DHP Benchmark: Measuring Discernment Ability of LLM-as-a-Judge

Anonymous Author(s)
Affiliation
Address
email

Abstract

Large Language Models (LLMs) are increasingly serving as evaluators in Natural Language Generation (NLG) tasks; this is often referred to as "LLM-as-a-judge" paradigm. However, the capabilities of LLMs in evaluating NLG quality remain underexplored. Current studies depend on human assessments and simple metrics that fail to capture the discernment of LLMs across diverse NLG tasks. To address this gap, we propose the Discernment of Hierarchical Perturbation (DHP) benchmarking framework, which provides quantitative discernment scores for LLMs. DHP systematically degrades reference texts at character (typos, deletions), word (grammatical errors, entity substitutions), and sentence levels (reordering, factual inconsistencies), then uses Wilcoxon Signed-Rank Tests to measure whether LLMs assign lower scores to perturbed texts. We benchmark 19 LLMs from 8 families (GPT, Llama, Qwen, Vicuna, Mistral) across 6 datasets spanning summarization, story completion, question answering, and translation tasks. Our results provide critical insight into their strengths and limitations as NLG evaluators.

1 Introduction

1.1 Background and Challenges

Large Language Models (LLMs) are increasingly used as evaluators in Natural Language Generation (NLG) [5, 20, 19, 34]. As "LLM-as-a-judge" scales to summarization, story completion, question answering, and translation [19, 31, 7], a central question is whether these models reliably score text quality across metrics. Two obstacles persist: (i) unbiased measurement is hard—human alignment is confounded by annotator and model response styles [28]; and (ii) multiple, correlated metrics (e.g., coherence, consistency, fluency, relevance) complicate evaluation [10, 11, 14]. Figure 1 illustrates these issues; we show response-style analysis in Appendix B.

Key community questions include:

- How do we assess reliability beyond human alignment? Relative discernment under controlled degradations complements absolute correlations.
- Which metrics matter by task, and do models truly optimize them? Expert-weighted aggregation emphasizes the most impacted criteria per perturbation.
- Are judgments stable across prompts, seeds, and context length? Averaging runs and paired tests improve robustness to these factors.
- Do evaluators generalize across tasks and perturbations? Worst-case discernment highlights limits in harder domains (e.g., long scientific, multilingual).

Submitted to 39th Conference on Neural Information Processing Systems (NeurIPS 2025). Do not distribute.

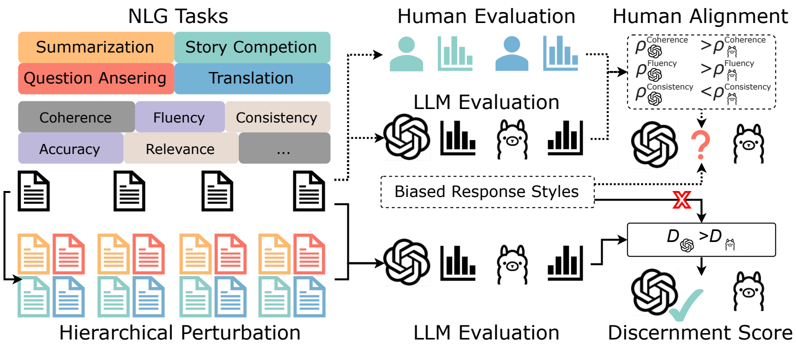

Figure 1: Challenges in Assessing LLMs as NLG Evaluators: Biased Response Styles and Multiple Evaluation Metrics. Our DHP Framework employs hierarchical perturbation and statistical tests to address these challenges, offering quantitative discernment scores for effective comparison.

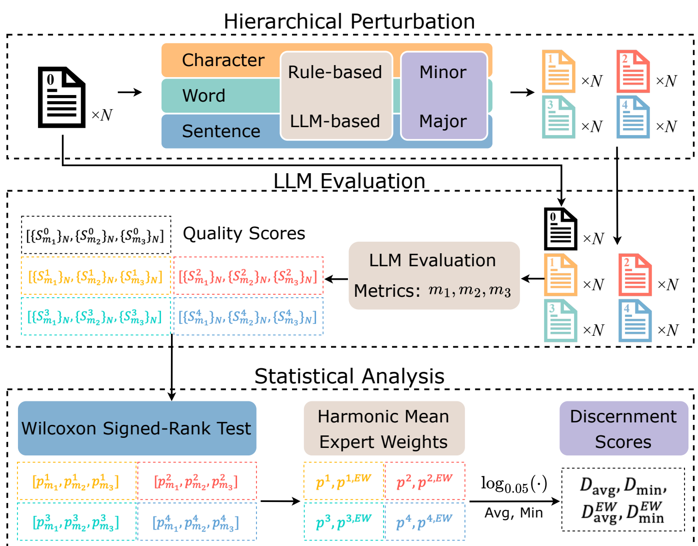

Figure 2: The DHP framework: (1) Hierarchical Perturbation, (2) LLM Evaluation, and (3) Statistical Analysis. Example shown with $P = 4$ perturbation types and $M = 3$ metrics.

1.2 DHP Overview and Findings

We propose DHP (Discernment of Hierarchical Perturbation), which perturbs references at multiple levels and tests whether an evaluator assigns lower scores to perturbed texts relative to originals. Using a one-sided Wilcoxon Signed-Rank Test [33], harmonic-mean p-values, and expert-weighted combinations, we convert evidence into a positive **Discernment Score** that is robust to absolute scoring styles and emphasizes relative judgments. We benchmark eight model families across six datasets; most show significant average discernment, with stronger models more stable and harder datasets exposing weaknesses.

2 DHP Benchmarking Framework

We propose DHP (Discernment of Hierarchical Perturbation) to measure whether an LLM can reliably penalize degraded text. Given a dataset with N items and M evaluation metrics, we create P perturbed variants per item, independently ask an LLM to score originals and each perturbation using standardized prompts, and then quantify discernment via nonparametric tests combined across metrics. This design emphasizes relative judgments, mitigating response-style bias [24].

The overall framework is shown in Figure 2. First, for a specific NLG task, we employ a hierarchical perturbation pipeline to transform high-quality reference data into various forms of lower-quality data. Subsequently, an LLM evaluates both the original and perturbed texts, respectively, using predefined metrics, generating several sets of rating scores. We then conduct a statistical analysis of these

Table 1: The quality metrics and perturbation methods for the four NLG tasks. **C**: Character Level. **W**: Word Level. **S**: Sentence Level. **(R)**: Rule-based Perturbation. **(L)**: LLM-based Perturbation. **(M)**: Major and Minor Perturbations for each method. Details in Table 2.

