Finding Blind Spots in Evaluator LLMs with Interpretable Checklists

Sumanth Doddapaneni*[1,2] Mohammed Safi Ur Rahman Khan*[1,2]
Sshubam Verma[1] Mitesh M. Khapra[1,2]

[1]Nilekani Centre at AI4Bharat [2]Indian Institute of Technology, Madras
Correspondence: {sumanthd, miteshk}@cse.iitm.ac.in, safikhan@ai4bharat.org

https://huggingface.co/datasets/ai4bharat/FBI
https://github.com/AI4Bharat/FBI

Abstract

Large Language Models (LLMs) are increasingly relied upon to evaluate text outputs of other LLMs, thereby influencing leaderboards and development decisions. However, concerns persist over the accuracy of these assessments and the potential for misleading conclusions. In this work, we investigate the effectiveness of LLMs as evaluators for text generation tasks. We propose FBI, a novel framework designed to examine the proficiency of Evaluator LLMs in assessing four critical abilities in other LLMs: factual accuracy, instruction following, coherence in long-form writing, and reasoning proficiency. By introducing targeted perturbations in answers generated by LLMs, that clearly impact one of these key capabilities, we test whether an Evaluator LLM can detect these quality drops. By creating a total of 2400 perturbed answers covering 22 perturbation categories, we conduct a comprehensive study using different evaluation strategies on five prominent LLMs commonly used as evaluators in the literature. Our findings reveal significant shortcomings in current Evaluator LLMs, which failed to identify quality drops in over 50% of cases on average. Single-answer and pairwise evaluations demonstrated notable limitations, whereas reference-based evaluations showed comparatively better performance. *These results underscore the unreliable nature of current Evaluator LLMs and advocate for cautious implementation in practical applications.* Code and data are available at https://github.com/AI4Bharat/FBI.

1 Introduction

Large Language Models (LLMs) are gaining widespread acceptance as the gold standard for evaluation in numerous applications, thanks to their efficiency and significant reductions in cost & time compared to human evaluators (Kim et al., 2023, 2024a; Chiang and Lee, 2023; Chen et al., 2023; Dubois et al., 2023). Furthermore, Evaluator LLMs are increasingly being utilized in the creation and maintenance of leaderboards for benchmarking various AI models (Watts et al., 2024; Zheng et al., 2023). While this reliance on LLMs offers significant advantages, it also presents potential drawbacks that warrant careful consideration. If LLMs are not effective evaluators, the resulting rankings and assessments could be fundamentally flawed, leading to inaccurate conclusions and misguided decisions. Therefore, it is crucial to pause and rigorously assess the evaluation capabilities of LLMs.

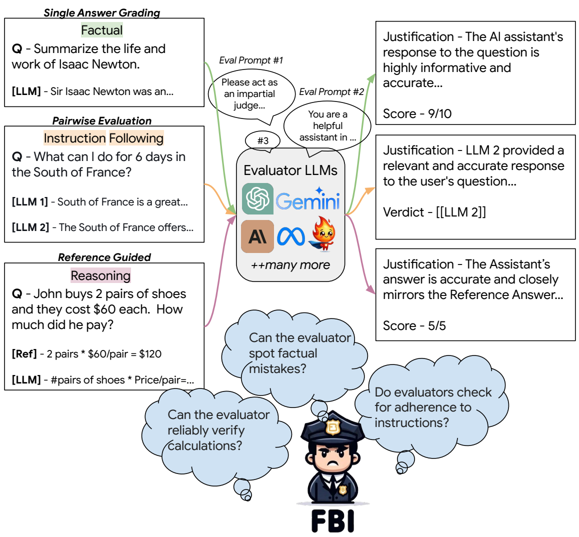

Figure 1: We present FBI, our novel meta-evaluation framework designed to assess the robustness of evaluator LLMs across diverse tasks and evaluation strategies.

