HypoEval: Hypothesis-Guided Evaluation For Natural Language Generation

Anonymous ACL submission

Abstract

Large language models (LLMs) have demonstrated great potential for automating the evaluation of natural language generation. Previous frameworks of LLM-as-a-judge fall short in two ways: they either use zero-shot setting without consulting any human input, which leads to low alignment, or fine-tune LLMs on labeled data, which requires a non-trivial number of samples. Moreover, previous methods often provide little reasoning behind automated evaluations. In this paper, we propose HYPOEVAL, **Hypo**thesis-guided **Eval**uation framework, which first uses a small corpus of human evaluations to generate more detailed rubrics for human judgments and then incorporates a checklist-like approach to combine LLM's assigned scores on each decomposed dimension to acquire overall scores [1]. With only 30 human evaluations, HypoEval achieves state-of-the-art performance in alignment with both human rankings (Spearman correlation) and human scores (Pearson correlation), on average outperforming G-Eval by 11.86% and fine-tuned LLAMA-3.1-8B-INSTRUCT with at least 3 times more human evaluations by 11.95%. Furthermore, we conduct systematic studies to assess the robustness of HYPOEVAL, highlighting its effectiveness as a reliable and interpretable automated evaluation framework.

1 Introduction

Automated evaluation of natural language generation has been an important and challenging task with the rapid development of automated systems for summarization, translation, open-ended story generation, and more (Fang et al., 2024; Yao et al., 2024). Traditional lexical metrics such as BLEU (Papineni et al., 2002) and ROUGE (Lin, 2004) have been shown to have low agreement with human judgments (Krishna et al., 2021). With the advancements of LLMs, recent research has extensively focused on LLM-as-a-judge, or using LLMs to perform reference-free automated evaluations of natural language generation (Chen et al., 2023; Fu et al., 2023; Gu et al., 2025).

Following Li et al. (2025), we broadly categorize existing LLM-based automated evaluation frameworks into prompting-based and tuning-based methods. On one hand, prompting-based evaluation methods such as G-Eval (Liu et al., 2023a) mostly use a zero-shot approach, which imposes a strict yet unnecessary restriction on the use of human evaluations or groundings. As a result, they lead to limited correlations with human annotations and leave room for improvement (Bavaresco et al., 2024; Krumdick et al., 2025). On the other hand, tuning-based methods (Yue et al., 2023; Liu et al., 2024a) require a large corpus of high-quality training data, with performance constrained to the specific dataset that they are trained on (Liu et al., 2025b), and can be computationally expensive or hard to apply to proprietary models. Furthermore, both categories of methods often lack explainability in their evaluation process. Although recent works on checklist-based frameworks for evaluating instruction-following or factuality (Que et al., 2024; Min et al., 2023; Tan et al., 2024) shed light on providing reasoning behind evaluations, their performance on evaluating other aspects of text generation can be inferior to non-checklist frameworks (Lee et al., 2024). This can be due to the fundamental difficulty of decomposing subjective aspects (e.g., engagement) of texts into atomic and easy-to-verify checklists.

To address these limitations, we propose HYPOEVAL, the first LLM-based evaluation framework that combines state-of-the-art hypothesis generation techniques to guide judge LLMs and improve human alignment. HYPOEVAL consists of a light training stage for hypothesis generation and then uses hypotheses to provide evaluation scores. With

[1] In this study, the term "hypothesis", "checklist", "rubric", and "decomposed dimension" are used interchangeably.

a small corpus of human evaluation scores (in our implementation, we limit the number to 30), HYPOEVAL first uses a hypothesis generation framework to generate high-quality hypotheses, which are framed as decomposed dimensions for evaluation, from human evaluation results and existing literature on evaluation. The decomposed dimensions serve as rubrics and break down a subjective aspect of evaluation into different attributes that are easier for an LLM to understand. Then in the evaluation stage, with each decomposed dimension formulated as a non-binary checklist (i.e., the answer to the checklist can be a range of numbers on the Likert scale), HYPOEVAL combines an evaluator LLM's assigned scores on each decomposed dimension and gets an overall score for an evaluated text. We show that with only small-scale human evaluations and $O(N)$ computational complexity, HYPOEVAL is able to achieve state-of-the-art performance on representative tasks: on average outperforming G-Eval by 11.86% and fine-tuned LLAMA-3.1-8B-INSTRUCT- with more than 3 times more human evaluation scores - by 11.95%.

To summarize, our main contributions are as follows:

- We introduce HYPOEVAL, a tuning-free, sample-efficient framework for evaluating natural language generation.
- We demonstrate that HYPOEVAL achieves state-of-the-art performance in terms of correlation with human judgments across multiple datasets.
- With the generated hypotheses or decomposed dimensions in HypoEval, we provide more interpretable explanations in automated evaluations than previous methods.

2 Methods

We first give a formulation of hypothesis-guided text evaluation. In direct scoring, we evaluate on an input-output pair (x, y), where x is the prompt for generation (e.g. source text for summarization), and y is the generated content (e.g. summary). With an LLM $\mathcal{M}$ and an instruction prompt I that consists of task descriptions and definition of the evaluated aspect (e.g. coherence of summaries), we want to produce a score $\mathcal{M}(x, y, I)$ that matches human score s well (usually measured in Pearson or Spearman correlation).

In hypothesis-guided text evaluation, we first generate a hypothesis bank $\mathcal{H} = \{h_1, h_2, \ldots, h_n\}$ from a training set $S_{\text{tr}} = \{(x_1, y_1, s_1), \ldots, (x_m, y_m, s_m)\}$ and a corpus of summaries $\mathcal{L}$ of relevant literature, where x_i and y_i are inputs and outputs, and s_i are scores given by human experts on (x_i, y_i). Here, each hypothesis is formulated as a rubric on a decomposed dimension. Ideally, the hypothesis bank $\mathcal{H}$ contains multiple decomposed dimensions that human experts consider when giving the scores in $\mathcal{S}_{\text{tr}}$. Then, we use a checklist-like approach to use the hypotheses to evaluate on unseen sample (x, y) and give a score $\mathcal{M}(x, y, I, \mathcal{H})$.

Hypothesis Generation from Small-scale Data and Literature Following HYPOGENIC (Zhou et al., 2024b) and HYPOREFINE (Liu et al., 2025a), we utilize both data-driven and literature-driven hypothesis generation approaches to generate the hypothesis bank $\mathcal{H}$. That is, with an LLM $\mathcal{M}$ and hypothesis generation algorithm g, we have $\mathcal{H} = g_{\mathcal{M}}(S_{\text{tr}}, \mathcal{L})$.

Specifically, we build upon HYPOREFINE and introduce extensions tailored for continuous value prediction. Given a small-scale training set $S_{\text{tr}} = \{(x_1, y_1, s_1), \ldots, (x_m, y_m, s_m)\}$ (for main experiments, we set $|S_{\text{tr}}| = 30$) and the corpus of LLM-generated summaries of relevant literature $\mathcal{L}$ (details of literature collection and summary generation in Appendix B.1), we first want to generate a hypothesis bank $\mathcal{H}$. During the initial stage, an LLM-based hypothesis generation agent $\mathcal{M}_G$ is prompted with a set of initial data points $S_{\text{init}} \subset S_{\text{tr}}$ and $\mathcal{L}$ to generate an initial hypothesis bank $\mathcal{H}^{\text{init}} = \mathcal{M}_G(S_{\text{init}}, \mathcal{L})$, and set $\mathcal{H}^0 = \mathcal{H}^{\text{init}}$. Inspired by the Upper Confidence Bound algorithm for strategic regression (Liu and Chen, 2016), for each $h \in \mathcal{H}^0$, we use an evaluator $\mathcal{M}_E$ to evaluate on all (x_i, y_i) in S_{init} based on the detailed rubric stated in h and give a score $\mathcal{M}_E(x_i, y_i, I, h)$, where I is the instruction prompt. Then, we set the reward for h by:

$$r_h := \frac{1}{|S_{\text{init}}|} \sum_{(x_j, y_j, s_j) \in S_{\text{init}}} L(x_j, y_j, s_j) + \alpha \sqrt{\frac{\log |S_{\text{init}}|}{|S_{\text{init}}|}},$$

$$L(x_j, y_j, s_j) := a - b(s_j - \mathcal{M}_E(x_j, y_j, I, h))^2.$$

where α is the reward coefficient that controls the exploration term of the reward function, and

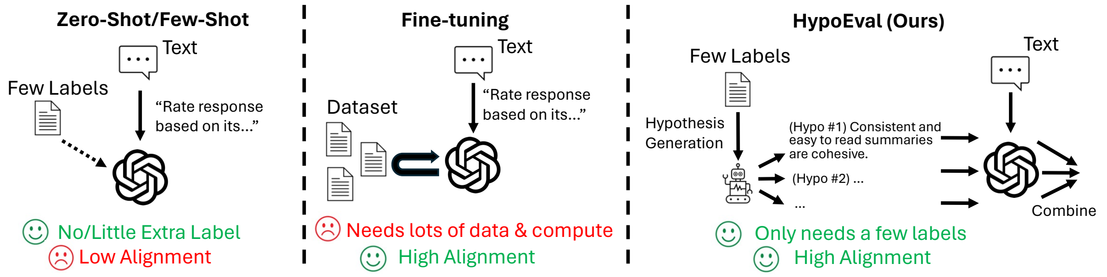

Figure 1: A Comparison between previous methods and HypoEval. We achieve high-alignment and explainable evaluation with only a few human labels per dataset.

a, b are coefficients that control the range of the exploitation term.

In the update stage, we iterate over all data points in $S_{\text{update}} = S_{\text{tr}} \setminus S_{\text{init}}$. For time t, we consider the training sample $(x_t, y_t, s_t) \in S_{\text{update}}$. We first choose the top k hypotheses $\mathcal{H}_{\text{top}}$ with the highest reward from $\mathcal{H}^{t-1}$. Then for each $h \in \mathcal{H}_{\text{top}}$, we update the reward with:

$$r_h := \frac{1}{|S_h^t|} \sum_{(x_j, y_j, s_j) \in S_h^t} L(x_j, y_j, s_j) + \alpha \sqrt{\frac{\log(t + |S_{\text{init}}|)}{|S_h^t|}},$$

where S_h^t is the set of training samples seen by hypothesis h at time t.

For all hypotheses from $\mathcal{H}_{\text{top}}$, if at least w_{hyp} predicted a score with $|\mathcal{M}_E(x_t, y_t, I, h) - s_t| > \theta$, where θ is the threshold for identifying a wrong prediction, the datapoint (x_t, y_t, s_t) is added to a wrong sample bank $\mathcal{W}$. Once $|\mathcal{W}| \geq w_{\text{max}}$, a new set of hypotheses $\mathcal{H}_{\mathcal{W}}$ is generated using $\mathcal{W}$ and $\mathcal{L}$ by an iterative refinement process using a refinement model $\mathcal{M}_R$:

$$\mathcal{H}_{\mathcal{W}}^0 = \mathcal{M}_G(\mathcal{W}),$$
$$\mathcal{H}_{\mathcal{W}}^i, i > 0 = \begin{cases} \mathcal{M}_R(\mathcal{H}_{\mathcal{W}}^{i-1}, \mathcal{L}) & \text{if } i \bmod 2 = 0 \\ \mathcal{M}_R(\mathcal{H}_{\mathcal{W}}^{i-1}, \mathcal{W}) & \text{if } i \bmod 2 = 1. \end{cases}$$

The refinement finishes in N_{refine} rounds, and we get $\mathcal{H}_{\mathcal{W}} = \mathcal{H}_{\mathcal{W}}^{N_{\text{refine}}}$. The wrong sample bank $\mathcal{W}$ is then set to $\emptyset$. For $\mathcal{H}^t$, we choose H_{max} hypotheses with the highest reward from $\mathcal{H}_{\text{top}} \cup \mathcal{H}_{\mathcal{W}}$.