Task	Metrics	Perturbations
Summarization	Coherence Consistency Fluency Relevance	**C (M)**: Random Deletions **(R)**, Random Typos **(R)** **W (M)**: Fictional Named Entities **(L)**, Grammatical Errors **(L)** **S (M)**: Reordering **(R)**, Rewriting and Insertion **(L)**
Story Completion	Coherence Consistency Fluency	**C**: Random Deletions **(R)**, Random Typos **(R)** **W**: Fictional Named Entities **(L)**, Grammatical Errors **(L)** **S**: Random Ending Sentence **(R)**, Wrong Ending Sentence **(R)**
Question Answering	Answer Quality	**C (M)**: Random Deletions **(R)**, Random Typos **(R)** **W (M)**: Fictional Named Entities **(L)**, Grammatical Errors **(L)** **S**: Random Answer **(R)**
Translation	Accuracy Fluency	**C (M)**: Random Deletions **(R)**, Random Typos **(R)** **W (M)**: Random Deletions **(R)**, Fictional Named Entities **(L)**, Grammatical Errors **(L)**

scores. For each pair of scores, original and perturbed, we apply the Wilcoxon Signed-Rank Test to determine the differences in their distributions, achieving this with a confidence level expressed as a p-value. This test specifically assesses differences in pairwise scores without focusing on absolute values, thereby minimizing the impact of models' response styles. Following this, we combine the p-values from different metrics, incorporating Expert Weights (EW) to tailor the aggregated p-values to the specific metrics of the corresponding perturbation methods. These combined p-values are then transformed into discernment scores, which serve as a direct measure for assessing and comparing the NLG evaluation capabilities of LLMs for this particular task.

2.1 Step 1: Hierarchical Perturbation

We generate degraded texts by perturbing references at character, word, and sentence levels, using rule-based and LLM-based methods with minor/major severities (Figure 2, Table 1). Each perturbation targets specific quality issues (e.g., typos, factual inconsistency, discourse breaks). A capable evaluator should assign lower scores to perturbed versions than to the originals.

2.2 Step 2: LLM evaluation

Following G-Eval [20] with Auto-CoT [36], we prompt LLMs to score originals and each perturbed set independently for each metric $m_j, j \in \{1, \dots, M\}$. Over N items we obtain metric-wise score sets for originals and for each perturbation type $i \in \{1, \dots, P\}$:

$$[\{S^0_{m_1}\}, \{S^0_{m_2}\}, \dots, \{S^0_{m_M}\}], \quad [\{S^1_{m_1}\}, \{S^1_{m_2}\}, \dots, \{S^1_{m_M}\}], \cdots, [\{S^P_{m_1}\}, \{S^P_{m_2}\}, \dots, \{S^P_{m_M}\}].$$

Here, each $\{S\}$ is a multiset of N scalar scores; superscripts index original (0) and perturbation types ($i = 1, \dots, P$); subscripts index metrics ($m_1, \dots, m_M$; e.g., coherence, consistency, fluency, relevance in SummEval [10]).

2.3 Step 3: Statistical Analysis

For each perturbation type i and metric m_j, we test whether original scores tend to exceed perturbed scores using a one-sided Wilcoxon Signed-Rank Test (paired, nonparametric): null $H_0 : S^0_{m_j}$ and $S^i_{m_j}$ have the same distribution; alternative $H_1 : S^0_{m_j} > S^i_{m_j}$ in location. This yields a p-value $p^i_{m_j}$. We then combine metric-wise evidence for each perturbation via the harmonic-mean p-value (optionally expert-weighted), and finally map to positive discernment scores. Formally,

$$p^i_{m_j} = \text{W-Test}(\{S^0_{m_j}\}, \{S^i_{m_j}\};\ H_1 : S^0_{m_j} > S^i_{m_j}).$$

$$p^i = \frac{1}{\sum_{j=1}^{M} \frac{1}{p^i_{m_j}}}, \qquad p^{i,EW} = \frac{1}{\sum_{j=1}^{M} \frac{EW^i_{m_j}}{p^i_{m_j}}}, \quad \text{with} \sum_{j=1}^{M} EW^i_{m_j} = 1.$$

$$D^i = \log_{0.05}(p^i), \qquad D^{i,EW} = \log_{0.05}(p^{i,EW}).$$

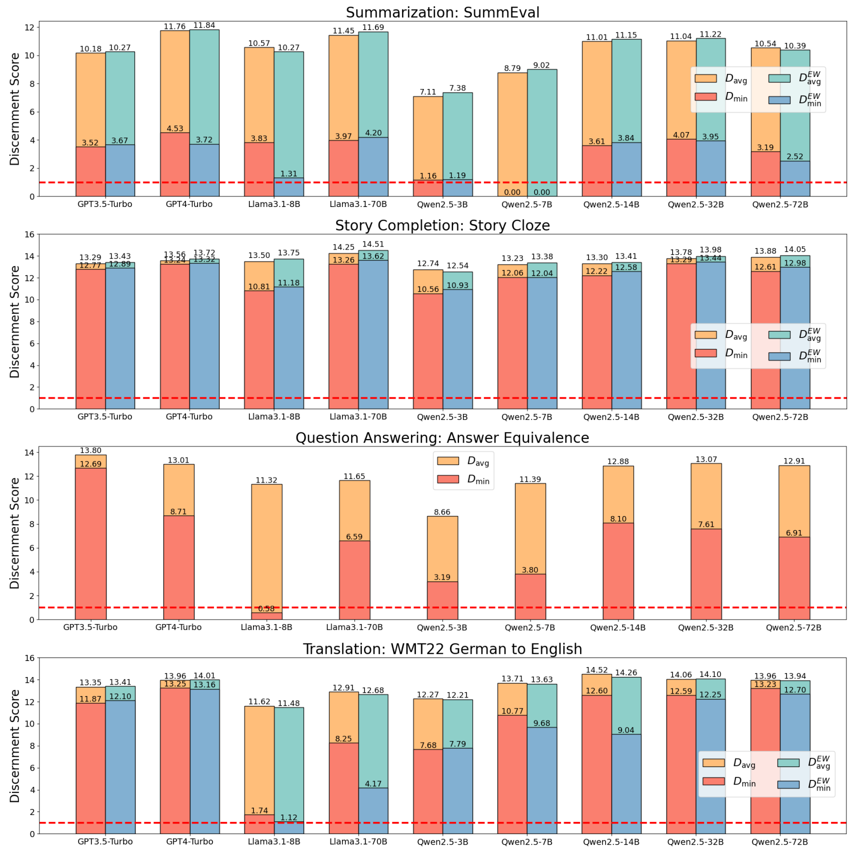

Figure 3: Selected DHP results across four tasks, red line means $p = 0.05$. Full results across all datasets and model families are provided in Appendix Figures 4 and 5.

We summarize per model by reporting averages and minima over perturbations: D_{avg}, D_{avg}^{EW} and D_{min}, D_{min}^{EW}. To avoid over-weighting prolific levels, we use level-balanced averaging so character-, word-, and sentence-level perturbations contribute equally. Scores above 1 correspond to $p < 0.05$, which means a statistically significant good discernment ability (larger score means stronger discernment).

3 Benchmarking LLM Discernment

We benchmark eight model families—GPT-3.5, GPT-4, Llama 3, Llama 3.1, Qwen 1.5, Qwen 2.5, Vicuna, and Mistral—across four tasks and six datasets (details in Tables 1 and 2). We evaluate N=100 items per dataset and compute D and D^{EW} over perturbations.