Recent studies have explored the effectiveness of LLMs as evaluators and have reported strong correlations with human evaluations (Dubois et al., 2023; Zheng et al., 2023). While these findings are promising, accepting LLMs as reliable evaluators necessitates more nuanced assessments (Zeng et al., 2023). As LLMs become integral in a diverse range of tasks, they are expected to demonstrate a wide array of abilities, including factual accu-

*Equal Contribution.

racy, instruction following, coherence in long-form writing, and reasoning proficiency. Consequently, it is crucial to determine if Evaluator LLMs can indeed do a fine grained assessment of these varied abilities. Specifically, can they evaluate factual correctness, grammar, spelling, mathematical proficiency, and adherence to instructions in answers generated by other LLMs? (ref. Fig. 1) The necessity for such thorough fine-grained assessments is underscored by the Checklist (Ribeiro et al., 2020) approach, initially applied to BERT (Devlin et al., 2019) and subsequently adapted in studies across various tasks and models (Sai et al., 2021).

In this work, we introduce **FBI**, a comprehensive framework designed to **F**ind **B**lind spots in evaluator LLMs using an **I**nterpretable checklist across four fundamental text generation abilities: (a) factual accuracy, (b) instruction following, (c) coherence in long-form writing, and (d) reasoning proficiency. To rigorously assess an Evaluator LLM's ability to grade answers along these dimensions, we introduce perturbations that degrade the quality of the answer in one of these areas, expecting that good Evaluator LLMs will detect these quality drops and adjust their scores accordingly. Additionally, we develop quality-preserving perturbations where an Evaluator LLM should maintain consistent scoring. A detailed description of the 22 perturbation categories that we used is provided in Table 2. Staring with 500 prompts, we first generate long-form responses using GPT-4-TURBO. We then use a human-in-the-loop approach, to systematically perturb these responses, resulting in a dataset of 2400 tuples, where each tuple contains a prompt, response, and perturbed response.

Using the generated perturbations, we employed three evaluation paradigms (a) single-answer evaluation, (b) pairwise evaluation, and (c) reference-guided evaluation. Within each paradigm, we try multiple popular strategies of using Evaluator LLMs, such as, providing a rubric, asking for a justification, specifying the axis of evaluation, etc. Using these strategies, we assess the evaluation capabilities of five widely-used Evaluator LLMs. *Our findings indicate that LLMs are currently far from being reliable evaluators for text generation tasks*. Even with the best models and evaluation strategies, Evaluator LLMs failed to identify errors in over 50% of cases, on average. Interestingly, across all evaluation strategies, we observed that all popular Evaluator LLMs consistently performed poorly. Notably, even basic perturbation categories, such as, fluency perturbations (e.g. spellings and grammar) posed challenges for the evaluators. We also observed cases where Evaluator LLMs did not adjust their scores for perturbed responses despite correctly identifying the perturbations in their explanations. When used for single-answer grading and pairwise evaluation, Evaluator LLMs showed significant limitations, suggesting they are not reliable in these setups. In contrast, when used for reference-based evaluation, they demonstrated relatively better performance. Overall, our experiments uncovered significant blind spots in Evaluator LLMs, warranting caution in their direct application in practical settings.

2 Related Work

LLMs as Evaluators. LLMs have been increasingly used for automated evaluation for various NLG tasks (Wang et al., 2023a; Chiang and yi Lee, 2023; Kocmi and Federmann, 2023). We broadly classify this into two paradigms - (i) reference-driven evaluations (Fu et al., 2023; Kim et al., 2023), and (ii) reference-free evaluations (Liu et al., 2023; Zheng et al., 2023). The evaluator is either asked for a score (score-based evaluation) (Liu et al., 2023; Zheng et al., 2023; Hada et al., 2023) or to choose the best amongst two given responses (pairwise comparison evaluation) (Zheng et al., 2023; Wang et al., 2023b; Liusie et al., 2023). Additionally, various open-source evaluation-specific trained models have also been proposed (Wang et al., 2023d; Kim et al., 2023; Zhu et al., 2023). Further, advanced ensemble approaches include evaluation via multi-agent interactions (Chan et al., 2023; Zhang et al., 2023) or with external agents (Min et al., 2023; Hasanbeig et al., 2023).

