Proceedings of the 37th Annual ACM Symposium on User Interface Software and Technology*, UIST '24, Association for Computing Machinery, New York, NY, USA.

Shen, Chenhui, Liying Cheng, Xuan-Phi Nguyen, Yang You, and Lidong Bing. 2023. Large language models are not yet human-level evaluators for abstractive summarization. In *EMNLP (Findings)*.

Sheng, Shuqian, Yi Xu, Tianhang Zhang, Zanwei Shen, Luoyi Fu, Jiaxin Ding, Lei Zhou, Xiaoying Gan, Xinbing Wang, and Chenghu Zhou. 2024. Repeval: Effective text evaluation with LLM representation. In *EMNLP*, pages 7019–7033, Association for Computational Linguistics.

Shu, Lei, Nevan Wichers, Liangchen Luo, Yun Zhu, Yinxiao Liu, Jindong Chen, and Lei Meng. 2023. Fusion-eval: Integrating evaluators with llms. *CoRR*, abs/2311.09204.

Stent, Amanda, Matthew Marge, and Mohit Singhai. 2005. Evaluating evaluation methods for generation in the presence of variation. In *CICLing*, volume 3406 of *Lecture Notes in Computer Science*, pages

LEFT

LEFT

Survey Article

LEFT

LEFT
841–351, Springer

Sun, Tianxiang, Junliang He, Xipeng Qiu, and Xuanjing Huang. 2022. Bertscore is unfair: On social bias in language model-based metrics for text generation. In *EMNLP*.

Törnberg, Petter. 2023. Chatgpt-4 outperforms experts and crowd workers in annotating political twitter messages with zero-shot learning. *CoRR*, abs/2304.06588.

Touvron, Hugo, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian

Canton-Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar

Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao,

Xavier Martinet, Iodori Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi

Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh

Lang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang,

Aurélien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023.

Llama 2: Open foundation and fine-tuned chat models. *CoRR*, abs/2307.09288.

Varshney, Neeraj, Wenlin Yao, Hongming Zhang, Jianshu Chen, and Dong Yu. 2023. A stitch in time saves nine: Detecting and mitigating hallucinations of llms by validating low-confidence generation. *CoRR*, abs/2307.03987.

Wang, Jiaan, Yunlong Liang, Fandong Meng, Haoxiang Shi, Zhixu Li, Jinan Xu, Jianfeng Qu, and Jie Zhou. 2023a. Is ChatGPT a good NLG evaluator? a preliminary study. In *Proceedings of the 4th New Frontiers in Summarization Workshop*.

Wang, Peiyi, Lei Li, Liang Chen, Dawei Zhu, Binghuai Lin, Yunbo Cao, Qi Liu, Tianyu Liu, and Zhifang Sui. 2023b. Large language models are not fair evaluators. *CoRR*, abs/2305.17926.

LEFT

Gao, Hu, Yin, Ruan, Pu, and Wan 27

LEFT

Wang, Tianlu, Ilia Kulikov, Olga Golovneva, Ping Yu, Weizhe Yuan, Jane Dwivedi-Yu, Richard Yuanzhe Pang, Maryam

Fazel-Zarandi, Jason Weston, and Xian Li. 2024. Self-taught evaluators. *arXiv preprint arXiv:2408.02666*.

Wang, Tianlu, Ping Yu, Xiaoqing Ellen Tan, Sean O'Brien, Ramakanth Pasunuru, Jane Dwivedi-Yu, Olga Golovneva, Luke

Zettlemoyer, Maryam Fazel-Zarandi, and Asli Celikyilmaz. 2023c. Shepherd: A critic for language model generation. *CoRR*, abs/2308.04592.

Wang, Yaqing, Jiepu Jiang, Mingyang Zhang, Cheng Li, Yi Liang, Qiaozhu Mei, and Michael Bendersky. 2023d. Automated evaluation of personalized text generation using large language models. *CoRR*, abs/2310.11593.

Wang, Yidong, Zhuohao Yu, Zhengran Zeng, Linyi Yang, Cunxiang Wang, Hao Chen, Chaoya Jiang, Rui Xie, Jindong Wang, Xing Xie, Wei Ye, Shikun Zhang, and Yue Zhang. 2023e. Pandalm: An automatic evaluation benchmark for LLM instruction tuning optimization. *CoRR*, abs/2306.05087.

Wang, Zifan, Kotaro Funakoshi, and Manabu Okumura. 2023. Automatic answerability evaluation for question generation. *CoRR*, abs/2309.12546.

Wiseman, Sam, Stuart M. Shieber, and Alexander M. Rush. 2017. Challenges in data-to-document generation. In *EMNLP*, pages 2253–2263, Association for Computational Linguistics.