Figure 3 shows the selected result. Overall, most modern models achieve $D_{\text{avg}} > 1$, indicating consistent penalization of degraded texts. Larger and stronger models are generally more stable; smaller models struggle with challenging settings (e.g., long scientific summaries and multilingual translation). We suggest using models where $D_{\text{min}} > 1$ to ensure that they have a strong enough discernment ability. For full results and analysis, please refer to Appendix A.

4 Conclusion

We introduce the DHP benchmark to assess the discernment capabilities of LLMs as evaluators across various NLG tasks. Our approach not only provides benchmarking results for LLMs but also establishes a robust framework to evaluate how effectively LLMs can identify quality issues, thus serving as competent NLG evaluators. While most models generally perform well, their performance is significantly influenced by factors such as model size, task type, and dataset complexity. By identifying specific weaknesses of LLMs in evaluating NLG tasks, this benchmark aids researchers in enhancing "LLM-as-a-judge" methodologies and improving overall LLM performance.

References

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

[2] Jannis Bulian, Christian Buck, Wojciech Gajewski, Benjamin Börschinger, and Tal Schuster. Tomayto, tomahto. beyond token-level answer equivalence for question answering evaluation. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 291–305, 2022.

[3] Aljoscha Burchardt. Multidimensional quality metrics: a flexible system for assessing translation quality. In *Proceedings of Translating and the Computer 35*, 2013.

[4] Chi-Min Chan, Weize Chen, Yusheng Su, Jianxuan Yu, Wei Xue, Shanghang Zhang, Jie Fu, and Zhiyuan Liu. Chateval: Towards better llm-based evaluators through multi-agent debate. *arXiv preprint arXiv:2308.07201*, 2023.

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

[6] Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality. March 2023.

[7] Yu-Neng Chuang, Ruixiang Tang, Xiaoqian Jiang, and Xia Hu. Spec: a soft prompt-based calibration on performance variability of large language model in clinical notes summarization. *Journal of Biomedical Informatics*, 151:104606, 2024.

[8] Michael Desmond, Zahra Ashktorab, Qian Pan, Casey Dugan, and James M Johnson. Evalullm: Llm assisted evaluation of generative outputs. In *Companion Proceedings of the 29th International Conference on Intelligent User Interfaces*, pages 30–32, 2024.

[9] Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

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

[11] Saadia Gabriel, Asli Celikyilmaz, Rahul Jha, Yejin Choi, and Jianfeng Gao. GO FIGURE: A meta evaluation of factuality in summarization. In Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli, editors, *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 478–487, Online, August 2021. Association for Computational Linguistics.

[12] Mingqi Gao, Xinyu Hu, Jie Ruan, Xiao Pu, and Xiaojun Wan. Llm-based nlg evaluation: Current status and challenges. *arXiv preprint arXiv:2402.01383*, 2024.

[13] Vivek Gupta, Prerna Bharti, Pegah Nokhiz, and Harish Karnick. Sumpubmed: Summarization dataset of pubmed scientific article. In *Proceedings of the 2021 Conference of the Association for Computational Linguistics: Student Research Workshop*. Association for Computational Linguistics, 2020.

[14] Xinyu Hu, Mingqi Gao, Sen Hu, Yang Zhang, Yicheng Chen, Teng Xu, and Xiaojun Wan. Are LLM-based evaluators confusing NLG quality criteria? In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9530–9570, Bangkok, Thailand, August 2024. Association for Computational Linguistics.

[15] C Harry Hui and Harry C Triandis. Effects of culture and response format on extreme response style. *Journal of cross-cultural psychology*, 20(3):296–309, 1989.

[16] 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. Mistral 7b. *arXiv preprint arXiv:2310.06825*, 2023.

[17] Natalia D Kieruj and Guy Moors. Response style behavior: question format dependent or personal style? *Quality & Quantity*, 47:193–211, 2013.

[18] Ruosen Li, Teerth Patel, and Xinya Du. Prd: Peer rank and discussion improve large language model based evaluations. *arXiv preprint arXiv:2307.02762*, 2023.

[19] Zhen Li, Xiaohan Xu, Tao Shen, Can Xu, Jia-Chen Gu, and Chongyang Tao. Leveraging large language models for nlg evaluation: A survey. *arXiv preprint arXiv:2401.07103*, 2024.

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

[21] Yuxuan Liu, Tianchi Yang, Shaohan Huang, Zihan Zhang, Haizhen Huang, Furu Wei, Weiwei Deng, Feng Sun, and Qi Zhang. Calibrating llm-based evaluator. *arXiv preprint arXiv:2309.13308*, 2023.

[22] Adian Liusie, Potsawee Manakul, and Mark Gales. Llm comparative assessment: Zero-shot nlg evaluation through pairwise comparisons using large language models. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 139–151, 2024.

[23] Nasrin Mostafazadeh, Michael Roth, Annie Louis, Nathanael Chambers, and James Allen. Lsdsem 2017 shared task: The story cloze test. In *Proceedings of the 2nd Workshop on Linking Models of Lexical, Sentential and Discourse-level Semantics*, pages 46–51, 2017.

[24] Jekaterina Novikova, Ondřej Dušek, and Verena Rieser. Rankme: Reliable human ratings for natural language generation. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 72–78, 2018.

[25] OpenAI. Gpt4-turbo. 2023.

[26] Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. Squad: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, 2016.

[27] Aadesh Salecha, Molly E Ireland, Shashanka Subrahmanya, João Sedoc, Lyle H Ungar, and Johannes C Eichstaedt. Large language models show human-like social desirability biases in survey responses. *arXiv preprint arXiv:2405.06058*, 2024.

[28] Stephanie Schoch, Diyi Yang, and Yangfeng Ji. "this is a problem, don't you agree?" framing and bias in human evaluation for natural language generation. In *Proceedings of the 1st Workshop on Evaluating NLG Evaluation*, pages 10–16, 2020.

[29] Sijun Tan, Siyuan Zhuang, Kyle Montgomery, William Y Tang, Alejandro Cuadron, Chenguang Wang, Raluca Ada Popa, and Ion Stoica. Judgebench: A benchmark for evaluating llm-based judges. *arXiv preprint arXiv:2410.12784*, 2024.

[30] Yves Van Vaerenbergh and Troy D Thomas. Response styles in survey research: A literature review of antecedents, consequences, and remedies. *International journal of public opinion research*, 25(2):195–217, 2013.

[31] Jiaan Wang, Yunlong Liang, Fandong Meng, Zengkui Sun, Haoxiang Shi, Zhixu Li, Jinan Xu, Jianfeng Qu, and Jie Zhou. Is chatgpt a good nlg evaluator? a preliminary study. *arXiv preprint arXiv:2303.04048*, 2023.

[32] Hui Wei, Shenghua He, Tian Xia, Andy Wong, Jingyang Lin, and Mei Han. Systematic evaluation of llm-as-a-judge in llm alignment tasks: Explainable metrics and diverse prompt templates. *arXiv preprint arXiv:2408.13006*, 2024.