Biases in Evalautor LLMs. Studies around Evaluator LLMs have highlighted the various biases - position bias (Zheng et al., 2023; Wang et al., 2023c), self preference bias (Panickssery et al., 2024; Liu et al., 2023), verbosity bias (Wu and Aji, 2023; Zeng et al., 2023), etc. Various approaches, including chain-of-thought reasoning (Zheng et al., 2023; Zeng et al., 2023), position-swapping (Zeng et al., 2023), among others, have been suggested to mitigate some of these. Recent studies (Hada et al., 2023; Saha et al., 2023) also show the effectiveness of the evaluators can be increased by evaluating specific axes and providing detailed rubrics/rules (Ye et al., 2023; Kim et al., 2024a).

Evaluation of Evaluator LLMs. Critically analysing evaluation metrics and suggesting methods to improve their robustness has always been of interest to the NLP community (Sai B et al., 2023; Mathur et al., 2020). Recent studies have evaluated the efficacy of LLMs as evaluators for specific types of tasks (Hada et al., 2024; Shen et al., 2023) and evaluation paradigms (Wang et al., 2023b,a) by assessing their agreement with human evaluations (Hada et al., 2023; Chiang and Lee, 2023; Zheng et al., 2023). Additionally, the robustness of these evaluators has been tested using adversarial examples (Kamoi et al., 2024; Chen et al., 2024; Wu and Aji, 2023), further showing their strengths and weaknesses.

Our proposed framework represents a significant departure from these existing approaches in several key aspects. First, we focus on a broader set of essential abilities: factual understanding, instruction following, long-form writing, and reasoning. Second, all prompts and the 2400 perturbed answers in our framework are carefully crafted and/or validated by humans, ensuring high quality and relevance to the abilities being evaluated. Third, our framework offers finer granularity in perturbation types, allowing us to finely identify and isolate the capabilities and limitations of Evaluator LLMs. This detailed analysis assists in making more knowledgeable choices about when to utilize LLMs as evaluators. Lastly, we focus on three popular evaluation paradigms, viz., reference-less single answer scoring, reference-less pairwise comparison, and reference based scoring, thereby providing a comprehensive toolkit for evaluating LLM performance across different dimensions.

3 FBI: Meta-Evaluation Checklist

We introduce FBI, a meta-evaluation benchmark designed to assess the capabilities of Evaluator LLMs in examining the outputs of other LLMs across four distinct task abilities: (i) *Factual* accuracy, (ii) *Reasoning* ability, (iii) *instruction* following, and (iv) proficiency in *long-form* writing. Each instance within the benchmark comprises a tuple $(I, A_{gold}, A_{perturb})$, where I represents the input instruction or prompt given to the model, A_{gold} denotes the correct or *gold* answer, and $A_{perturb}$ signifies a *perturbed* version of the gold answer. The perturbed answers, $A_{perturb}$, are generated by introducing specific types of errors across each of the four task abilities (Table 2) to evaluate whether LLM evaluators can accurately identify and account for these errors in the perturbed answers.

Category	# Instances
Long Form (LF)	**528**
Grammar	92
Spelling	100
Consistency	84
Chronology	71
Coherence	91
Comprehensiveness	90
Factual (F)	**483**
Contextual	94
Entity	87
Incorrect Fact	68
Number Errors	74
Opposite Fact	91
Remove Fact	69
Instruction Following (IF)	**381**
Do More	50
Do Less	100
Ignore Format	99
Sequence Errors	49
Assumptions	81
Reasoning (R)	**494**
Calculations	149
Copying Numbers	83
Final Errors	97
Incorrect Units	77
Wrong Formula	88
Score Invariant (SI)	**516**
Total	**2400**

Table 1: Statistics of perturbations across all the 4 task abilities and each of the perturbation categories.

The perturbations are based on perturbation categories carefully crafted by human annotators, informed by the prevalent failure modes in current LLMs (Min et al., 2023; Wu et al., 2023; Zhou et al., 2023b). These human annotators are graduate students who are well aware of the typical errors made by LLMs. Such human oversight is used throughout the benchmark's development, from prompt selection (§ 3.1) to defining perturbation categories (§ 3.2) and creating the perturbations (§ 3.3). To ensure a high standard of accuracy and reliability, all perturbations within FBI undergo rigorous manual vetting(§ 3.4). Table 1 presents some statistics about FBI, and the detailed generation process is discussed in the following sub-sections.