Following (Liu et al., 2025a), to accommodate for that literature-based hypotheses (potential hypotheses that are generated solely from summaries of relevant literature) can be undervalued during the update stage, we use a union approach to combine hypotheses from literature only $\mathcal{H}_{\mathcal{L}} = \mathcal{M}_G(\mathcal{L})$ and $\mathcal{H}^{|S_{\text{update}}|}$. Specifically, for a final hypothesis bank with size H_{max}, we first remove redundant hypotheses from $\mathcal{H}_{\mathcal{L}} = \mathcal{M}_G(\mathcal{L})$ and $\mathcal{H}^{|S_{\text{update}}|}$, and then randomly choose at most $\frac{H_{\text{max}}}{2}$ from each of them for the final hypothesis bank $\mathcal{H}$.

Hypothesis selection. Since that we are encouraging diverse and novel hypotheses in the hypothesis generation process by both the exploration term in reward and the incorporation of information from literature, it is possible that we also include some hypotheses that are interesting but are not suitable for a specific evaluation task. To accommodate this, we perform hypothesis selection from $\mathcal{H}$ based on the hypotheses' performance on S_{tr}.

Specifically, we choose the top H_{ev} hypotheses with the highest Pearson correlations with human scores on S_{tr}:

$$\mathcal{H}_{\text{ev}} = \arg \max_{\mathcal{H}' \subset \mathcal{H}, |\mathcal{H}'| = H_{\text{ev}}} \sum_{h \in \mathcal{H}'} r(h, S_{\text{tr}}),$$

where $r(h, S_{\text{tr}})$ is the Pearson correlation between the human scores and the scores given by the evaluator agent $\mathcal{M}_E$ based specifically on the decomposed dimension stated in h.

Hypothesis-guided text evaluation For each hypothesis h, we first evaluate the text based solely on the dimension entailed in h (e.g. logical structure of events for evaluating coherence of summaries) with chain-of-thought prompting (Wei et al., 2022). The evaluator $\mathcal{M}_E$ is asked to give a score rating between 1 and 5. Then, since different hypotheses in $\mathcal{H}_{\text{ev}}$ often entail different decomposed dimensions that are important for a holistic evaluation, we combine the scores on all hypotheses to acquire the final overall score for a sample:

$$\mathcal{M}_E(x, y, I, \mathcal{H}_{\text{ev}}) = \frac{1}{|\mathcal{H}_{\text{ev}}|} \sum_{h \in \mathcal{H}_{\text{ev}}} \mathcal{M}_E(x, y, I, h).$$

For more detailed information of the implementation, please refer to Appendix B.1.

Efficiency Analysis The computational complexity of HYPOEVAL can be separated into two parts: a preparation stage of hypothesis generation and selection, and an evaluation stage of hypothesis-guided automated evaluation. Let N be the total number of texts to be evaluated. For the preparation stage, the complexity of hypothesis generation can be expressed as $O(N_{\text{paper}} + (k + N_{\text{refine}})|S_{\text{tr}}| + H_{\text{max}}^2)$, where N_{paper} is the number of papers as relevant literature, and the complexity of hypothesis selection is $O(H_{\text{max}}|S_{\text{tr}}|)$. For the evaluation stage, computational complexity is $O(H_{\text{ev}}N)$. Under our setting where $S_{\text{tr}}, k, N_{\text{refine}}, N_{\text{paper}}, H_{\text{max}} \leq 30$ and $H_{\text{ev}} = 5$, the total complexity of HYPOEVAL can be expressed as $O(N)$ and is equivalent to other pointwise evaluators.

3 Experiment Setup

To demonstrate the effectiveness of our hypothesis-guided evaluation framework, we first compare our method with baselines on two NLG tasks with four datasets. We report both Spearman correlation and Pearson correlation to account for both alignment with human rankings and with human scores.

Tasks and datasets. We report two representative tasks, summarization and open-ended story generation, to evaluate our framework.

For the summarization task, we choose SummEval (Fabbri et al., 2021) and NewsRoom (Grusky et al., 2020) for our experiments. SummEval consists of 100 source texts, each with 16 summaries annotated on four aspects: coherence (CH), consistency (CON), fluency (FLU), and relevance (RE). We use the average of annotation scores of 3 human experts. NewsRoom has 60 source texts and 7 summaries for each text, and is annotated on four aspects: coherence (CH), informativeness (INF), fluency (FLU), and relevance (RE). For each dataset, we randomly sample 30 text-summary pairs and their human evaluation scores as training data, and perform automated evaluation on summaries of 40 texts for SummEval and summaries of 30 texts for NewsRoom, with a total of 640 and 210 summaries, respectively. We report summary-level Spearman and Pearson correlations.

For the open-ended story generation task, we use HANNA (Chhun et al., 2022) and part of WritingPrompt (WritingPrompt-A) with human annotations collected by (Chiang and yi Lee, 2023). HANNA includes 96 writing prompts, each with 11 stories annotated on 6 aspects: coherence (CH), complexity (CX), empathy (EM), engagement (EG), relevance (RE), and surprise (SU). WritingPrompt-A consists of 400 prompt-story pairs, annotated on grammaticality (GRA), cohesiveness (COH), likability (LIK), and relevance (RE). For both datasets, we choose 30 prompt-story pairs for training. For HANNA, we randomly select 60 prompts for testing and report story-level Spearman correlations and Pearson correlations. For WritingPrompt-A, we choose 300 prompt-story pairs for testing. Due to the lack of story batches grouped by the same prompts, we report dataset-level correlations.

Baselines and implementation. We largely characterize baselines into two categories: zero-shot evaluators that do not consult human evaluations, and data-augmented evaluators that utilize specific datasets or are trained on specific tasks.

For zero-shot evaluators, we include ROUGE-L (Lin, 2004), BERTScore (Zhang et al., 2020), G-Eval (Liu et al., 2023a) with probabilities and automatically generated chain-of-thought (CoT) prompting, and *direct scoring* (evaluator LLM assigns a score directly to a text) with CoT. We also consider PairS-beam (Liu et al., 2025b), a pairwise ranking evaluator, for comparison in Spearman correlation. To compare with other checklist-based approaches, we implemented CheckEval, (Lee et al., 2024) on own, because human-curated key components for each aspect were used in the original paper but not released. We leverage the LLM's prior knowledge to generate atomic checklists.

For data-augmented evaluators, we include UniEval (Zhong et al., 2022) and BARTScore (Yuan et al., 2021), which are task-specific evaluators trained with large corpora of data. We also fine-tune LLAMA-3.1-8B-INSTRUCT on each aspect of each dataset with 30 (FT-A) and 200 (FT-B) human-annotated data points. Due to the use of significantly larger amount of training data than HYPOEVAL, we regard UniEval, BARTScore, and FT-B as strong but not directly comparable methods. Furthermore, we consider direct scoring with few-shot demonstrations from human-annotated data. For the WritingPrompt-A dataset, due to its size (see Section 3) and the lack of references, we implement FT-B with 100 human evaluations and omit the performance of reference-based evaluators.

Implementation details of baselines are available in Appendix B.2.

Our framework works for any LLM $\mathcal{M}$. In the experiments, we utilize two models, GPT-4O-MINI (OpenAI, 2023) and LLAMA-3.3-70B-INSTRUCT (Dubey et al., 2024) to reflect a range of different model sizes. We abbreviate GPT-4O-MINI as GPT-MINI and LLAMA-3.3-70B-INSTRUCT as LLAMA-70B. For main experiments, we let $H_{\text{ev}} = 5$, and use the same LLM backbone for the evaluator model $\mathcal{M}_E$, the hypothesis generator model $\mathcal{M}_G$, and the refinement model $\mathcal{M}_R$.

4 Results

HYPOEVAL achieves highest correlations across most dataset-aspect configurations. Table 1 presents the main results across a total of 18 aspect-dataset settings. Comparing with baselines without large-scale tuning, HYPOEVAL with GPT-4O-MINI achieves state-of-the-art (SOTA) performance on 15 settings with Spearman correlation and 16 settings with Pearson correlation, on average outperforming G-Eval with CoT by 9.8% and 15.7% respectively; HYPOEVAL with LLAMA-3.3-70B-INSTRUCT achieves SOTA on 13 settings for Spearman correlation and 15 settings for Pearson correlation, outperforming G-Eval by 9.9% and 11.8%.

Some exceptions, such as the consistency and fluency aspects of SummEval, could be due to the human scores being highly skewed towards 5, illustrated in Appendix D.

In addition, though HYPOEVAL is not explicitly optimized for pairwise comparison, it still outperforms the ranking-based evaluator PairS-beam on 16/18 and 13/18 settings for GPT-4O-MINI and LLAMA-3.3-70B-INSTRUCT respectively.

Comparing with tuning-based evaluators that use at least more than 3 times more annotated data (FT-B, BARTScore, UniEval), HYPOEVAL still demonstrates strong performance. For the story generation task, HYPOEVAL with GPT-4O-MINI or LLAMA-3.3-70B-INSTRUCT outperforms FT-B across all settings. For the summarization task, HYPOEVAL on average outperforms BARTScore by 18.66%; after excluding the exceptions of the consistency and fluency aspects of SummEval, HYPOEVAL on average outperforms FT-B by 4.8%.

To further illustrate our method, we include examples of hypotheses in Table 2. As demonstrated by the coherence aspect of SummEval in the table, these different hypotheses cover different decomposed dimensions of what a human would consider. For example, the first hypothesis covers that the answer should be "logically organized, with a clear introduction, body, and conclusion"; the second hypothesis highlights the "consistent tone and style", and the third one contains that the answer should not introduce "unrelated themes or topics". We include full versions of more examples in Appendix C, and examples of hypotheses that are not selected together with their failure modes in Table 11 in the appendix.

Ablation Studies To evaluate the effectiveness of both the hypothesis generation stage and the hypothesis-guided evaluation stage, we conduct two ablation studies. We first study the performance of HYPOEVAL when hypotheses are generated solely from an LLM's prior knowledge. Specifically, we consider 0-shot hypothesis generation, where we directly prompt LLM $\mathcal{M}$ to generate H_{ev} hypotheses for evaluating specific aspects of a text generation task, and then perform hypothesis-guided text evaluation.

We also study the effectiveness of the hypothesis-guided evaluation stage that first uses different hypotheses to generate scores and then combines them with a checklist-like approach. Specifically, we test against a pipeline similar to Liu et al. (2023b), where we concatenated all H_{ev} hypotheses after the hypothesis selection stage into one single criterion and let the evaluator model directly assign scores on a given text based on the criterion.

As shown in Table 3, we observe performance drops in most settings when either the hypothesis generation stage or the hypothesis-guided evaluation stage is removed. On average across both models, all settings, and both correlations, replacing the hypothesis generation stage by 0-shot generation drops performance by 7.25%, replacing the hypothesis-guided evaluation stage with single criterion drops the performance by 8.19%.

Additionally, we also conduct a smaller-scale ablation study by deploying HYPOEVAL with hypotheses generated solely from the summaries of relevant literature without the human evaluation scores. As shown in Table 7, we observe performance drops across 5 different dataset-aspect configurations.