[33] F Wilcoxon. Individual comparisons by ranking methods. *Biometrics Bulletin*, 1(6):80–83, 1945.

[34] Jiayi Yuan, Jiamu Zhang, Andrew Wen, and Xia Hu. The science of evaluating foundation models. *arXiv preprint arXiv:2502.09670*, 2025.

[35] Xinghua Zhang, Bowen Yu, Haiyang Yu, Yangyu Lv, Tingwen Liu, Fei Huang, Hongbo Xu, and Yongbin Li. Wider and deeper llm networks are fairer llm evaluators. *arXiv preprint arXiv:2308.01862*, 2023.

[36] Zhuosheng Zhang, Aston Zhang, Mu Li, and Alex Smola. Automatic chain of thought prompting in large language models. In *The Eleventh International Conference on Learning Representations*, 2022.

A Full Result and Findings

A.1 Overall Assessment

A broad analysis confirms that most evaluated LLMs demonstrate a foundational ability to discern quality issues, with nearly all models achieving average discernment scores (D_{avg} and D_{avg}^{EW}) above 1. A clear hierarchy of capability is evident. The top-performing models are consistently GPT-4 Turbo, Llama 3.1-70B, and Qwen 2.5-72B (Figure 4), which exhibit high discernment and stability across all tasks.

However, several older and smaller models show significant weaknesses. In particular, Vicuna 1.5-7B and smaller Qwen 1.5 models (e.g., Qwen 1.5-7B) perform poorly, with average discernment scores falling below 1 in the challenging WMT22 Chinese-to-English translation task (Figure 5f). This indicates a failure to reliably penalize degraded translations, rendering them unsuitable for such evaluation scenarios.

A.2 Other Observations

Trends Regarding the Size of LLMs The trend that larger models yield better discernment is strongly supported by the newer model families but shows inconsistencies in older ones. The Qwen 2.5 series (14B, 32B, 72B) in Figure 4 displays a clear and predictable improvement in performance and stability with increased model size. In contrast, the older Qwen 1.5 series in Figure 5 exhibits anomalies, such as the 4B model outperforming the 7B model in translation tasks (Figure 5e, f), suggesting that scaling does not always guarantee improved performance across all capabilities.

Limitations of Smaller LLMs Across both model generations, smaller LLMs (under 13B) consistently struggle with stability, reflected in very low minimum discernment scores (D_{min} and D_{min}^{EW}). This is particularly evident in complex tasks like SumPubMed summarization and WMT22 translation. Models such as Llama3-8B (in both figures), Mistral-7B, Vicuna 1.5-7B, and the smaller Qwen 1.5 models (Figure 5) all demonstrate this vulnerability. Their instability in challenging scenarios makes them unreliable evaluators where nuance and domain knowledge are critical.

Metric Misunderstanding Phenomenon The discrepancy between scores with and without expert weights (D vs. D^{EW}) highlights a "metric misunderstanding" phenomenon in less advanced models. This is most prominent in Llama3-8B and Vicuna 1.5-7B on translation tasks (Figure 5e, f), where D_{min}^{EW} is substantially lower than D_{min}. In stark contrast, the newest large models like Llama 3.1-70B and the Qwen 2.5 series (Figure 4) show almost no gap between the two metrics, indicating a more robust and reliable interpretation of the evaluation criteria.

Variations in Task Performance The relative difficulty of the evaluation tasks is remarkably consistent across all tested models. The Story Cloze Test (Figure 4c, Figure 5c) is uniformly the easiest task, with nearly all models achieving high and stable scores. Conversely, the SumPubMed dataset (Figure 4b, Figure 5b) is the most challenging. For this task, most models, including many large ones, see their minimum discernment scores drop below 1. This reinforces that evaluating content in specialized, knowledge-intensive domains remains a significant hurdle for LLM-based evaluators and necessitates careful, task-specific model selection.

B Response Styles: Additional Analysis

Previous studies focus on the alignment between human and LLM evaluators, using correlation metrics to gauge the LLMs' performance in NLG evaluation tasks [20, 5]. However, these studies often overlook an important variable of evaluators: **Response Styles** which refer to a respondent's consistent manner of answering survey questions, regardless of the content [30]. Despite similar levels of professionalism, annotators may assign different scores to the same questionnaire due to differences in age, gender, personality, cultural background, and ethnic group [30, 15, 17]. Similarly, LLMs, trained on diverse datasets, may also exhibit biases in their responses [27]. This discrepancy casts doubt on the previous methods used to compare human and LLM scores. Since quality-based scoring often relies heavily on a few experts' annotations, the final alignment scores tend to favor models that share similar response styles with these specific experts.

We illustrate this with an example of the response styles of five LLMs tasked with annotating quality scores for human reference data from the SummEval dataset [10]. We averaged the scores across four metrics for each data point and plotted both the Pearson correlation coefficient (ρ) and the average score distributions of the five models. After perturbing the original data by replacing some named entities with fictional ones in the summaries (Fictional Named Entities in Table 1), we repeated the quality evaluation. As shown in Figure 6, all models detected the changes and adjusted their scores accordingly, though their scoring distributions varied significantly. For instance, Llama3 [9], Mistral [16], and Qwen [1] models assign higher scores to the original data and moderate scores to the perturbed data. In contrast, GPT4-Turbo [25] and Vicuna [6] models tend to give moderate scores to the original data and much lower scores to the perturbed data. The variance in the response distributions indicates the presence of bias that can significantly affect alignment (ρ), illustrating that alignment is not a direct or credible metric for assessing the ability of LLMs as NLG evaluators. It is crucial to develop a new metric and measurement for evaluation that is not influenced by the evaluators' biased response styles, ensuring a more accurate and fair assessment of LLM capabilities.

C NLG Tasks and Metrics

C.1 Summarization

We utilize the SummEval [10] (MIT license) and SumPubMed [13] datasets (MIT license) for our summarization tasks. The SummEval dataset comprises 100 news articles, each accompanied by multiple reference and generated summaries. For our analysis, we exclusively use the reference summaries, selecting the one with the highest number of sentences from each article to facilitate perturbation. The SumPubMed dataset contains 32,000 long scientific articles along with their abstracts serving as reference summaries. We only use the "BACKGROUND" sections of these articles and summaries. We randomly select 100 pairs of articles and their corresponding summaries.

For the evaluation of summarization performance, we adhere to the metrics defined by SummEval [10], specifically focusing on Coherence, Consistency, Fluency, and Relevance.

C.2 Story Completion

In this story completion task, we utilize the public Story Cloze Test dataset [23], which comprises four-sentence stories each paired with a reference and wrong ending. We select 100 datapoints at random from the validation set for our analysis.

Given the absence of explicitly defined quality metrics for the dataset, we adapt metrics from summarization tasks—Coherence, Consistency, and Fluency. Coherence evaluates the story's overall structure and narrative flow. Consistency measures how well the ending maintains the established tone, setting, character development, and narrative style of the story. Fluency focuses on the linguistic and stylistic quality of the story's conclusion.