3.1 Prompt Selection

We selected six test sets containing prompts in English, viz., WizardLM (Xu et al., 2023), MT Bench (Zheng et al., 2023), UltraChat (Ding et al., 2023), LIMA (Zhou et al., 2023a), LLMBar (Zeng et al., 2023), and IFEval (Zhou et al., 2023b). These test sets were selected for their recency and because they contain prompts for long-form generation, creativity, and open-ended tasks that require instruction-following. Collectively, these test sets comprise of 1809 prompts. We manually categorized each prompt into one of the 4 task categories:

Long Form Writing (LF): These prompts require generating long pieces of text and explore generic topics, often including detailed analysis and storytelling. For example, *How can I improve my time management skills?*

Factual (F): These prompts seek objective information or facts. For example, *What is the primary function of a capacitor in an electrical circuit?*

Instruction Following (IF): These prompts require executing specific steps or guidelines to achieve a particular outcome or answer. For example, *Write a poem with* ***four*** *lines and the following words: peace, sky, race, ground.*

Reasoning (R): These prompts necessitate the application of logic, mathematics, and critical thinking to analyze information and draw conclusions. For example, *A bat and a ball together cost $1.10. The bat costs $1.00 more than the ball. How much does the ball cost?*

We sampled 100 questions from each of the four abilities, supplementing prompts requiring reasoning ability from the GSM8k (Cobbe et al., 2021) and MATH (Hendrycks et al., 2021) benchmarks. Additionally, we created 200 prompts tailored to instruction following to address specific perturbation categories[1]. The gold answers (A_{gold}) for all prompts were generated using the GPT-4-TURBO model. To ensure the quality and accuracy of A_{gold}, we conducted manual verification by randomly sampling 25% instances from each category and found that the gold answers maintain a high level of correctness. *Importantly, we emphasize that the quality of gold answers is not critical in our study, as our primary focus is on directional score changes* (i.e., we are interested in knowing if a perturbed answer with clear errors scores *relatively lower* than the original answer which did not have these errors).

3.2 Perturbation Categories

LLMs exhibit numerous failure modes, encompassing shortcomings in reasoning (Wu et al., 2023; Wei et al., 2022), factuality (Hu et al., 2024; Min et al., 2023), instruction-following (Zhou et al., 2023b; Li et al., 2023), and, in some instances, coherence and consistency (Naismith et al., 2023; Shen et al., 2023) in generated text. Given that we utilize Evaluator LLMs to assess responses in one or more of these abilities, it is imperative for the evaluator to excel in them. Our perturbations across each task ability are crafted keeping these failure modes in mind, as presented in Table 2. While our perturbations are primarily designed to decrease scores, we also develop score-invariant perturbations (§ 3.5), which are intended not to affect the score relative to the gold answer.

[1]Based on our categorization, we were unable to find a sufficient number of prompts in existing test sets to fit the perturbation categories.

3.3 Perturbation Generation

To generate perturbed answers ($A_{perturb}$) along each of the defined categories (§ 3.2), we use GPT-4-TURBO by prompting it with specific instructions tailored to each perturbation category. The model was tasked with producing perturbed answers and explaining the reasoning behind each perturbation. We iteratively refined the instructions by manually reviewing a sample of 25% of perturbed answers for each category, till we were satisfied with the generated perturbations.

3.4 Human-In-The-Loop

While GPT generally succeeds in generating the expected perturbations, we observed instances where the model (i) deviates from the intended perturbation, (ii) produces the incorrect style of perturbation, or (iii) accurately generates the reasoning but fails to reflect it in $A_{perturb}$. To address these inconsistencies, we meticulously vet all generated perturbations through a manual review process. Each perturbed answer produced by GPT-4-TURBO is examined against A_{gold}, and then categorized as valid, invalid, or score invariant. A perturbation is considered valid only if it should logically result in a scoring penalty as determined by human annotators. The vetting is carried out by students who possess a comprehensive understanding of LLM literature, holding at least a bachelor's or master's degree. To aid in validating perturbations, we developed a tool, the details of which are outlined in Appendix A.