Out-of-distribution generalizability. In this section, we will demonstrate the out-of-distribution (OOD) generalizability of HYPOEVAL on different datasets. We conducted a cross-dataset study

Models	Methods	SummEval								NewsRoom							
		CH		CON		FLU		RE		CH		INF		FLU		RE	
		ρ	r	ρ	r	ρ	r	ρ	r	ρ	r	ρ	r	ρ	r	ρ	r
Other	ROUGE-L	0.10	0.12	0.14	0.18	0.09	0.12	0.19	0.20	0.08	-0.11	0.08	-0.08	0.07	-0.10	0.08	-0.06
	BERTScore	0.26	0.27	0.21	0.21	0.18	0.17	0.38	0.40	0.31	0.18	0.31	0.20	0.33	0.17	0.28	0.18
	BARTScore	0.47	0.50	0.26	0.25	0.25	0.25	0.35	0.35	0.66	0.72	0.59	0.75	0.64	0.70	0.56	0.74
	UniEval	0.56	0.51	0.47	0.63	0.39	0.47	0.43	0.42	–	–	–	–	–	–	–	–
	FT-A	0.47	0.48	0.38	0.40	0.31	0.39	0.40	0.40	0.50	0.51	0.53	0.56	0.58	0.59	0.48	0.56
	FT-B	0.57	0.59	0.57	0.61	0.46	0.57	0.46	0.47	0.61	0.62	0.69	0.72	0.61	0.63	0.61	0.71
GPT-MINI	direct scoring	0.50	0.49	**0.54**	0.62	0.22	0.23	0.47	0.51	0.51	0.51	0.47	0.46	0.59	0.59	0.46	0.44
	few-shot scoring	0.37	0.38	0.47	0.49	0.33	0.35	0.42	0.44	0.60	0.61	0.57	0.62	0.65	0.67	0.53	0.60
	G-Eval	0.54	0.54	0.51	**0.65**	0.31	0.30	0.47	0.55	0.59	0.53	0.58	0.52	0.63	0.62	0.51	0.44
	PairS-beam	0.52	–	0.53	–	0.31	–	0.49	–	0.53	–	0.61	–	0.43	–	0.55	–
	CheckEval	0.47	0.47	0.34	0.36	0.39	0.38	0.39	0.43	0.53	0.56	0.34	0.42	0.57	0.57	0.46	0.52
	HYPOEVAL	**0.58**	**0.58**	0.51	0.63	**0.40**	**0.45**	**0.54**	**0.58**	**0.64**	**0.69**	**0.62**	**0.75**	**0.67**	**0.69**	**0.60**	**0.78**
LLAMA-70B	direct scoring	0.56	0.57	0.56	0.64	0.33	0.34	0.47	0.50	0.51	0.48	0.46	0.51	0.58	0.53	0.43	0.48
	few-shot scoring	0.45	0.45	**0.61**	0.67	0.40	0.47	0.47	0.48	0.59	0.60	0.63	0.68	0.60	0.62	0.51	0.69
	G-Eval	0.61	0.57	0.51	**0.68**	**0.41**	**0.48**	0.52	0.49	0.53	0.57	0.57	0.59	0.54	0.55	0.51	0.65
	PairS-beam	0.60	–	0.54	–	0.37	–	0.50	–	0.58	–	**0.65**	–	0.58	–	**0.59**	–
	CheckEval	0.56	0.58	0.43	0.45	0.45	0.45	0.47	0.50	0.40	0.42	0.32	0.34	0.30	0.30	0.48	0.59
	HYPOEVAL	**0.63**	**0.63**	0.49	0.62	0.35	0.35	**0.54**	**0.56**	**0.62**	**0.65**	**0.65**	**0.74**	**0.65**	**0.64**	0.52	**0.73**

Models	Methods	HANNA												WritingPrompt-A							
		CH		CX		EM		EG		RE		SU		GRA		COH		LIK		RE	
		ρ	r	ρ	r	ρ	r	ρ	r	ρ	r	ρ	r	ρ	r	ρ	r	ρ	r	ρ	r
Other	ROUGE-L	0.17	0.23	0.26	0.29	0.15	0.16	0.21	0.24	0.10	0.09	0.16	0.17	–	–	–	–	–	–	–	–
	BERTScore	0.30	0.36	0.42	0.48	0.28	0.31	0.34	0.39	0.17	0.18	0.26	0.26	–	–	–	–	–	–	–	–
	BARTScore	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
	UniEval	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
	FT-A	0.40	0.46	0.37	0.42	0.32	0.37	0.35	0.41	0.43	0.48	0.17	0.20	0.29	0.28	0.34	0.30	0.22	0.22	0.53	0.52
	FT-B	0.40	0.45	0.52	0.58	0.38	0.43	0.44	0.51	0.46	0.52	0.33	0.39	0.37	0.36	0.51	0.49	0.36	0.36	0.59	0.58
GPT-MINI	direct scoring	0.47	0.59	0.51	0.54	0.35	0.41	0.49	0.56	0.48	0.59	0.37	0.45	0.47	0.40	0.56	0.51	0.41	0.38	0.63	0.63
	few-shot scoring	0.47	0.55	0.44	0.47	0.40	0.47	0.46	0.51	0.42	0.51	0.36	0.42	0.40	0.41	0.55	0.54	0.42	0.39	0.65	0.65
	G-Eval	0.48	0.62	0.53	0.56	0.41	0.47	0.46	0.55	0.49	0.61	**0.38**	**0.48**	0.53	0.48	0.58	0.55	0.43	0.40	0.66	0.66
	PairS-beam	0.39	–	0.51	–	0.43	–	0.48	–	0.39	–	**0.38**	–	0.20*	–	0.57	–	0.48	–	0.09*	–
	CheckEval	0.46	0.55	0.40	0.41	0.37	0.40	0.46	0.50	**0.52**	0.57	0.26	0.27	0.17	0.17	0.58	0.59	0.29	0.30	0.67	0.67
	HYPOEVAL	**0.55**	**0.67**	**0.55**	**0.61**	**0.50**	**0.57**	**0.54**	**0.63**	0.49	**0.62**	**0.38**	0.45	**0.54**	**0.53**	**0.64**	**0.60**	**0.53**	**0.52**	**0.70**	**0.68**
LLAMA-70B	direct scoring	0.49	0.59	0.45	0.46	0.40	0.46	0.42	0.48	0.46	0.55	0.30	0.33	0.45	0.44	0.58	0.58	0.27	0.24	0.63	0.63
	few-shot scoring	0.52	0.62	0.54	0.56	0.43	0.46	0.48	0.51	0.44	0.50	0.35	0.35	0.42	0.40	0.59	0.57	0.37	0.36	0.62	0.61
	G-Eval	0.53	0.65	0.49	0.53	0.37	0.44	0.45	0.52	0.48	0.57	0.37	0.44	**0.47**	**0.45**	0.61	0.61	0.24	0.19	0.65	0.63
	PairS-beam	0.47	–	0.55	–	0.46	–	0.46	–	0.46	–	0.42	–	0.26*	–	0.57	–	0.49	–	0.04*	–
	CheckEval	0.38	0.51	0.40	0.39	0.45	0.48	0.38	0.47	0.48	0.57	0.24	0.23	0.41	0.37	0.63	0.62	0.50	0.48	0.66	0.66
	HYPOEVAL	**0.54**	**0.67**	**0.56**	**0.66**	**0.47**	**0.54**	**0.52**	**0.60**	**0.51**	**0.63**	0.40	**0.50**	0.44	0.41	**0.63**	**0.62**	**0.53**	**0.51**	**0.69**	**0.68**

Table 1: Evaluation results of GPT-4O-MINI and LLAMA-3.3-70B-INSTRUCT. We report Spearman correlation (ρ) and Pearson correlation (r) for a total of 18 aspects of the 4 datasets. Some especially lower performance of PairS-beam marked with * is due to that the model frequently failed to generate pairwise preferences.

that uses hypotheses generated from one dataset on another (OOD) dataset of the same task.

Specifically, for the summarization task, we use hypotheses generated from SummEval to perform hypothesis-guided evaluation for NewsRoom and vice versa on 3 aspects: CH, FLU, and RE. For the story generation task, we use hypotheses generated from HANNA to perform evaluation for WritingPrompt-A and vice versa on CH or COH, and RE. As shown in Table 4, the hypotheses generated from one dataset can be effectively used for hypothesis-guided evaluation on an OOD dataset of the same task, with an average performance change of less than 1% for both models.

Cross-model generalizability. To further assess the generalizability of the hypotheses, we conduct a cross-model study, where hypotheses generated by one model $\mathcal{M}_G$ are used for evaluation by another model $\mathcal{M}_E$. Results are shown in Table 5 in the Appendix. On average, for hypotheses generated by GPT-4O-MINI, changing the evaluator model to LLAMA-3.3-70B-INSTRUCT leads to a 2.0% drop in performance, still outperforming the baselines; for LLAMA-3.3-70B-INSTRUCT as the generator model, changing the evaluator model to GPT-4O-MINI increases the performance by 1.37%. This shows that hypotheses can be effectively transferred to different evaluator models.

Example Hypotheses (Decomposed Dimensions) on SummEval - CH
- The overall structure and organization of the summary play a vital role in determining coherence scores. Summaries that are logically organized, with a clear introduction, body, and conclusion, will score higher (4 or 5), while those . . . - Summaries that maintain a consistent tone and style throughout will be rated higher for coherence (4 or 5), as this consistency aids in reader comprehension. In contrast, . . . - The thematic consistency of a summary is essential for achieving higher coherence scores. A summary that introduces multiple unrelated themes or topics, resulting in confusion and lack of focus, would likely receive a score of one. . . .

Example Hypotheses (Decomposed Dimensions) on HANNA - EG
- The originality and creativity of the story's premise and execution are crucial for engagement. A score of 1 is given to stories that are entirely derivative, relying on clichés and predictable plots . . . - The clarity and coherence of the narrative structure will significantly affect engagement scores. A score of 1 will be assigned to stories that are chaotic and incoherent . . . - Stories that are overly simplistic and fail to follow the prompt effectively will receive a score of 1, while those that showcase original ideas and a compelling narrative voice will receive a score of 5.

Table 2: Example hypotheses for the coherence (CH) aspect of SummEval and the engagement (EG) aspect of HANNA, generated by GPT-4O-MINI. Each hypothesis is formulated as an evaluation rubric on a specific decomposed dimension for the aspect.

		SummEval								NewsRoom							
Models	Methods	CH		CON		FLU		RE		CH		INF		FLU		RE	
		ρ	r	ρ	r	ρ	r	ρ	r	ρ	r	ρ	r	ρ	r	ρ	r
GPT-MINI	HYPOEVAL	0.58	0.58	0.51	0.62	0.40	0.45	0.54	0.58	0.64	0.69	0.62	0.75	0.67	0.69	0.60	0.78
	0-shot generation	0.55 ↓	0.56 ↓	0.49 ↓	0.58 ↓	0.37 ↓	0.40 ↓	0.54 =	0.56 ↓	0.59 ↓	0.64 ↓	0.63 ↑	0.71 ↓	0.58 ↓	0.61 ↓	0.57 ↓	0.72 ↓
	Single criterion	0.52 ↓	0.52 ↓	0.50 ↓	0.54 ↓	0.32 ↓	0.35 ↓	0.47 ↓	0.50 ↓	0.61 ↓	0.63 ↓	0.64 ↑	0.70 ↓	0.60 ↓	0.59 ↓	0.62 ↑	0.72 ↓
LLAMA-70B	HYPOEVAL	0.63	0.63	0.49	0.62	0.35	0.35	0.54	0.56	0.62	0.65	0.65	0.74	0.65	0.64	0.52	0.73
	0-shot generation	0.62 ↓	0.64 ↑	0.51 ↑	0.67 ↑	0.24 ↓	0.24 ↓	0.47 ↓	0.50 ↓	0.60 ↓	0.65 =	0.63 ↓	0.75 ↑	0.51 ↓	0.50 ↓	0.50 ↓	0.73 =
	Single criterion	0.50 ↓	0.51 ↓	0.48 ↓	0.53 ↓	0.37 ↑	0.41 ↑	0.46 ↓	0.48 ↓	0.57 ↓	0.58 ↑	0.60 ↓	0.67 ↓	0.61 ↓	0.61 ↓	0.53 ↑	0.65 ↓