Wu, Minghao and Alham Fikri Aji. 2023. Style over substance: Evaluation biases for large language models. *CoRR*, abs/2307.03025.

Wu, Ning, Ming Gong, Linjun Shou, Shining Liang, and Daxin Jiang. 2023. Large language models are diverse role-players for summarization evaluation. In *NLPCC*.

Xiao, Ziang, Susu Zhang, Vivian Lai, and Q. Vera Liao. 2023. Evaluating evaluation metrics: A framework for analyzing NLG evaluation metrics using measurement theory. In *EMNLP*, pages 10967–10982, Association for Computational

Linguistics.

Xie, Zhuohan, Miao Li, Trevor Cohn, and Jey Han Lau. 2023. Deltascore: Fine-grained story evaluation with perturbations. In *EMNLP (Findings)*.

Xu, Wenda, Danqing Wang, Liangming Pan, Zhenqiao Song, Markus Freitag, William Wang, and Lei Li. 2023.

LEFT

LEFT

Survey Article

LEFT

INSTRUCTSCORE: towards explainable text generation evaluation with automatic feedback. In *EMNLP*.
Ye, Seonghyeon, Doyoung Kim, Sungdong Kim, Hyeonbin Hwang, Seungone Kim, Yongrae Jo, James Thorne, Juho Kim, and Minjoon Seo. 2023a. FLASK: fine-grained language model evaluation based on alignment skill sets. *CoRR*, abs/2307.10928.
Ye, Xi, Srinivasan Iyer, Asli Celikyilmaz, Veselin Stoyanov, Greg Durrett, and Ramakanth Pasunuru. 2023b. Complementary explanations for effective in-context learning. In *ACL (Findings)*.
Yin, Kayo and Graham Neubig. 2022. Interpreting language models with contrastive explanations. In *EMNLP*.
Yuan, Peiwen, Shaoxiong Feng, Yiwei Li, Xinglin Wang, Boyuan Pan, Heda Wang, and Kan Li. 2024. Batcheval: Towards human-like text evaluation. *CoRR*, abs/2401.00437.
Yuan, Weizhe, Graham Neubig, and Pengfei Liu. 2021. Bartscore: Evaluating generated text as text generation. In *NeurIPS*.
Zhang, Chen, Luis Fernando D'Haro, Yiming Chen, Malu Zhang, and Haizhou Li. 2023a. A comprehensive analysis of the effectiveness of large language models as automatic dialogue evaluators. *CoRR*, abs/2312.15407.
Zhang, Susan, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona T. Diab, Xian Li, Xi Victoria Lin, Todor Mihaylov, Myle Ott, Sam Shleifer, Kurt Shuster, Daniel Simig, Punit Singh Koura, Anjali Sridhar, Tianlu Wang, and Luke Zettlemoyer. 2022. OPT: open pre-trained transformer language models. *CoRR*, abs/2205.01068.
Zhang, Xinghua, Bowen Yu, Haiyang Yu, Yangyu Lv, Tingwen Liu, Fei Huang, Hongbo Xu, and Yongbin Li. 2023b. Wider and deeper LLM networks are fairer LLM evaluators. *CoRR*, abs/2308.01862.
Zhang, Yangjun, Pengjie Ren, and Maarten de Rijke. 2021. A human-machine collaborative framework for evaluating malevolence in dialogues. In *ACL/IJCNLP*.
Zhang, Ying, Stephan Vogel, and Alex Waibel. 2004. Interpreting BLEU/NIST scores: How much improvement do we need to have a better system? In *LREC*, European Language Resources Association.
Zheng, Lianmin, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu,

LEFT

Gao, Hu, Yin, Ruan, Pu, and Wan 28

LEFT

Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. *CoRR*, abs/2306.05685.
Zhou, Kaitlyn, Su Lin Blodgett, Adam Trischler, Hal Daumé III, Kaheer Suleman, and Alexandra Olteanu. 2022. Deconstructing NLG evaluation: Evaluation practices, assumptions, and their implications. In *NAACL-HLT*, pages 314–324, Association for Computational Linguistics.
Zhou, Yongxin, Fabien Ringeval, and François Portet. 2023. A survey of evaluation methods of generated medical textual reports. In *ClinicalNLP@ACL*, pages 447–459, Association for Computational Linguistics.
Zhu, Lianghui, Xinggang Wang, and Xinlong Wang. 2023. Judgelm: Fine-tuned large language models are scalable judges. *CoRR*, abs/2310.17631.
Zhuang, Ziyu, 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. *CoRR*, abs/2308.07902.
Zouhar, Vilem, Tom Kocmi, and Mrinmaya Sachan. 2025. Ai-assisted human evaluation of machine translation.

LEFT