3.5 Score-Invariant Perturbations

Score-invariant perturbations are those modifications that do not warrant a scoring penalty. These are collected in two ways: (i) human annotators

Task	Perturbation Axis	Description
LF	GRAMMAR	Introducing grammatical errors in the answer. Eg: This is good → This are good.
	SPELLING	Introducing "valid" spelling errors in the answer. Eg: Toxicity → Tocixity.
	CONSISTENCY	Introducing errors in the "consistency" of the answer (like tone, terminology, etc.)
	CHRONOLOGY	Introducing errors in the chronological or the logical flow of the answer.
	COHERENCE	Introducing errors that affect the coherence of the answer.
	COMPREHENSIVENESS	Introducing vagueness, irrelevance or lack of context in the answer.
F	CONTEXTUAL	Replacing fact with a contextually similar incorrect fact. Eg: electricity → magnetism.
	ENTITY	Replacing a named entity with an incorrect entity. Eg: Poland → London.
	INCORRECT FACT	Adding a new contextually relevant incorrect fact in the answer.
	NUMBER ERRORS	Introducing errors in the various numbers reported in the answer. Eg: 1987 → 1887.
	OPPOSITE FACT	Replacing a fact in the answer with its negation. Eg: ... will have ... → ... wont have ....
	REMOVE FACT	Removing a fact critical to the correctness and completeness of the answer.
IF	DO LESS	Doing less than what is *explicitly* requested in the question.
	DO MORE	Doing more than what is *explicitly* requested in the question.
	IGNORE FORMAT	Ignoring the formatting and other constraints mentioned in the question.
	SEQUENCE ERRORS	Ignoring the sequence in the response when *explicitly* requested in the instruction.
	ASSUMPTIONS	Making new incorrect assumptions about the instruction.
R	CALCULATIONS	Introducing calculation errors in the answer. Eg: $2+3=5 \rightarrow 2+3=6$
	COPYING NUMBERS	Introducing errors while considering the numbers mentioned in the instruction.
	FINAL ERRORS	Introducing errors only the final reported answer while retaining the correct solution.
	INCORRECT UNITS	Introducing errors in the units reported and considered in the answer.
	WRONG FORMULA	Introducing errors in the formula used in the answer. Eg: $\pi r^2 \rightarrow 2\pi r$
SI	SCORE INVARIANT	Introducing modifications in the answer which would not result in a score penalty.

Table 2: Perturbation categories across each of the task abilities. The green highlights indicate the original text and the red highlights indicated the perturbed text. Complete examples of each perturbation can be found in supplementary material.

categorize specific instances from our initial list as invariant (§ 3.4), and (ii) prompting GEMINI-1.5-PRO model to paraphrase A_{gold} ensuring retention of all original facts and details followed by human verification on a sample. We collect 516 score invariant perturbations in total.

4 Strategies for using Evaluator LLMs

In this section, we outline the prompting strategies employed by Evaluator LLMs benchmarked on FBI. An Evaluator LLM, $f(\cdot)$, takes the input instruction, LLM generated response and an evaluation prompt, P_{eval}, as input, and is required to generate a score and an optional explanation. To make the evaluation more robust, the evaluator may also be provided with additional information specifying the axes of evaluation, rubrics, rules, and other criteria. Our study focuses on 3 evaluation paradigms: (i) Single-answer scoring (§4.1), (ii) Pairwise comparison (§4.2), and (iii) Reference-guided evaluation (§4.3). For all the strategies evaluation prompts P_{eval} are adapted from Zheng et al. (2023); Zeng et al. (2023); Hada et al. (2023).

4.1 Single Answer Scoring

In this paradigm, evaluator $f(\cdot)$ is tasked with scoring a model response based solely on its parameterized knowledge.

Vanilla* (Zheng et al., 2023): In this strategy, the evaluator $f(\cdot)$ is presented with only the input instruction I and a model response A_{model}. The role of $f(\cdot)$ is to evaluate A_{model} and assign a score, denoted as $f(P_{eval}, I, A_{model}) \rightarrow (score)$.