		HANNA												WritingPrompt-A							
Models	Methods	CH		CX		EM		EG		RE		SU		GRA		COH		LIK		RE	
		ρ	r	ρ	r	ρ	r	ρ	r	ρ	r	ρ	r	ρ	r	ρ	r	ρ	r	ρ	r
GPT-MINI	HYPOEVAL	0.55	0.68	0.55	0.61	0.50	0.57	0.54	0.63	0.49	0.62	0.38	0.45	0.54	0.53	0.64	0.60	0.53	0.52	0.70	0.68
	0-shot generation	0.46 ↓	0.55 ↓	0.46 ↓	0.41 ↓	0.47 ↓	0.40 ↓	0.51 ↓	0.50 ↓	0.58 ↑	0.57 ↓	0.41 ↑	0.27 ↓	0.54 =	0.54 ↑	0.62 ↓	0.57 ↓	0.45 ↓	0.43 ↓	0.71 ↑	0.70 ↑
	Single criterion	0.49 ↓	0.59 ↓	0.56 ↑	0.63 ↑	0.44 ↓	0.49 ↓	0.50 ↓	0.57 ↓	0.48 ↓	0.57 ↓	0.42 ↑	0.46 ↑	0.44 ↓	0.42 ↓	0.57 ↓	0.54 ↓	0.50 ↓	0.48 ↓	0.65 ↓	0.65 ↓
LLAMA-70B	HYPOEVAL	0.54	0.67	0.56	0.66	0.47	0.54	0.52	0.60	0.51	0.63	0.40	0.50	0.44	0.41	0.63	0.62	0.53	0.51	0.69	0.68
	0-shot generation	0.52 ↓	0.65 ↓	0.50 ↓	0.56 ↓	0.45 ↓	0.51 ↓	0.50 ↓	0.57 ↓	0.58 ↑	0.68 ↑	0.35 ↓	0.38 ↓	0.46 ↑	0.43 ↑	0.61 ↓	0.59 ↓	0.37 ↓	0.36 ↓	0.70 ↑	0.70 ↑
	Single criterion	0.49 ↓	0.60 ↓	0.54 ↓	0.63 ↓	0.43 ↓	0.51 ↓	0.47 ↓	0.55 ↓	0.50 ↓	0.60 ↓	0.42 ↑	0.51 ↑	0.39 ↓	0.35 ↓	0.59 ↓	0.58 ↓	0.51 ↓	0.50 ↓	0.66 ↓	0.66 ↓

Table 3: Evaluation results of the ablation studies. We use ↑ and ↓ to indicate performance changes relative to HYPOEVAL, where ↑ denotes an increase and ↓ a decrease, or = for no significant change.

Prompt robustness. As LLM-as-a-judge methods often exhibit high prompt sensitivity (Zhou et al., 2024a; Sclar et al., 2024), we analyze the robustness of HYPOEVAL to variations in evaluation instructions and compare it with direct scoring with automatic chain-of-thought prompting. We use GPT-4o (OpenAI, 2023) to generate 10 variations of the initial evaluation prompt in the main experiments for both HYPOEVAL and direct scoring. We then perform evaluation with GPT-4O-MINI on SummEval-CH and HANNA-EG to showcase prompt robustness for the two tasks (Fig. 2). HYPOEVAL shows significantly lower sensitivity to evaluation prompt variations on both settings and both meta-evaluation metrics, on average reducing the spread of Spearman correlation and Pearson correlation by 47.5% and 29.2%.

Scaling human data. We also conduct a preliminary investigation into the performance of HYPOEVAL with increased numbers of human scores on SummEval-CH and HANNA-EG. As shown in Table 6, we observe a stable increase in correlation numbers as we add more human scores for hypothesis generation and selection.

5 Related Work

LLMs as evaluators. Our work follows the extensive research line on utilizing language models to automatically evaluate natural language generations (Liu et al., 2023a; Fu et al., 2023; Chen et al., 2023; Chiang and yi Lee, 2023; Li et al., 2025). LLM evaluators are usually cheaper than human evaluations and have better alignment with human judgments than lexical metrics. GPTScore (Fu et al., 2023) utilizes generated pre-trained models and formulates automatic evaluation as a conditional generation task. G-Eval (Liu et al., 2023a) similarly uses pre-trained models but adopts a

Models	Methods	SummEval						NewsRoom						HANNA				WritingPrompt-A			
		CH		FLU		RE		CH		FLU		RE		CH/COH		RE		CH/COH		RE	
		ρ	r	ρ	r	ρ	r	ρ	r	ρ	r	ρ	r	ρ	r	ρ	r	ρ	r	ρ	r
GPT-MINI	IND HYPOEVAL	0.58	0.58	0.40	0.45	0.54	0.58	0.64	0.69	0.67	0.69	0.60	0.78	0.55	0.68	0.49	0.62	0.64	0.60	0.70	0.68
	OOD HYPOEVAL	0.59	0.61	0.42	0.44	0.53	0.56	0.63	0.68	0.70	0.73	0.60	0.77	0.56	0.68	0.54	0.65	0.64	0.59	0.68	0.66
LLAMA-70B	IND HYPOEVAL	0.63	0.63	0.35	0.35	0.54	0.56	0.62	0.65	0.65	0.64	0.52	0.73	0.54	0.67	0.51	0.63	0.63	0.62	0.69	0.68
	OOD HYPOEVAL	0.62	0.63	0.31	0.36	0.51	0.53	0.62	0.65	0.67	0.66	0.53	0.69	0.53	0.66	0.54	0.65	0.61	0.60	0.70	0.70

Table 4: Results for OOD generalizability study, where columns are the settings that HYPOEVAL evaluates on. IND HYPOEVAL refers to hypothesis generation using in-distribution (IND) training data, while OOD HYPOEVAL refers to using OOD data.

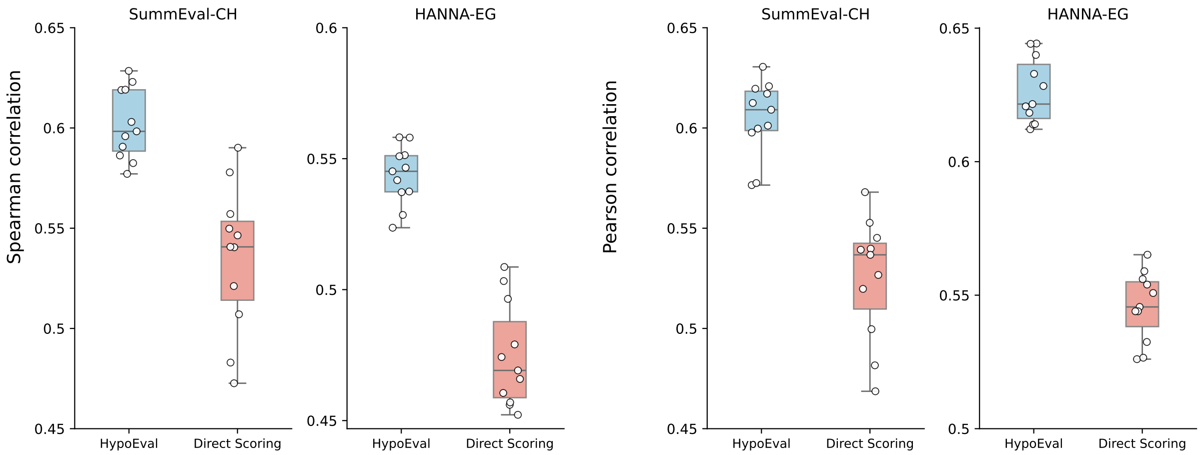

Figure 2: Results of prompt robustness study comparing HYPOEVAL with direct scoring, where each dot in the box plots refers to a specific prompt variation. HYPOEVAL shows significantly stronger robustness to evaluation prompts on representative evaluation settings.

prompt-based scoring approach. Liu et al. (2024b); Li et al. (2024a,b) consider calibration methods to mitigate the inference bias when using LLMs to assign scores. Specifically, Liu et al. (2024b) utilizes an approach similar to ours by prompting LLMs to generate scoring criteria from Monte-Carlo samples. However, their framework requires a much larger corpus of ground-truth samples (e.g. up to 188 samples for summarization) and there is no publicly available code. Alternatively, a significant amount of research has focused on automatic evaluation as a pairwise ranking problem (Qin et al., 2024b; Liusie et al., 2024). Liu et al. (2025b) develops an uncertainty-guided search method for ranking text generations, but is limited to an offline evaluation setting. Zhou et al. (2024a) introduces a prompt optimization framework that elicits both fairer preferences and better alignment with humans. However, pairwise ranking evaluation can face problems in terms of scalability, online evaluation, and cost or efficiency issues.

Checklist-based evaluation. Similar to our hypothesis-guided evaluation, where we aggregate scores from each decomposed dimension to acquire an overall score, there has also been previous research on aggregating evaluation results on atomic checklists for better correlation with human judgments. However, the checklist line of work mainly focuses on binary checklists where the answer is restricted to YES or NO. Tan et al. (2024); Que et al. (2024); Zhou et al. (2023); Qin et al. (2024a) use human-curated proxy questions, checklists, or "verifiable instructions" to benchmark LLMs' long-form text generation or instruction-following capabilities. Cook et al. (2024) explores automatic checklist generation by prompting with few-shot templates and shows effectiveness in evaluating instruction-following. Lee et al. (2024) and Pereira et al. (2024) further utilize the binary checklist method, but on average it does not yield better results than non-checklist methods like G-Eval.

6 Conclusion

We propose HYPOEVAL, a tuning-free and sample-efficient automated evaluation framework for natural language generation that achieves state-of-the-art performance in alignment with human evaluation rankings and scores. The generated hypotheses serve as decomposed dimensions of desiderate and provide interpretable explanations of the automated evaluation process. Through systematic studies, we show the robustness of HYPOEVAL to OOD data, prompt variations, and different evaluator models.

7 Limitations

To demonstrate the effectiveness of HypoEval as an automated evaluation framework, we conducted experiments on four datasets of general natural language generation tasks with GPT-4o-mini and Llama-3.3-70B-Instruct. However, we did not further test our framework on more domain-specific natural language generation tasks, such as legal summarization or clinical note generation, which could be addressed in future work. Also, though our model selection covers both big and small LLM sizes, we were unable to run our main experiments on models like GPT-4.1 or Claude Sonnet 4 due to budget limits.

Additionally, though we have already achieved satisfactory performance with HypoEval using the same set of hyperparameters across both LLMs and the 4 datasets as stated in Appendix B, we did not conduct an exhaustive hyperparameter search, which could potentially further enhance the performance of our framework.

Moreover, we drew on both relevant literature and human-provided scores to generate our hypotheses. The primary focus of this work is to introduce the first hypothesis-guided automated evaluation framework, and our sampled human scores and selected literature served this purpose effectively. However, we did not conduct a comprehensive analysis of how the quality of the human scores or the literature might impact the generated hypotheses. We believe that future work exploring these influences could provide valuable insights for further improving the robustness of our framework.

References

Anna Bavaresco, 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. Llms instead of human judges? a large scale empirical study across 20 nlp evaluation tasks. *Preprint*, arXiv:2406.18403.

Yi Chen, Rui Wang, Haiyun Jiang, Shuming Shi, and Ruifeng Xu. 2023. Exploring the use of large language models for reference-free text quality evaluation: An empirical study. *Preprint*, arXiv:2304.00723.

Cyril Chhun, Pierre Colombo, Chloé Clavel, and Fabian M. Suchanek. 2022. Of human criteria and automatic metrics: A benchmark of the evaluation of story generation. *Preprint*, arXiv:2208.11646.