C.3 Question Answering

For the question answering task, we employ the Answer Equivalence dataset [2] (Apache-2.0 license), which is a modified version of the SQuAD dataset [26]. We specifically select reference answers that exceed 150 characters to facilitate perturbation. From this filtered set, we randomly choose 100 question-answer pairs.

We adapt the original rating tasks of the dataset into a single metric: Answer Quality. This metric assesses whether the answer provides a comprehensive and accurate response to the question, effectively capturing the essence of the content discussed in the paragraph.

C.4 Translation

We utilize two subsets from the WMT-22 general (news) translation dataset: German-to-English and Chinese-to-English sets which are freely available for research purposes. For our analysis, we select the test sets with reference translations, ensuring each translation exceeds 300 characters in length. We randomly choose 100 datapoints from each subset for evaluation.

In assessing translation tasks, we adopt two principal metrics from the Multidimensional Quality Metrics (MQM) framework [3]: Accuracy and Fluency. Accuracy measures how closely the translation mirrors the source text, focusing on the absence of additions, omissions, or mistranslations. Fluency evaluates the translation's compliance with the linguistic norms of the target language, specifically examining spelling, grammar, and consistency.

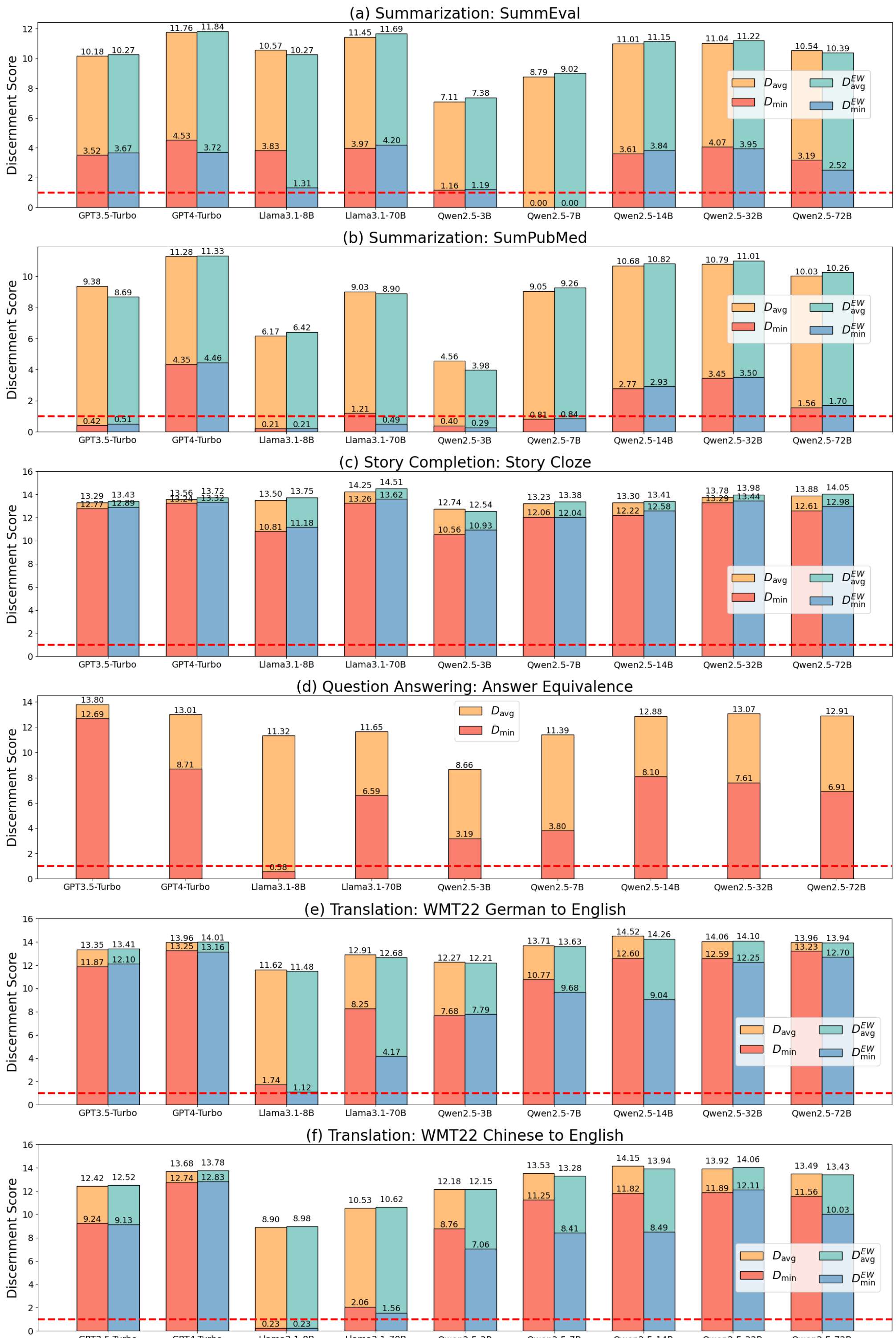

Figure 4: Full DHP results for all datasets and model families: SummEval, SubPubMed, Story Cloze, Answer Equivalence, WMT22 De–En, and WMT22 Zh–En.