Vanilla (Zheng et al., 2023): This strategy extends "Vanilla*", where the evaluator $f(\cdot)$ is tasked not only with scoring the model response A_{model} but also providing an explanation for the score - represented as $f(P_{eval}, I, A_{model}) \rightarrow (exp, score)$.

Rubric (Zeng et al., 2023): In this strategy, in addition to the instruction I and the model response A_{model}, we also provide a grading rubric R. The evaluator $f(\cdot)$ is prompted to first generate an explanation followed by a score- represented as $f(P_{eval}, R, I, A_{model}) \rightarrow (exp, score)$.

Axis (Hada et al., 2023): In this strategy, the evaluator $f(\cdot)$ is prompted to assess the model response, A_{model}, along a designated axis, Ax, aligning with the category of the instruction (§ 3.1).

For instance, factual questions are evaluated along the *hallucination* axis to determine the presence of fabricated content. This process is formally represented as $f(P_{eval}, Ax, I, A_{model}) \rightarrow (exp, score)$.

Axis+Rubric (Hada et al., 2023): In this strategy, the evaluator $f(\cdot)$ is provided with both a specific evaluation axis Ax and detailed scoring rubrics R for that axis. The is formally represented as $f(P_{eval}, Ax, R, I, A_{model}) \rightarrow (exp, score)$.

4.2 Pairwise Comparison

In this paradigm, evaluator $f(\cdot)$ is tasked to choose the better response from the two given options by again relying on its parameterized knowledge.

Pairwise* (Zheng et al., 2023): The evaluator $f(\cdot)$ here is given only an instruction I and two model responses A_1 and A_2 and is tasked to determine the better response or mark both as equally valid. This is formally represented as $f(P_{eval}, I, A_1, A_2) \rightarrow (verdict)$.

Pairwise (Zheng et al., 2023): This strategy extends "Pairwise*", where the evaluator is tasked not only with choosing the better response but also providing an explanation for the verdict - represented as $f(P_{eval}, I, A_1, A_2) \rightarrow (exp, verdict)$.

Rules (Zeng et al., 2023): In this strategy, in addition to the instruction I and the two model responses A_1, A_2, the evaluator $f(\cdot)$ is given detailed rules for evaluation and is asked to generate an explanation followed by the verdict. This process is formally represented as $f(P_{eval}, R, I, A_1, A_2) \rightarrow (exp, verdict)$.

Axis (Hada et al., 2023): Extending the Axis strategy defined in Sec §4.1, the evaluator $f(\cdot)$ is asked to choose the better response along a designated axis Ax. The evaluator is prompted with the instruction I, two model responses A_1, A_2, and the description of the axis Ax - represented as $f(P_{eval}, Ax, R, I, A_1, A_2) \rightarrow (exp, verdict)$.

Axis+Rules (Zeng et al., 2023; Hada et al., 2023): Extending the Axis+Rubric strategy defined in Sec §4.1, this strategy involves choosing the better response along the designated axis Ax. The evaluator is prompted with the instruction I, two model responses A_1, A_2, details about the axis Ax, and detailed rules for evaluation - represented as $f(P_{eval}, Ax, R, I, A_1, A_2) \rightarrow (exp, verdict)$.

4.3 Reference-guided Single Answer Scoring

In this paradigm, the evaluator $f(\cdot)$ is tasked to score a response by comparing against a reference. *It is important to note that this approach may not be feasible for many open-ended questions.*

Reference (Zheng et al., 2023): In this strategy, given an instruction I, a model response A_{model}, and a ground truth reference answer A_{gold}, the evaluator $f(\cdot)$ is tasked with scoring the model response, along with giving an explanation. This is formally represented as $f(P_{eval}, I, A_{gold}, A_{model}) \rightarrow (exp, score)$.

5 Experiments

We use GPT-4-TURBO as our primary evaluation model, given its widespread adoption (Zeng et al., 2023; Hada et al., 2024; Min et al., 2023). We also extend our analysis to other proprietary models - GEMINI-1.5-PRO (Team et al., 2024) and CLAUDE-3-OPUS (Anthropic, 2024), open-source models like LLAMA-3-70B-INSTRUCT (Meta, 2024), and trained evaluator models like PROMETHEUS 2 (Kim et al., 2024b)[2]. All evaluations are conducted at a temperature of zero to ensure reproducibility.