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

Jonathan Cook, Tim Rocktäschel, Jakob Foerster, Dennis Aumiller, and Alex Wang. 2024. Ticking all the boxes: Generated checklists improve llm evaluation and generation. *Preprint*, arXiv:2410.03608.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, and et al. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Alexander R. Fabbri, Wojciech Kryściński, Bryan McCann, Caiming Xiong, Richard Socher, and Dragomir Radev. 2021. Summeval: Re-evaluating summarization evaluation. *Preprint*, arXiv:2007.12626.

Jiangnan Fang, Cheng-Tse Liu, Jieun Kim, Yash Bhedaru, Ethan Liu, Nikhil Singh, Nedim Lipka, Puneet Mathur, Nesreen K. Ahmed, Franck Dernoncourt, Ryan A. Rossi, and Hanieh Deilamsalehy. 2024. Multi-llm text summarization. *Preprint*, arXiv:2412.15487.

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

Max Grusky, Mor Naaman, and Yoav Artzi. 2020. Newsroom: A dataset of 1.3 million summaries with diverse extractive strategies. *Preprint*, arXiv:1804.11283.

Jiawei Gu, Xuhui Jiang, Zhichao Shi, Hexiang Tan, Xuehao Zhai, Chengjin Xu, Wei Li, Yinghan Shen, Shengjie Ma, Honghao Liu, Saizhuo Wang, Kun Zhang, Yuanzhuo Wang, Wen Gao, Lionel Ni, and Jian Guo. 2025. A survey on llm-as-a-judge. *Preprint*, arXiv:2411.15594.

Kalpesh Krishna, Aurko Roy, and Mohit Iyyer. 2021. Hurdles to progress in long-form question answering. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4940–4957, Online. Association for Computational Linguistics.

Michael Krumdick, Charles Lovering, Varshini Reddy, Seth Ebner, and Chris Tanner. 2025. No free labels: Limitations of llm-as-a-judge without human grounding. *Preprint*, arXiv:2503.05061.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*.

Yukyung Lee, Joonghoon Kim, Jaehee Kim, Hyowon Cho, and Pilsung Kang. 2024. Checkeval: Robust evaluation framework using large language model via checklist. *Preprint*, arXiv:2403.18771.

Dawei Li, Bohan Jiang, Liangjie Huang, Alimohammad Beigi, Chengshuai Zhao, Zhen Tan, Amrita Bhattacharjee, Yuxuan Jiang, Canyu Chen, Tianhao Wu, Kai Shu, Lu Cheng, and Huan Liu. 2025. From generation to judgment: Opportunities and challenges of llm-as-a-judge. *Preprint*, arXiv:2411.16594.

Haitao Li, Junjie Chen, Qingyao Ai, Zhumin Chu, Yujia Zhou, Qian Dong, and Yiqun Liu. 2024a. Calibraeval: Calibrating prediction distribution to mitigate selection bias in llms-as-judges. *Preprint*, arXiv:2410.15393.

Zongjie Li, Chaozheng Wang, Pingchuan Ma, Daoyuan Wu, Shuai Wang, Cuiyun Gao, and Yang Liu. 2024b. Split and merge: Aligning position biases in llm-based evaluators. *Preprint*, arXiv:2310.01432.

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

Haokun Liu, Yangqiaoyu Zhou, Mingxuan Li, Chenfei Yuan, and Chenhao Tan. 2025a. Literature meets data: A synergistic approach to hypothesis generation. *Preprint*, arXiv:2410.17309.

Minqian Liu, Ying Shen, Zhiyang Xu, Yixin Cao, Eunah Cho, Vaibhav Kumar, Reza Ghanadan, and Lifu Huang. 2024a. X-eval: Generalizable multi-aspect text evaluation via augmented instruction tuning with auxiliary evaluation aspects. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 8560–8579, Mexico City, Mexico. Association for Computational Linguistics.

Yang Liu and Yiling Chen. 2016. A bandit framework for strategic regression. In *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc.

Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. 2023a. G-eval: Nlg evaluation using gpt-4 with better human alignment. *Preprint*, arXiv:2303.16634.

Yinhong Liu, Han Zhou, Zhijiang Guo, Ehsan Shareghi, Ivan Vulić, Anna Korhonen, and Nigel Collier. 2025b. Aligning with human judgement: The role of pairwise preference in large language model evaluators. *Preprint*, arXiv:2403.16950.

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

Yuxuan Liu, Tianchi Yang, Shaohan Huang, Zihan Zhang, Haizhen Huang, Furu Wei, Weiwei Deng, Feng Sun, and Qi Zhang. 2024b. Calibrating LLM-based evaluator. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 2638–2656, Torino, Italia. ELRA and ICCL.

Adian Liusie, Potsawee Manakul, and Mark J. F. Gales. 2024. Llm comparative assessment: Zero-shot nlg evaluation through pairwise comparisons using large language models. *Preprint*, arXiv:2307.07889.

Kyle Lo, Lucy Lu Wang, Mark Neumann, Rodney Kinney, and Daniel Weld. 2020. S2ORC: The semantic scholar open research corpus. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4969–4983, Online. Association for Computational Linguistics.

Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike Lewis, Wen tau Yih, Pang Wei Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2023. Factscore: Fine-grained atomic evaluation of factual precision in long form text generation. *Preprint*, arXiv:2305.14251.

OpenAI. 2023. GPT-4 technical report.

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

Jayr Pereira, Andre Assumpcao, and Roberto Lotufo. 2024. Check-eval: A checklist-based approach for evaluating text quality. *Preprint*, arXiv:2407.14467.

Yiwei Qin, Kaiqiang Song, Yebowen Hu, Wenlin Yao, Sangwoo Cho, Xiaoyang Wang, Xuansheng Wu, Fei Liu, Pengfei Liu, and Dong Yu. 2024a. Infobench: Evaluating instruction following ability in large language models. *Preprint*, arXiv:2401.03601.

Zhen Qin, Rolf Jagerman, Kai Hui, Honglei Zhuang, Junru Wu, Le Yan, Jiaming Shen, Tianqi Liu, Jialu Liu, Donald Metzler, Xuanhui Wang, and Michael Bendersky. 2024b. Large language models are effective text rankers with pairwise ranking prompting. *Preprint*, arXiv:2306.17563.

Haoran Que, Feiyu Duan, Liqun He, Yutao Mou, Wangchunshu Zhou, Jiaheng Liu, Wenge Rong, Zekun Moore Wang, Jian Yang, Ge Zhang, Junran Peng, Zhaoxiang Zhang, Songyang Zhang, and Kai Chen. 2024. Hellobench: Evaluating long text generation capabilities of large language models. *Preprint*, arXiv:2409.16191.

Melanie Sclar, Yejin Choi, Yulia Tsvetkov, and Alane Suhr. 2024. Quantifying language models' sensitivity to spurious features in prompt design or: How i learned to start worrying about prompt formatting. *Preprint*, arXiv:2310.11324.

Haochen Tan, Zhijiang Guo, Zhan Shi, Lu Xu, Zhili Liu, Yunlong Feng, Xiaoguang Li, Yasheng Wang, Lifeng Shang, Qun Liu, and Linqi Song. 2024. Proxyqa: An alternative framework for evaluating long-form text generation with large language models. *Preprint*, arXiv:2401.15042.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed Chi, Quoc V Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, volume 35, pages 24824–24837. Curran Associates, Inc.

xAI. 2025. Grok 3 beta — the age of reasoning agents. `https://x.ai/news/grok-3`. Accessed: 2025-03-27.

Binwei Yao, Ming Jiang, Tara Bobinac, Diyi Yang, and Junjie Hu. 2024. Benchmarking machine translation with cultural awareness. *Preprint*, arXiv:2305.14328.

Weizhe Yuan, Graham Neubig, and Pengfei Liu. 2021. Bartscore: Evaluating generated text as text generation. *Preprint*, arXiv:2106.11520.

Xiang Yue, Boshi Wang, Ziru Chen, Kai Zhang, Yu Su, and Huan Sun. 2023. Automatic evaluation of attribution by large language models. *Preprint*, arXiv:2305.06311.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert. *Preprint*, arXiv:1904.09675.

Ming Zhong, Yang Liu, Da Yin, Yuning Mao, Yizhu Jiao, Pengfei Liu, Chenguang Zhu, Heng Ji, and Jiawei Han. 2022. Towards a unified multidimensional evaluator for text generation. *Preprint*, arXiv:2210.07197.

Han Zhou, Xingchen Wan, Yinhong Liu, Nigel Collier, Ivan Vulić, and Anna Korhonen. 2024a. Fairer preferences elicit improved human-aligned large language model judgments. *Preprint*, arXiv:2406.11370.

Jeffrey Zhou, Tianjian Lu, Swaroop Mishra, Siddhartha Brahma, Sujoy Basu, Yi Luan, Denny Zhou, and Le Hou. 2023. Instruction-following evaluation for large language models. *Preprint*, arXiv:2311.07911.

Yangqiaoyu Zhou, Haokun Liu, Tejes Srivastava, Hongyuan Mei, and Chenhao Tan. 2024b. Hypothesis generation with large language models. In *Proceedings of the 1st Workshop on NLP for Science (NLP4Science)*, page 117–139. Association for Computational Linguistics.

A Prompts

We include some example prompts for both the hypothesis generation and the hypothesis-guided evaluation stages of HypoEval.

A.1 Summarization

```
Instruction Prompt
You are a helpful assistant for predicting what score on
<aspect>, between 1 to 5 (the higher the better), will a
summary of a passage receive when judged by human experts.
Given a set of summaries and their source texts, we want to
 generate hypotheses that are useful for predicting what
score on <aspect>, between 1 to 5 (the higher the better),
will a summary of a passage receive when judged by human
experts.
The definition of <aspect> is given by: <definition>

Using the given examples and relevant literatures, please
propose <num_hypotheses> possible hypotheses.
These hypotheses should identify specific patterns that
occur across the provided summaries.

Each hypothesis should be about a specific trait or
dimension that human experts considers when giving score on
 <aspect>.
Each hypothesis should clearly state that based on the
trait or dimension, what kind of summary would be given a
score of one, what kind of summary a score of two, what
kind of summary a score of three, what kind of summary a
score of four, and what kind of summary a score of five.

Generate them in the format of hypothesis1. [hypothesis],
hypothesis2. [hypothesis], ... hypothesis <num_hypotheses>.
 [hypothesis].
The hypotheses should analyze what are the traits of the
summaries human experts considers when giving a score of
one, two, three, four, or five.
Remember! when generating hypotheses, always put "
hypothesis1.", "hypothesis2.", etc. as your index, do not
just generate "1.", "2.", etc.

User Prompt
We have seen some summaries and their source texts,
together with their scores on <aspect> given by human
experts:
<observations>
Please generate hypotheses that are useful for predicting
what score on <aspect>, between 1 to 5 (the higher the
better), will a summary of a passage receive when judged by
 human experts.
The definition of <aspect> is given by: <definition>
Propose <num_hypotheses> possible hypotheses. Generate them
 in the format of hypothesis1. [hypothesis], hypothesis2. [
hypothesis], ... hypothesis <num_hypotheses>. [hypothesis].
Remember! when generating hypotheses, always put "
hypothesis1.", "hypothesis2.", etc. as your index, do not
just generate "1.", "2.", etc.
Proposed hypotheses:
```