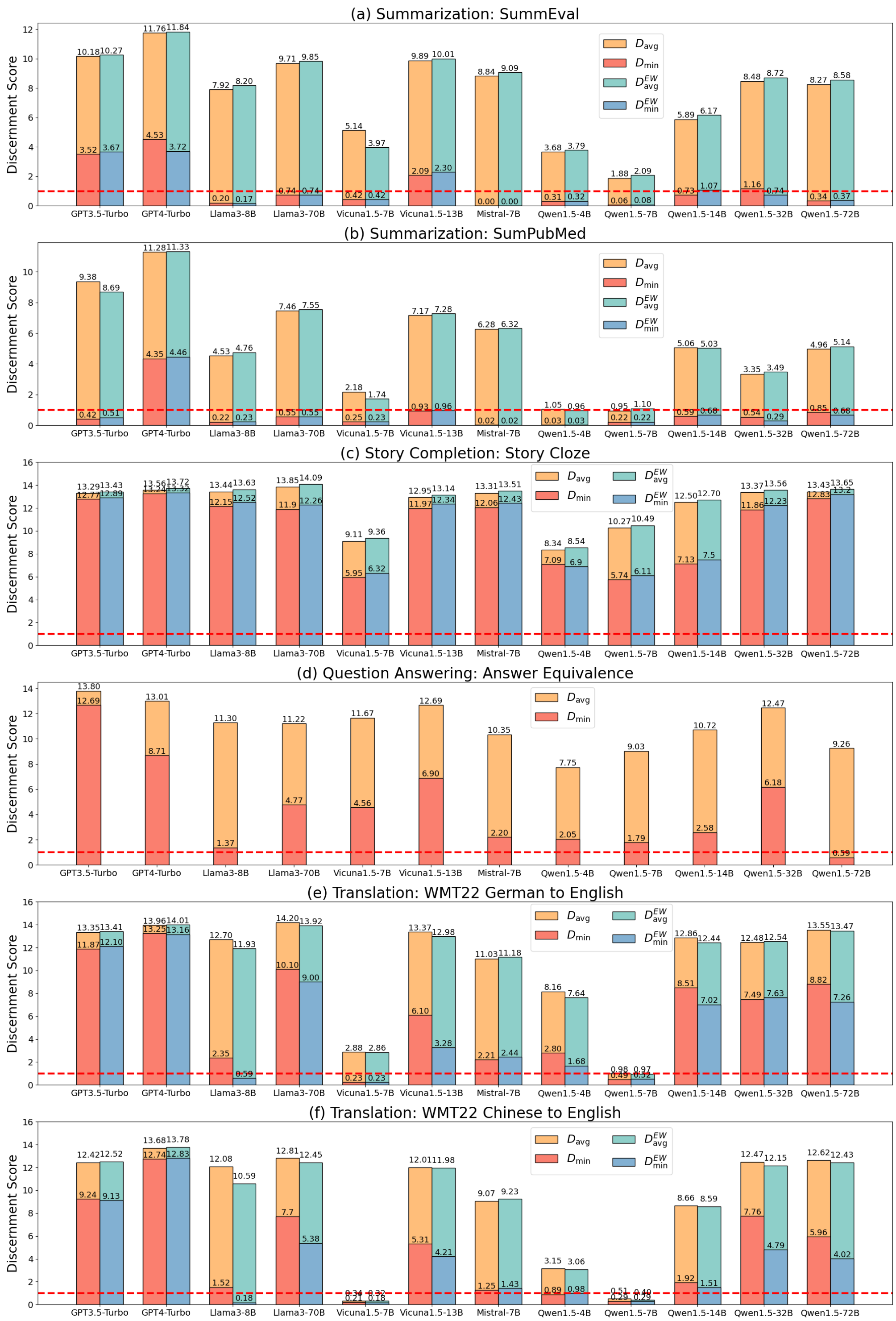

Figure 5: Results for older model families (Llama 3, Vicuna, Qwen 1.5).

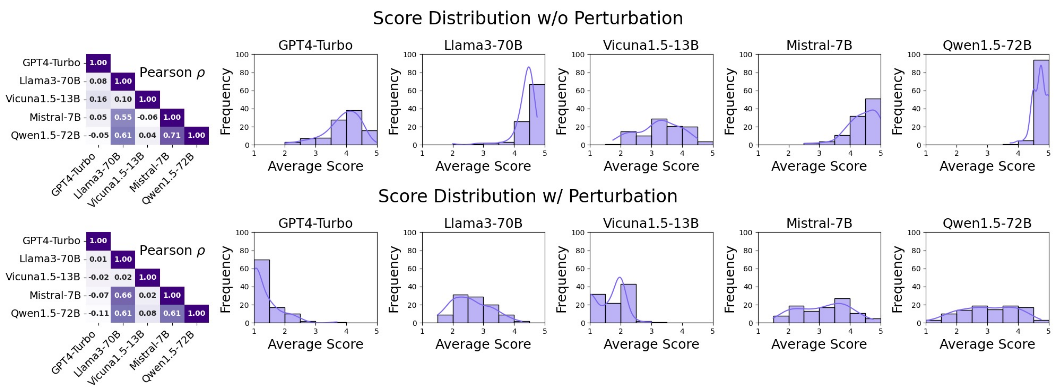

Figure 6: Response styles of five LLMs on SummEval [10]. Models react to perturbations but follow distinct score distributions, revealing style differences that can bias alignment metrics.

Table 2: Summary of hierarchical perturbation methods applied to different NLG tasks, detailing the types of perturbations and their respective implementations based on character (C), word (W), and sentence-level (S) modification with rule-based (R) or LLM-based (L) approaches.

Task	Avg NLTK Statistics	Perturbation	Description
Summarization	SummEval: 340.4 Characters 58.3 Words 4.0 Sentences SumPubMed 803.5 Characters 114.9 Words 5.5 Sentences	(C, R) Random Deletions	Delete k alphanumeric characters randomly. SummEval: k=10 for Minor, k=50 for Major; SumPubMed: k=20 for Minor, k=100 for Major.
		(C, R) Random Typos	Add k random typographical errors with "typo" package. SummEval: k=10 for Minor, k=50 for Major; SumPubMed: k=20 for Minor, k=100 for Major.
		(W, L) Fictional Named Entities	Substitute one ore more named entities with in the summary (e.g., names, locations, specific numbers, technical terms, etc.) with fictional counterparts.
		(W, L) Grammatical Errors	Modify the summary for creating two or more grammatical errors, such as subject-verb disagreement, noun-pronoun disagreement, incorrect verb tense, misuse of preposition, and sentence fragment, etc.
		(S, R) Reordering	Random shuffle k sentences in the summary. k=2 for Minor, k=all for Major.
		(S, L) Rewriting and Insertion	Select one or more sentences from the summary, then rephrase them and insert the rewritten versions immediately after the original sentences.
Story Completion	Story Cloze Test: 38.7 Characters 7.4 Words 1.0 Sentences	(C, R) Random Deletions	Delete 5 alphanumeric characters randomly.
		(C, R) Random Typos	Add 5 random typographical errors with "typo" package.
		(W, L) Fictional Named Entities	Substitute one critical named entities within the ending sentence (e.g., a name, a location, a specific number, etc.) with a fictional counterpart.
		(W, L) Grammatical Errors	Modify the ending for creating one grammatical error, such as subject-verb disagreement, noun-pronoun disagreement, incorrect verb tense, misuse of preposition, and sentence fragment, etc.
		(S, R) Random Ending Sentence	Replace the ending with a random one from another story.
		(S, R) Wrong Ending Sentence	Replace the ending with the wrong ending of the dataset.
Question Answering	Answer Equivalence: 156.2 Characters 23.9 Words 1.0 Sentences	(C, R) Random Deletions	Delete k alphanumeric characters randomly. k=5 for Minor, k=25 for Major.
		(C, R) Random Typos	Add k random typographical errors with "typo" package. k=5 for Minor, k=25 for Major.
		(W, L) Fictional Named Entities	Substitute one or more critical named entities within the answer (e.g., names, locations, specific numbers, technical terms, etc.) with fictional counterparts.
		(W, L) Grammatical Errors	Modify the answer for creating one or more grammatical errors, such as subject-verb disagreement, noun-pronoun disagreement, incorrect verb tense, misuse of preposition, and sentence fragment, etc.
		(S, R) Random Answer	Replace the answer with a random one to another question.
Translation	WMT-22 German-to-English: 436.8 Characters 71.0 Words 3.8 Sentences WMT-22 Chinese-to-English: 434.1 Characters 66.4 Words 1.1 Sentences	(C, R) Random Deletions	Delete k alphanumeric characters randomly. k=10 for Minor, k=50 for Major.
		(C, R) Random Typos	Add k random typographical errors with "typo" package. k=10 for Minor, k=50 for Major.
		(W, R) Random Deletions	Delete k continuous words in the translation randomly. k=5 for Minor, k=25 for Major.
		(W, L) Fictional Named Entities	Substitute one or more critical named entities within the translation (e.g., names, locations, specific numbers, technical terms, etc.) with fictional counterparts.
		(W, L) Grammatical Errors	Modify the translation for creating two or more grammatical errors, such as subject-verb disagreement, noun-pronoun disagreement, incorrect verb tense, misuse of preposition, and sentence fragment, etc.