Example 1: Hypothesis Generation.

```
Instruction Prompt
You are a helpful assistant for predicting what score on
<aspect>, between 1 to 5 (the higher the better), will a
summary of a passage receive when judged by human experts.
Given a set of summaries and their source texts, we want to
 generate hypotheses that are useful for predicting what
score on <aspect>, between 1 to 5 (the higher the better),
will a summary of a passage receive when judged by human
experts.
The definition of <aspect> is given by: <definition>

Using the given examples, refine the hypotheses provided.
The desired hypotheses should identify specific patterns
that occur across the provided summaries.

Each hypothesis should be about a specific trait or
dimension that human experts considers when giving score on
 <aspect>.
Each hypothesis should clearly state that based on the
trait or dimension, what kind of summary would be given a
score of one, what kind of summary a score of two, what
kind of summary a score of three, what kind of summary a
score of four, and what kind of summary a score of five.

Generate them in the format of hypothesis1. [hypothesis],
hypothesis2. [hypothesis], ... <num_hypotheses>. [
hypothesis].
The hypotheses should analyze what are the traits of the
summaries human experts considers when giving a score of
one, two, three, four, or five.
Remember! when generating hypotheses, always put "
hypothesis1.", "hypothesis2.", etc. as your index, do not
just generate "1.", "2.", etc.
User Prompt
We have seen some summaries and their source texts,
together with their scores on <aspect> given by human
experts:
<observations>
We have some hypotheses need to be refined:
<hypotheses>
Please refine these hypotheses to make them more specific
and useful for predicting what score on <aspect>, between 1
 to 5 (the higher the better), will a summary of a passage
receive when judged by human experts.
When refining the hypotheses, feel free to change the key
information or topic of a hypothesis based on the provided
prevailing patterns in data if you think it is necessary.
Generate the refined hypotheses in the format of
hypothesis1. [hypothesis], hypothesis2. [hypothesis], ...
hypothesis <num_hypotheses>. [hypothesis].
The refined hypotheses should analyze what are the traits
of the summaries human experts considers when giving a
score of one, two, three, four, or five.
Remember! when generating the refined hypotheses, always
put "hypothesis1.", "hypothesis2.", etc. as your index, do
not just generate "1.", "2.", etc.
Refined hypotheses:
```

Example 2: Hypothesis Refine with Data.

```
Instruction Prompt
You are a helpful assistant for predicting what score on
<aspect>, between 1 to 5 (the higher the better), will a
summary of a passage receive when judged by human experts.
Given a set of summaries and their source texts, we want to
 generate hypotheses that are useful for predicting what
score on <aspect>, between 1 to 5 (the higher the better),
will a summary of a passage receive when judged by human
experts.
The definition of <aspect> is given by: <definition>

Using the given relevant literatures, refine the hypotheses
 provided.
The desired hypotheses should identify specific patterns
that occur across the provided summaries.

Each hypothesis should be about a specific trait or
dimension that human experts considers when giving score on
 <aspect>.
Each hypothesis should clearly state that based on the
trait or dimension, what kind of summary would be given a
score of one, what kind of summary a score of two, what
kind of summary a score of three, what kind of summary a
score of four, and what kind of summary a score of five.

Generate them in the format of hypothesis1. [hypothesis],
hypothesis2. [hypothesis], ... hypothesis <num_hypotheses>.
 [hypothesis].
The hypotheses should analyze what are the traits of the
summaries human experts considers when giving a score of
one, two, three, four, or five.
Remember! when generating hypotheses, always put "
hypothesis1.", "hypothesis2.", etc. as your index, do not
just generate "1.", "2.", etc.
User Prompt
We have some key findings from a series of research papers
that might be useful for generating hypotheses:
<relevant_papers>
We have some hypotheses need to be refined:
<hypotheses>
Please refine these hypotheses to make them more specific
and useful for predicting what score on <aspect>, between 1
 to 5 (the higher the better), will a summary of a passage
receive when judged by human experts.
```

When refining the hypotheses, feel free to change the key information or topic of a hypothesis based on the provided prevailing patterns in data if you think it is necessary.
Generate the refined hypotheses in the format of hypothesis1. [hypothesis], hypothesis2. [hypothesis], ... hypothesis <num_hypotheses>. [hypothesis].
The refined hypotheses should analyze what are the traits of the summaries human experts considers when giving a score of one, two, three, four, or five.
Remember! when generating the refined hypotheses, always put "hypothesis1.", "hypothesis2.", etc. as your index, do not just generate "1.", "2.", etc.
Refined hypotheses:

Example 3: Hypothesis Refine with Literature.

Instruction Prompt

You are a helpful assistant for predicting what score on <aspect>, between 1 to 5 (the higher the better), will a summary of a passage receive when judged by human experts.

From past experiences, you learned two hypotheses that are useful for predicting what score on <aspect>, between 1 to 5 (the higher the better), will a summary of a passage receive when judged by human experts.

You need to determine if the two hypotheses are so similar to the level of "repeating hypotheses".
Finally, answer "yes" if the two hypotheses are repetitive and "no" if they are not.
Keep your answer short.

Give your final answer in the format of "Final answer: [answer]".

User Prompt

We have two hypotheses that need you to determine if they are repetitive:

<hypotheses>
Are these two hypotheses so similar to the level that they are repetitive? If the both of them can provide significantly more information than only one of them could, and the information is important and useful for predicting what score on <aspect>, between 1 to 5 (the higher the better), will a summary of a passage receive when judged by human experts, they should not be considered repetitive.

Note that adding specific examples does not count as " provide significantly more information".

Give a short explanation of your decision.
Then give your final answer in the format of "Final answer: [answer]".
Your answer:

Example 4: Check Hypothesis Repetition

Instruction Prompt

You are a helpful assistant in answering questions about a summary of a story.
You will be given the story, the summary, and a pattern that talks about a specific trait to evaluate the <aspect> of the summary.
You should be generous and not too strict when evaluating.
The definition of <aspect> is given by: <definition>.
Story: [story]
Summary: [summary]
Pattern: [hypothesis]
The pattern talks about a specific trait that is related to the summary's score on <aspect>.
You need to evaluate the summary based on the trait and the rubric that the pattern talks about.
You should give a score (ranging from 1 to 5) on that trait according to the rubric.
Give your final evaluation score in the format of {Final score: [your score]}.

User Prompt

Given story, summary, and pattern:
Story: <story>
Summary: <summary>
Pattern: <hypothesis>
The pattern talks about a specific trait that is related to the summary's score on <aspect>.
The definition of <aspect> is given by: <definition>
You need to evaluate the summary based on the trait and the rubric that the pattern talks about.
You should give a score (ranging from 1 to 5) on that trait according to the rubric.
Follow the steps and provide reasoning when giving your score.
Step 1: What is the trait that the pattern talks about?
Step 2: Based on the trait and the rubric provided in the pattern, how is the summary on the trait?
Step 3 (final answer): Based on the rubric and your evaluations in step 2, what should be the score of the summary on the trait?
You should be generous and not too strict when evaluating.
Give your final evaluation score in the format of {Final score: [your score]}.
Answer:

Example 5: Hypothesis-Guided Evaluation

A.2 Story Generation

Instruction Prompt

You are a helpful assistant for predicting what score on <aspect>, between 1 to 5 (the higher the better), will a written story of a given prompt receive when judged by human experts.
Given a set of stories and their prompts, we want to generate hypotheses that are useful for predicting what score on <aspect>, between 1 to 5 (the higher the better), will a written story of a given prompt receive when judged by human experts.
The definition of <aspect> is given by: <definition>

Using the given examples and relevant literatures, please propose <num_hypotheses> possible hypotheses.
These hypotheses should identify specific patterns that occur across the provided stories.

Each hypothesis should be about a specific trait or dimension that human experts considers when giving score on <aspect>.
Each hypothesis should clearly state that based on the trait or dimension, what kind of story would be given a score of one, what kind of story a score of two, what kind of story a score of three, what kind of story a score of four, and what kind of story a score of five.

Generate them in the format of hypothesis1. [hypothesis], hypothesis2. [hypothesis], ... hypothesis <num_hypotheses>. [hypothesis].
The hypotheses should analyze what are the traits of the stories human experts considers when giving a score of one, two, three, four, or five.
Remember! when generating hypotheses, always put " hypothesis1.", "hypothesis2.", etc. as your index, do not just generate "1.", "2.", etc.

User Prompt

We have some key findings from a series of research papers that might be useful for generating hypotheses:
<relevant_papers>
We have seen some stories and their prompts, together with their scores on <aspect> given by human experts:
<observations>
Please generate hypotheses that are useful for predicting what score on <aspect>, between 1 to 5 (the higher the better), will a written story of a given prompt receive when judged by human experts.
The definition of <aspect> is given by: <definition>
Propose <num_hypotheses> possible hypotheses. Generate them in the format of hypothesis1. [hypothesis], hypothesis2. [hypothesis], ... hypothesis <num_hypotheses>. [hypothesis].
Remember! when generating hypotheses, always put " hypothesis1.", "hypothesis2.", etc. as your index, do not just generate "1.", "2.", etc.
Proposed hypotheses:

Example 6: Hypothesis Generation.

Instruction Prompt

You are a helpful assistant for predicting what score on <aspect>, between 1 to 5 (the higher the better), will a written story of a given prompt receive when judged by

```
human experts.
Given a set of stories and their prompts, we want to generate hypotheses that are useful for predicting what score on <aspect>, between 1 to 5 (the higher the better), will a written story of a given prompt receive when judged by human experts.
The definition of <aspect> is given by: <definition>

Using the given examples, refine the hypotheses provided.
The desired hypotheses should identify specific patterns that occur across the provided stories.

Each hypothesis should be about a specific trait or dimension that human experts considers when giving score on <aspect>.
Each hypothesis should clearly state that based on the trait or dimension, what kind of story would be given a score of one, what kind of story a score of two, what kind of story a score of three, what kind of story a score of four, and what kind of story a score of five.

Generate them in the format of hypothesis1. [hypothesis], hypothesis2. [hypothesis], ... hypothesis <num_hypotheses>. [hypothesis].
The hypotheses should analyze what are the traits of the stories human experts considers when giving a score of one, two, three, four, or five.
Remember! when generating hypotheses, always put "hypothesis1.", "hypothesis2.", etc. as your index, do not just generate "1.", "2.", etc.
User Prompt
We have seen some stories and their prompts, together with their scores on <aspect> given by human experts:
<observations>
We have some hypotheses need to be refined:
<hypotheses>
Please refine these hypotheses to make them more specific and useful for predicting what score on <aspect>, between 1 to 5 (the higher the better), will a written story of a given prompt receive when judged by human experts.
When refining the hypotheses, feel free to change the key information or topic of a hypothesis based on the provided prevailing patterns in data if you think it is necessary.
Generate the refined hypotheses in the format of hypothesis1. [hypothesis], hypothesis2. [hypothesis], ... hypothesis <num_hypotheses>. [hypothesis].
The refined hypotheses should analyze what are the traits of the stories human experts considers when giving a score of one, two, three, four, or five.
Remember! when generating the refined hypotheses, always put "hypothesis1.", "hypothesis2.", etc. as your index, do not just generate "1.", "2.", etc.
Refined hypotheses:
```