D Hierarchical Perturbation

The specifics of the hierarchical perturbations are detailed in Table 2. We perform these perturbations based on character, word, and sentence-level statistical data of the texts, which are presented in Table 2. Our rule-based perturbations include simple text deletions, typographical errors using existing software tools, reordering of sentences, and the incorporation of random or incorrect sentences from other data.

For LLM-based perturbations, we employ GPT4-Turbo, modifying the reference text via Auto-CoT [36] prompts to generate the detailed procedural perturbation steps. Below, we provide an example of how the "Minor Fictional Named Entities" perturbation is applied to the summarization tasks:

Minor Fictional Named Entities Perturbation Prompt:

You will be given one summary written for an article. Your task is to adjust the summary by implementing a specific change.

Please make sure you read and understand these instructions carefully.

Adjustment: Please substitute only one critical named entity within the summary (e.g., a name, a location, a specific number, a technical term, etc.) with a fictional counterpart.

Adjustment Steps:

1. Identify the critical named entity within the summary. This could be a person's name, a location, a specific number, or any other specific detail that is crucial to the summary.

2. Create a fictional counterpart for the identified entity. This could be a fictional name, a fictional location, a fictional number, a fictional technical term etc. Make sure that the fictional counterpart is appropriate and fits within the context of the summary.

3. Replace the identified entity with its fictional counterpart in the summary. Ensure that the replacement is grammatically correct and maintains the overall meaning and flow of the summary.

4. Review the adjusted summary to ensure that it still makes sense and conveys the main points of the article, despite the change in one critical named entity.

Summary:

SUMMARY_HERE

Revised Summary:

E Expert Weights

We invite 10 volunteer experts with extensive backgrounds in NLP/NLG research to complete an expert weight survey. The interface of this survey is displayed in Figure 7, which includes the survey instructions, definitions of the tasks and metrics, data types, and descriptions of quality issues associated with the perturbation methods. The experts are asked to select the metric they believe is most impacted by each quality issue presented. We then utilize their responses as weights for combining the p-values. The results of these expert evaluations are detailed in Figure 8.

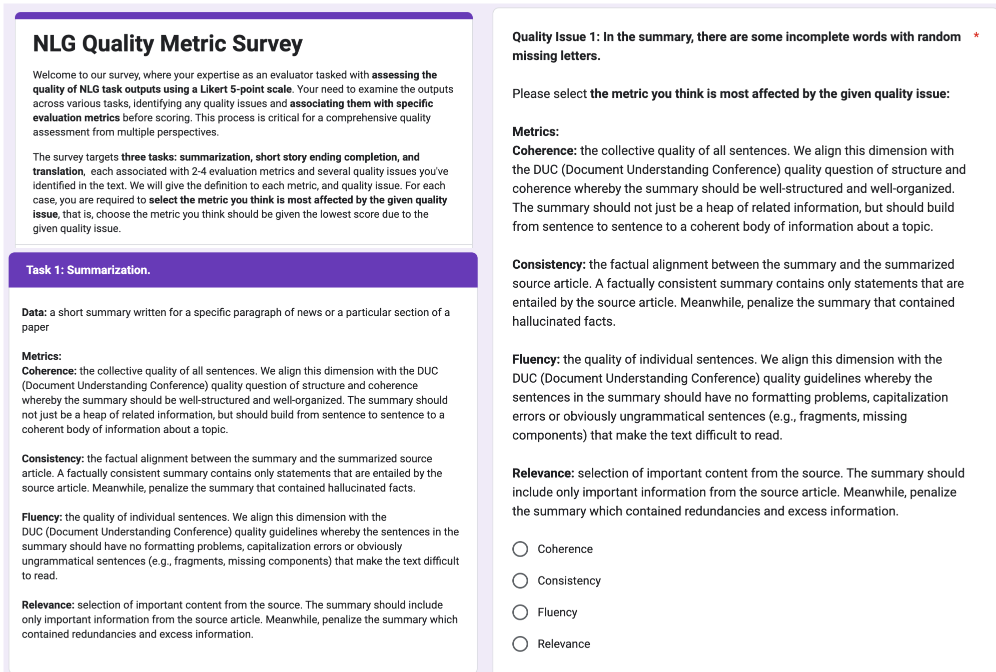

Figure 7: User interface of the expert weight survey conducted to determine the impact of various quality issues on NLG task metrics.

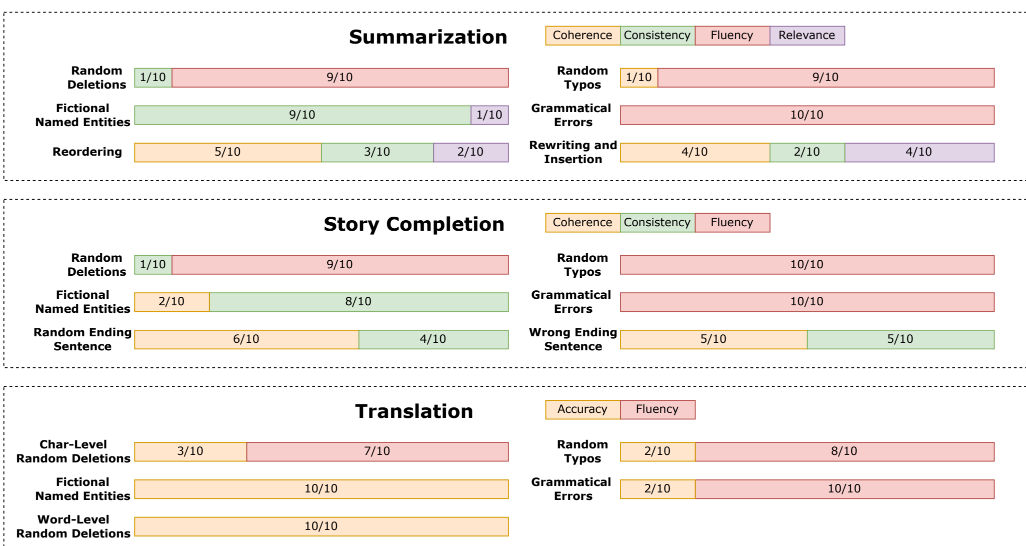

Figure 8: Graphical representation of the expert weights for each NLG task.

Table 3: Overview of large language models (LLMs) assessed in the DHP benchmark, specifying model versions and sources.