Example 7: Hypothesis Refine with Data.

```
Instruction Prompt
You are a helpful assistant for predicting what score on <aspect>, between 1 to 5 (the higher the better), will a written story of a given prompt receive when judged by human experts.
Given a set of stories and their prompts, we want to generate hypotheses that are useful for predicting what score on <aspect>, between 1 to 5 (the higher the better), will a written story of a given prompt receive when judged by human experts.
The definition of <aspect> is given by: <definition>

Using the given relevant literatures, refine the hypotheses provided.
The desired hypotheses should identify specific patterns that occur across the provided stories.

Each hypothesis should be about a specific trait or dimension that human experts considers when giving score on <aspect>.
Each hypothesis should clearly state that based on the trait or dimension, what kind of story would be given a score of one, what kind of story a score of two, what kind of story a score of three, what kind of story a score of four, and what kind of story a score of five.

Generate them in the format of hypothesis1. [hypothesis], hypothesis2. [hypothesis], ... hypothesis <num_hypotheses>. [hypothesis].
The hypotheses should analyze what are the traits of the stories human experts considers when giving a score of one, two, three, four, or five.
Remember! when generating hypotheses, always put "hypothesis1.", "hypothesis2.", etc. as your index, do not just generate "1.", "2.", etc.
User Prompt
We have some key findings from a series of research papers that might be useful for generating hypotheses:
 <relevant_papers>
 We have some hypotheses need to be refined:
 <hypotheses>
 Please refine these hypotheses to make them more specific and useful for predicting what score on <aspect>, between 1 to 5 (the higher the better), will a written story of a given prompt receive when judged by human experts.
 When refining the hypotheses, feel free to change the key information or topic of a hypothesis based on the provided prevailing patterns in data if you think it is necessary.
 Generate the refined hypotheses in the format of hypothesis1. [hypothesis], hypothesis2. [hypothesis], ... hypothesis <num_hypotheses>. [hypothesis].
 The refined hypotheses should analyze what are the traits of the stories human experts considers when giving a score of one, two, three, four, or five.
 Remember! when generating the refined hypotheses, always put "hypothesis1.", "hypothesis2.", etc. as your index, do not just generate "1.", "2.", etc.
 Refined hypotheses:
```

Example 8: Hypothesis Refine with Literature.

```
Instruction Prompt
You are a helpful assistant for predicting what score on <aspect>, between 1 to 5 (the higher the better), will a written story of a given prompt receive when judged by human experts.
From past experiences, you learned two hypotheses that are useful for predicting what score on <aspect>, between 1 to 5 (the higher the better), will a written story of a given prompt receive when judged by human experts.
You need to determine if the two hypotheses are so similar to the level of "repeating hypotheses".
Finally, answer "yes" if the two hypotheses are repetitive and "no" if they are not.
Keep your answer short.
Give your final answer in the format of "Final answer: [answer]".

User Prompt
We have two hypotheses that need you to determine if they are repetitive:
<hypotheses>
Are these two hypotheses so similar to the level that they are repetitive? If the both of them can provide significantly more information than only one of them could, and the information is important and useful for predicting what score on <aspect>, between 1 to 5 (the higher the better), will a written story of a given prompt receive when judged by human experts, they should not be considered repetitive.
Note that adding specific examples does not count as "provide significantly more information".
Give a short explanation of your decision.
Then give your final answer in the format of "Final answer: [answer]".
Your answer:
```

Example 9: Check Hypothesis Repetition (for removing redundant hypotheses, we use this prompt for each pair of hypotheses)

```
Instruction Prompt
You are a helpful assistant in answering questions about a written story of a given prompt.
You will be given the prompt, the written story, and a pattern that talks about a specific trait to evaluate the <aspect> of the story.
You should be generous and not too strict when evaluating.
The definition of <aspect> is given by: <definition>.
Prompt: [prompt]
Story: [story]
Pattern: [hypothesis]
```

```
The pattern talks about a specific trait that is related to
 the story's score on <aspect>.
You need to evaluate the story based on the trait and the
rubric that the pattern talks about.
You should give a score (ranging from 1 to 5) on that trait
 according to the rubric.
Give your final evaluation score in the format of {Final
score: [your score]}.

User Prompt
Given prompt, story, and pattern:
Prompt: <prompt>
Story: <story>
Pattern: <hypothesis>
Note: the story may have been abruptly cut in the middle of
 a sentence. Please rate it as if they ended just before
the unfinished sentence.
The pattern talks about a specific trait that is related to
 the story's score on <aspect>.
The definition of <aspect> is given by: <definition>

You need to evaluate the story based on the trait and the
rubric that the pattern talks about.
You should give a score (ranging from 1 to 5) on that trait
 according to the rubric.

Follow the steps and provide reasoning when giving your
score.
Step 1: What is the trait that the pattern talks about?
Step 2: Based on the trait and the rubric provided in the
pattern, how is the story on the trait?
Step 3 (final answer): Based on the rubric and your
evaluations in step 2, what should be the score of the
story on the trait?
You should be generous and not too strict when evaluating.
Give your final evaluation score in the format of {Final
score: [your score]}.
Answer:
```

Example 10: Hypothesis-Guided Evaluation

B Implementation Details

B.1 Implementation Details of HypoEval and Experiments

To collect relevant literature information $\mathcal{L}$, we first prompt Grok 3 with DeepSearch (xAI, 2025) to search for relevant academic papers on the two evaluation tasks (summarization and story generation) and retrieve 15 and 10 papers, respectively. Then, we use S2ORC-doc2json (Lo et al., 2020) to convert the raw PDF files to a set of JSON files that contain the abstracts and main texts of the papers. Subsequently, the hypothesis generator model $\mathcal{M}_G$ is prompted to generate a summary for each JSON file. The summaries are then concatenated to get the relevant literature information $\mathcal{L}$ that is later used for hypothesis generation with data and literature.

Then in the hypothesis generation stage, we set the size of S_{init} to 5, $|\mathcal{H}^{\text{init}}| = 5$, $k = 10$, $\theta = 0.5$, $\alpha = 0.5$, $w_{\max} = 10$, $N_{\text{refine}} = 6$, and $H_{\max} = 20$. For the hyperparameters a, b of the reward, we let $a = 1$, $b = \frac{1}{16}$ to ensure that the exploitation term is bounded in $[0, 1]$.

For hypothesis-guided evaluation, we let $H_{\text{ev}} = 5$. Following the implementation of PairS (Liu et al., 2025b), we set Spearman or Pearson correlation to 1 if the human annotation scores for all candidate responses of a source text or prompt are the same.

For all experiments and additional studies, excluding the prompt robustness study, we run all methods on all settings with 3 seeds: 42, 2, 114514.

B.2 Implementation Details of Baselines

For reference-based baselines, we implement ROUGE-L-F1, BERTScore-recall with default model choice for English language, UniEval, and the bart-score-cnn-src-hypo version of BARTScore.

For fine-tuning Llama-3.1-8B-Instruct, for FT-A, we use the same training set S_{tr} as HypoEval; for FT-B, we further sample 170 data points for SummEval, NewsRoom, and HANNA or 70 data points for WritingPrompt-A from the remaining data points, excluding the test sets. We fine-tune the model for 20 epochs.

For direct scoring and G-Eval, following the setup of the original G-Eval paper (Liu et al., 2023a), we first let the evaluator model $\mathcal{M}_E$ generate chain-of-thought steps for evaluation, and then let $\mathcal{M}_E$ give evaluation scores of given texts. To acquire the probabilities for G-Eval, we directly retrieve token probabilities for Llama-3.3-70B-Instruct, and sample 20 times with temperature set to 1 for GPT-4o-mini.

For direct scoring with few-shot demonstrations, we set the number of demonstrations $k = 3$, and randomly sample annotated data points from S_{tr}.

For PairS-beam, we use the same hyperparameter setting across all settings, where we set beam_size $= 1$ and prob_gap $= 0.1$.

B.3 Licensing Details

For the datasets we use in this study, SummEval and HANNA are under MIT License. NewsRoom is licensed under the Dataset Usage Agreement with the Cornell Newsroom Summaries Team that allows for non-commercial research and educational purposes. The human evaluation scores we used for WritingPrompt are under the Apache License 2.0.

For the LLMs, GPT-4o-mini is a proprietary and not released under any open-source license, while Llama-3.3-70B-Instruct is released under the Llama 3.3 Community License Agreement.

Throughout our study, we find that we are in compliance with the licensing agreements of all the datasets and LLMs used in this work.

$\mathcal{M}_G$	$\mathcal{M}_E$	SummEval								NewsRoom							
		CH		CON		FLU		RE		CH		INF		FLU		RE	
		ρ	r	ρ	r	ρ	r	ρ	r	ρ	r	ρ	r	ρ	r	ρ	r
GPT-MINI	GPT-MINI	0.58	0.58	0.51	0.63	0.40	0.45	0.54	0.58	0.64	0.69	0.62	0.75	0.67	0.69	0.60	0.78
	LLAMA-70B	0.63	0.66	0.50	0.60	0.38	0.44	0.56	0.59	0.59	0.66	0.59	0.75	0.68	0.69	0.54	0.74
LLAMA-70B	LLAMA-70B	0.63	0.63	0.49	0.62	0.35	0.35	0.54	0.56	0.62	0.65	0.65	0.74	0.65	0.64	0.52	0.73
	GPT-MINI	0.56	0.55	0.50	0.61	0.40	0.41	0.52	0.56	0.65	0.70	0.60	0.71	0.66	0.68	0.58	0.75

$\mathcal{M}_G$	$\mathcal{M}_E$	HANNA												WritingPrompt-A							
		CH		CX		EM		EG		RE		SU		GRA		COH		LIK		RE	
		ρ	r	ρ	r	ρ	r	ρ	r	ρ	r	ρ	r	ρ	r	ρ	r	ρ	r	ρ	r
GPT-MINI	GPT-MINI	0.55	0.68	0.55	0.61	0.50	0.57	0.54	0.63	0.49	0.62	0.38	0.45	0.54	0.53	0.64	0.60	0.53	0.52	0.70	0.68
	LLAMA-70B	0.48	0.65	0.54	0.65	0.50	0.57	0.53	0.61	0.53	0.65	0.41	0.49	0.42	0.39	0.60	0.58	0.56	0.54	0.65	0.65
LLAMA-70B	LLAMA-70B	0.54	0.67	0.56	0.66	0.47	0.54	0.52	0.60	0.51	0.63	0.40	0.50	0.44	0.41	0.63	0.62	0.53	0.51	0.69	0.69
	GPT-MINI	0.56	0.69	0.54	0.61	0.49	0.57	0.54	0.62	0.49	0.60	0.40	0.49	0.51	0.50	0.62	0.59	0.54	0.53	0.71	0.71

Table 5: Results for cross-model study, where the hypotheses generated by one model are used for evaluation with different evaluator models.

B.4 Estimated Cost

The cost of the hypothesis generation stage of our framework with GPT-4O-MINI is around $0.1 for one dataset-aspect configuration. For LLAMA-3.3-70B-INSTRUCT, we run all experiments on 4 NVIDIA A100s with vLLM (Kwon et al., 2023) as the backend. The hypothesis generation stage takes less than 30 minutes.

For the hypothesis-guided evaluation stage, the cost of running our pipeline is dependent on the number of evaluated texts together with their length.

C Example Hypotheses

We include full versions of more examples of generated hypotheses for SummEval - coherence, HANNA - engagement, and NewsRoom - relevance in Table 8, Table 9, and Table 10.

D Additional Illustrations

To further show the exceptions discussed in Section 4, we include the histograms of human annotation score distribution for the consistency and fluency aspects of SummEval in Fig. 3.

Dataset - Aspect	$\lvert S_{\text{tr}}\rvert = 30$		$\lvert S_{\text{tr}}\rvert = 60$		$\lvert S_{\text{tr}}\rvert = 100$	
	ρ	r	ρ	r	ρ	r
SummEval - CH	0.58	0.58	0.60	0.61	0.61	0.62
HANNA - EG	0.54	0.63	0.57	0.65	0.58	0.66

Table 6: Evaluation results with increased number of human evaluation scores used for hypothesis generation. We show that HYPOEVAL's correlations with human scores steadily increase as we scale the number of human scores used.

Dataset - Aspect	HYPOEVAL		Literature-only	
	ρ	r	ρ	r
SummEval - CH	0.58	0.58	0.54	0.56
NewsRoom - RE	0.60	0.78	0.56	0.74
HANNA - CH	0.55	0.67	0.53	0.64
HANNA - EM	0.50	0.57	0.48	0.53
WritingPrompt - GRA	0.54	0.53	0.54	0.52

Table 7: Evaluation results of the additional ablation study using hypotheses generated solely from literature. We observe performance drops compared to HYPOEVAL across all 5 tested dataset-aspect configurations.

Example Hypotheses on SummEval - Coherence

- The overall structure and organization of the summary play a vital role in determining coherence scores. Summaries that are logically organized, with a clear introduction, body, and conclusion, will score higher (4 or 5), while those that lack a coherent structure or appear haphazardly arranged will score lower (1 or 2). A well-structured summary that guides the reader through the main points will likely receive a score of 5, while a disorganized summary will score a 1.
- Summaries that maintain a consistent tone and style throughout will be rated higher for coherence (4 or 5), as this consistency aids in reader comprehension. In contrast, summaries that shift in tone or style abruptly, creating confusion or distraction for the reader, will be rated lower (1 or 2), reflecting a lack of coherence and engagement.
- Summaries that are exceptionally coherent, well-structured, and articulate, effectively conveying the main ideas and integrating them in a way that enhances understanding, will receive a score of five.
- Summaries that are poorly structured, lack logical flow, and fail to connect ideas will receive a score of one, as they may be disjointed and confusing, making it difficult for readers to follow the main ideas.
- The thematic consistency of a summary is essential for achieving higher coherence scores. A summary that introduces multiple unrelated themes or topics, resulting in confusion and lack of focus, would likely receive a score of one. A summary that partially maintains a central theme but includes several irrelevant details or tangents that distract from the main point may receive a score of two. A summary that presents a clear main theme but lacks depth or thorough development of supporting ideas, leading to a somewhat superficial understanding, might score a three. A summary that effectively ties together related ideas around a central theme, providing a coherent narrative with some depth and relevant context, would receive a score of four. Finally, a summary that maintains a singular, well-developed theme throughout, seamlessly integrating all points and enhancing the overall message with rich context and insights, would receive a score of five.

Table 8: Full version of additional example hypotheses generated by GPT-4O-MINI for the coherence aspect of SummEval.

Example Hypotheses on HANNA - Engagement
• The originality and creativity of the story's premise and execution are crucial for engagement. A score of 1 is given to stories that are entirely derivative, relying on predictable plots without any unique elements. A score of 2 may indicate a story that includes a few original ideas but is largely uninspired and fails to captivate the reader. A score of 3 suggests a moderately creative premise that engages the reader but lacks depth or surprising twists. A score of 4 reflects a highly original story that captivates the audience with innovative concepts and engaging execution, while a score of 5 is reserved for stories that present unique, unexpected twists and thought-provoking insights that challenge the reader's expectations and provoke deeper reflection.
• The clarity and coherence of the narrative structure will significantly affect engagement scores. A score of 1 will be assigned to stories that are chaotic and incoherent, making them nearly impossible to follow; a score of 2 for stories that have a basic structure but are confusing or lack logical flow, resulting in a disjointed reading experience; a score of 3 for stories with a clear but simplistic structure that conveys the plot adequately but lacks depth; a score of 4 for stories that are well-structured, logically flowing, and maintain reader interest through effective transitions and a clear narrative arc; and a score of 5 for stories that exhibit a sophisticated and intricate structure that enhances the narrative, captivates the reader, and seamlessly integrates various plot elements, creating a compelling reading experience.
• Stories that are overly simplistic and fail to follow the prompt effectively will receive a score of 1, while those that showcase original ideas and a compelling narrative voice will receive a score of 5.
• Emotional resonance and the ability to evoke feelings in the reader are key factors in engagement scoring. Stories that fail to connect emotionally with the audience will likely receive a score of 1 or 2, while those that successfully elicit strong emotional reactions, such as joy, sadness, or suspense, will score higher (4 or 5) due to their impactful storytelling.
• The richness of character development is a key factor in determining engagement. A score of 1 is assigned to stories featuring flat, one-dimensional characters that fail to evoke any emotional connection or interest. A score of 2 may indicate characters that are somewhat developed but lack complexity and relatability, making it hard for readers to connect. A score of 3 suggests characters that are relatable but not fully fleshed out, leading to moderate engagement. A score of 4 reflects well-developed characters that enhance the overall engagement of the story, showcasing growth, complexity, and emotional depth. Conversely, stories with multi-dimensional, relatable characters that undergo meaningful development, face internal and external challenges, and elicit empathy from the reader will score a 5.

Table 9: Full version of additional example hypotheses generated by GPT-4o-mini for the engagement aspect of HANNA.

Example Hypotheses on NewsRoom - Relevance
• A summary will receive a score of 1 if it contains information that directly contradicts the source text, a score of 2 if it contains some information not present in the source text, a score of 3 if it contains a mix of information present and not present in the source text, a score of 4 if it contains most information present in the source text, but lacks nuance or depth, and a score of 5 if it only contains information present in the source text, has excellent coherence, clarity, and demonstrates a high level of depth and insight, with effective use of transitional phrases and sentences to connect ideas, and the summary is accurate and reliable.
• A summary will receive a score of 1 if it is completely unrelated to the source text, a score of 2 if it is partially related but contains significant inaccuracies, a score of 3 if it is partially related and contains some accurate information, but also some inaccuracies, a score of 4 if it is mostly related and contains mostly accurate information, and has good coherence and clarity, but misses some key points or lacks depth, and a score of 5 if it is entirely related to the source text, contains all accurate and key information, and demonstrates a high level of coherence, clarity, and depth, with clear and concise language, and effective use of rhetorical devices to engage the reader and convey complex ideas, and the summary is comprehensive and well-written.
• A summary will receive a score of 1 if it introduces a significant amount of new information not present in the source text, such as external knowledge or opinions, that alters the meaning or tone of the original text, a score of 2 if it introduces some new information but also includes some relevant details from the source text, a score of 3 if it includes a mix of relevant and irrelevant details with some inconsistencies, such as including information from other sources, a score of 4 if it includes mostly relevant details with minor errors or omissions, and a score of 5 if it only includes details that are present in the source text and are relevant to the main points, without any external information or opinions that could change the original meaning, based on the trait of relevance and presence of extraneous information, including the ability to distinguish between essential and non-essential information.
• A summary will receive a score of 1 if it fails to capture any key concepts or relationships presented in the source text, a score of 2 if it captures some key concepts but misses important relationships or nuances, a score of 3 if it captures most key concepts and relationships but with some inaccuracies or inconsistencies, a score of 4 if it accurately captures most key concepts and relationships with minor inaccuracies, and a score of 5 if it accurately and comprehensively captures all key concepts and relationships presented in the source text, including underlying themes, motivations, and implications, based on the trait of depth and quality of analysis, including the ability to identify and explain complex relationships, patterns, and concepts presented in the source text.

Table 10: Full version of additional example hypotheses generated by Llama-3.3-70B-Instruct for the relevance aspect of NewsRoom.

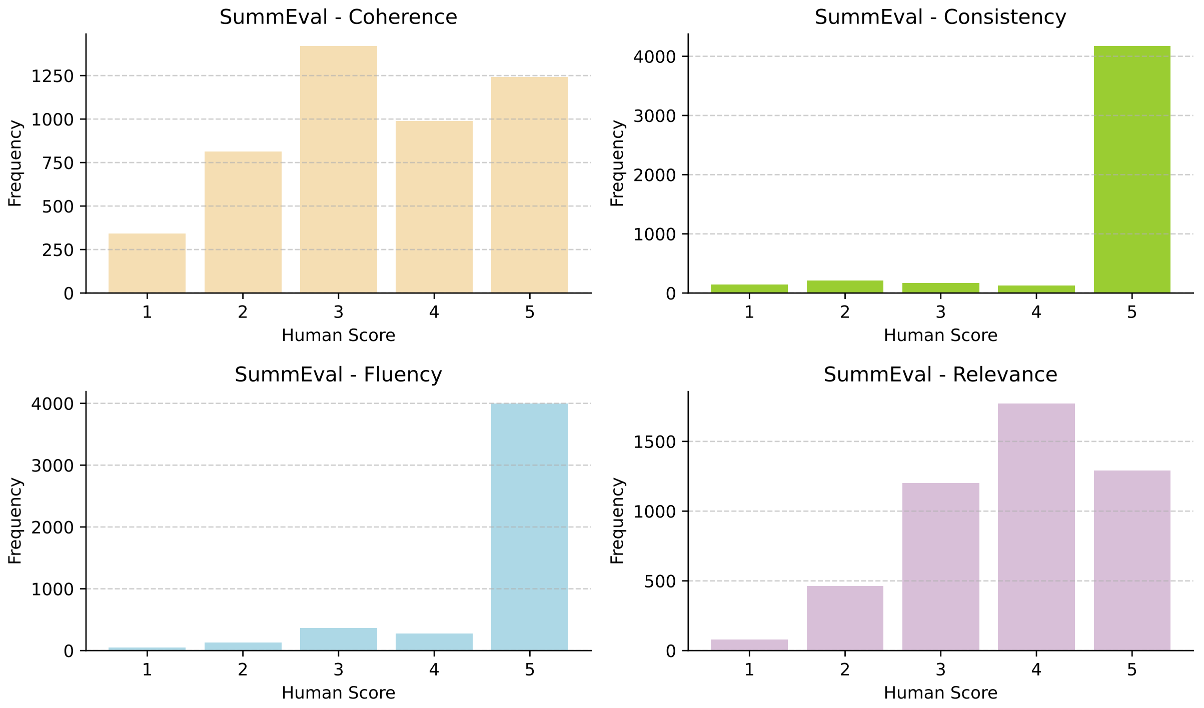

Figure 3: Illustration of the distribution of human evaluation scores of SummEval. The scores for the consistency and fluency aspects are highly skewed towards 5, which potentially leads to the decrease in performance of HYPOEVAL on theses aspects.

Example of not-selected hypothesis from NewsRoom - CH Failure Mode: The complexity of sentences does not necessarily influence coherence scores.
• A summary that uses varied sentence structures and vocabulary to enhance readability and engagement, while maintaining coherence throughout, will receive a score of 5. A summary that primarily uses simple sentences but maintains coherence and clarity will receive a score of 4. A summary that relies heavily on repetitive phrases or awkward constructions, leading to a lack of engagement and clarity, will receive a score of 3. A summary that is poorly written, with frequent grammatical errors or awkward phrasing that disrupts understanding, will receive a score of 2. A summary that is riddled with errors that make it nearly impossible to understand, severely impacting coherence, will receive a score of 1.
Example of not-selected hypothesis from NewsRoom - FLU Failure Mode: The hypothesis relies on irrelevant criteria (verbosity, informativeness rather than fluency).
• Summaries that are excessively verbose, contain irrelevant information, or fail to focus on the main ideas, making it hard for the reader to grasp the essential points, will receive a score of 1. Summaries that are somewhat concise but still include unnecessary details that detract from the main points and confuse the reader will receive a score of 2. Summaries that are mostly concise but may have a few extraneous details that do not significantly impact clarity, allowing for some understanding of the main ideas, will receive a score of 3. Summaries that are concise with only minor unnecessary details that do not detract from the overall message and maintain focus on the key points will receive a score of 4. Summaries that are succinct, focused, and contain only relevant information that enhances understanding and clarity will receive a score of 5.
Example of not-selected hypothesis from HANNA - EM Failure Mode: The hypothesis does not provide a comprehensive criteria (e.g. only talks about criteria for a score of 2)
• Stories rated with a score of 2 demonstrate minimal emotional engagement, where characters may be somewhat relatable but lack significant development or depth. The narrative may include basic emotional elements but fails to evoke strong feelings, resulting in a weak empathetic response from readers.

Table 11: Examples of hypotheses that are not selected during the hypothesis selection stage of HYPOEVAL. We also give failure modes describing why hypotheses are not selected.