Model	Version	Source
GPT3.5-Turbo	gpt-3.5-turbo-0125	platform.openai.com/docs/models
GPT4-Turbo	gpt-4-1106-preview	platform.openai.com/docs/models
Llama3-8B	Meta-Llama-3-8B-Instruct	huggingface.co/meta-llama/Meta-Llama-3-8B-Instruct
Llama3-70B	Meta-Llama-3-70B-Instruct	huggingface.co/meta-llama/Meta-Llama-3-70B-Instruct
Llama3.1-8B	Meta-Llama-3-8B-Instruct	huggingface.co/meta-llama/Meta-Llama-3.1-8B-Instruct
Llama3.1-70B	Meta-Llama-3-70B-Instruct	huggingface.co/meta-llama/Meta-Llama-3.1-70B-Instruct
Vicuna1.5-7B	vicuna-7b-v1.5-16k	huggingface.co/lmsys/vicuna-7b-v1.5-16k
Vicuna1.5-13B	vicuna-13b-v1.5-16k	huggingface.co/lmsys/vicuna-13b-v1.5-16k
Mistral-7B	Mistral-7B-Instruct-v0.2	huggingface.co/mistralai/Mistral-7B-Instruct-v0.2
Qwen1.5-4B	Qwen1.5-4B-Chat	huggingface.co/Qwen/Qwen1.5-4B-Chat
Qwen1.5-7B	Qwen1.5-7B-Chat	huggingface.co/Qwen/Qwen1.5-7B-Chat
Qwen1.5-14B	Qwen1.5-14B-Chat	huggingface.co/Qwen/Qwen1.5-14B-Chat
Qwen1.5-32B	Qwen1.5-32B-Chat	huggingface.co/Qwen/Qwen1.5-32B-Chat
Qwen1.5-72B	Qwen1.5-72B-Chat	huggingface.co/Qwen/Qwen1.5-72B-Chat
Qwen2.5-3B	Qwen1.5-4B-Chat	huggingface.co/Qwen/Qwen2.5-3B
Qwen2.5-7B	Qwen1.5-7B-Chat	huggingface.co/Qwen/Qwen2.5-7B
Qwen2.5-14B	Qwen1.5-14B-Chat	huggingface.co/Qwen/Qwen2.5-14B
Qwen2.5-32B	Qwen1.5-32B-Chat	huggingface.co/Qwen/Qwen2.5-32B
Qwen2.5-72B	Qwen1.5-72B-Chat	huggingface.co/Qwen/Qwen2.5-72B

F LLM Evaluation

We evaluate five series of large language models (LLMs), details of which are provided in Table 3. Due to the extensive length of text data from the SumPubMed dataset [13], which can exceed the 4K context window, we evaluate the models capable of processing long texts ($\geq$ 8K tokens). The GPT series is operated using the OpenAI API, and the open-source LLMs are executed on a server with 8 Nvidia A100 GPUs. We set the temperature parameters to 0 and maintain the default values for the top_p parameters. Throughout the evaluation process, each model score 5 times on each metric to calculate a final average score. We use the *scipy.stats.wilcoxon* to conduct the Wilcoxon Signed-Rank Test.

G Evaluation Prompts

We follow the guidelines of G-Eval [20] and utilize the Auto-CoT method [36] to construct our evaluation prompts. Below is an example of the prompt used for assessing the Coherence metric in summarization tasks:

You will be given a summary written for an article. Your task is to rate the summary on one metric. Please make sure you read and understand these instructions carefully. Please keep this document open while reviewing, and refer to it as needed.

Evaluation Criterion: Coherence (1-5) - the collective quality of all sentences. We align this dimension with the DUC quality question of structure and coherence whereby the summary should be well-structured and well-organized. The summary should not just be a heap of related information, but should build from sentence to sentence to a coherent body of information about a topic.

Evaluation Steps:

1. Read the Summary Thoroughly: Before diving into the evaluation, ensure that you have a clear understanding of the entire summary. Reading it more than once might be necessary.

2. Identify the Central Topic: A coherent summary will have a clear central topic or theme. Identify this topic and see if the subsequent information revolves around it.

3. Check for Logical Flow: Review the summary for logical sequencing. Sentences should follow one another in a way that makes sense and allows the reader to easily follow the progression of information.

4. Look for Transitional Elements: Coherent summaries often have clear transitions between sentences or ideas. This could be in the form of transitional words, phrases, or connecting ideas that tie one sentence to the next.

5. Identify Redundancies: Check if the same information is repeated in different sentences. Redundancies can disrupt the flow and coherence of a summary.

6. Note Any Gaps or Jumps: If there are sudden jumps in topics or if crucial information seems to be missing, this can harm the coherence of the summary. A well-organized summary should present a holistic view of the topic without leaving the reader with questions.

7. Assess Clarity: Even if the content is technically accurate, if it's written in a convoluted or unclear manner, it can disrupt coherence. The sentences should be clear and easily understandable.

8. Consider the Conclusion: A coherent summary often wraps up or comes to a conclusion that ties the presented information together. It doesn't necessarily need a formal conclusion, but the end should feel natural and not abrupt.

9. Rate the Summary: Based on the above steps, assign a score between 1-5 for coherence. - 1: Very incoherent. The summary lacks structure, has sudden jumps, and is difficult to follow. - 2: Somewhat incoherent. The summary has some semblance of structure, but has significant flaws in flow and organization. - 3: Neutral. The summary is decently organized, with minor issues in flow and structure. - 4: Mostly coherent. The summary is well-structured with very few minor coherence issues. - 5: Highly coherent. The summary is excellently organized, flows seamlessly, and builds information logically from start to end.

Source Article:

ARTICLE_HERE

Summary:

SUMMARY_HERE

Evaluation Score (please don't give any feedback, just give a score ONLY) - Coherence:

H Related Work

Recent advancements highlight the significant potential of utilizing LLMs as evaluators for a variety of NLP tasks. Extensive empirical evidence supports this viewpoint, as demonstrated by studies [20, 5, 14, 8, 31], which assert that the evaluation behaviors of pretrained LLM-based evaluators are well-aligned with those of human preference [21]. Liusie et al.[22] further show that comparative assessments using LLM evaluators outperform prompt-based techniques, though they identify potential positional biases and propose corresponding solutions. Despite the great assessment performance of a single LLM, advanced studies involve multi-LLM agents [4, 35, 18] or human experts [12, 19] to further increase the judging capability.

While the application of LLMs as judges is a burgeoning area of research, it is imperative to assess their reliability and effectiveness in evaluative roles. To this end, several benchmarks have been recently proposed to evaluate LLMs as judges. For example, JudgeBench [29] is designed to assess LLM-based judges on challenging response pairs spanning knowledge, reasoning, math, and coding. Additionally, LLM-judge-eval [32] evaluates tasks such as summarization and alignment, incorporating metrics like flipping noise and length bias.

However, despite the progress in LLMs as judges, several challenges persist. First, human involvement remains a crucial factor in both evaluation and alignment, which raises concerns about the extent to which human biases influence LLM-based evaluations. Second, human evaluators themselves are inherently biased, meaning that even if an LLM aligns well with human preferences, it does not necessarily guarantee fairness or accuracy. Additionally, LLMs may misinterpret NLG evaluation metrics [14], making simple alignment scores unreliable. To overcome these challenges, our work focuses on developing automated and comprehensive methodologies to test the reliability of LLM-based evaluations.