In single answer scoring (§ 4.1) paradigm, we measure the percentage of instances where the score remains unchanged by the perturbation as our metric. Ideally, except for score-invariant perturbations, the evaluator should penalize the score of the perturbed answer. For pairwise comparison paradigm (§ 4.2), we include our "gold" answer as one of the responses, requiring the evaluator to select the best response between the "gold" and the "perturbed" answer. Here, we measure the percentage of times the evaluator does not choose the gold answer as our metric. To mitigate position bias (Wang et al., 2023c), we conduct each evaluation twice, swapping the order of the gold and perturbed responses. For reference-guided single answer scoring paradigm (§ 4.3), the gold answer serves as the reference. Here, we measure the percentage of times the evaluator awards a perfect score to the perturbed answer as our metric.

5.1 Is GPT-4-Turbo a good evaluator?

Referring to the first section of Table 3, we observe that in the case of single answer scoring,

[2] We reuse the axes and rubrics defined in Section §4.1 as the evaluation rubrics for PROMETHEUS 2.

Strategy	LF↓	F↓	IF↓	R↓	SI↑
Single Answer Scoring					
Vanilla*	0.73	0.67	0.71	**0.22**	0.83
Vanilla	**0.57**	**0.54**	**0.57**	0.25	0.71
Rubric	0.85	0.73	0.80	0.33	0.96
Axis	0.83	0.74	0.75	0.43	0.96
Axis+Rubric	0.86	0.76	0.77	0.37	**0.97**
Pairwise Comparison					
Pairwise*	0.73	0.52	0.83	0.36	**0.93**
Pairwise	0.77	0.46	0.67	0.35	0.74
Rules	0.75	0.63	0.68	0.41	0.74
Axis	0.64	0.44	**0.59**	**0.27**	0.71
Axis+Rules	**0.64**	**0.42**	0.61	0.32	0.72
Reference-guided Single Answer Scoring					
Reference	0.26	0.11	0.49	0.04	0.63

Table 3: Comparison of different evaluation strategies using GPT-4-TURBO. The numbers indicate the percentage of instances where the score/verdict generated by the LLM evaluator is **not affected** by the perturbation. Lower values (↓) indicate better performance in all categories except SI. * denotes evaluators that only give a score without any justification.

Strategy	Model	LF↓	F↓	IF↓	R↓	SI↑
Vanilla		**0.57**	**0.54**	0.57	**0.25**	0.71
		0.61	0.73	**0.54**	0.41	0.71
		0.74	0.84	0.75	0.47	-
		0.86	0.95	0.90	0.71	**0.75**
Axis+Rules		**0.64**	**0.42**	**0.61**	**0.32**	**0.72**
		0.72	0.58	0.70	0.39	0.65
		0.75	0.69	0.70	0.60	0.64
Reference		0.26	0.11	0.49	0.04	**0.63**
		0.25	0.07	0.17	0.03	0.33
		0.03	**0.01**	**0.05**	**0.05**	0.13
		0.51	0.62	0.53	0.12	0.38

Table 4: Comparison of the performance of different models across the best-observed evaluation strategies. Lower values (↓) indicate better performance in all categories except SI.

GPT-4-TURBO fails to lower its score for the perturbed answer in a majority of the cases, except for Reasoning tasks. Further, the performance of GPT-4-TURBO is better when using simpler strategies, such as, Vanilla* and Vanilla, as compared to the more advanced strategies with explicit rubrics and/or specified axis of evaluation. This could imply that while adding additional rubrics and criteria may increase the overall thoroughness, it may not necessarily enhance the model's ability to detect subtler errors.

Now, referring to the second section of Table 3, we observe that in the case of pairwise comparison, GPT-4-TURBO fails to detect the perturbed answer in majority of the cases, except for Reasoning tasks. Further, in contrast to the above, in this case, advanced strategies perform better than the basic strategies. This indicates that for comparative evaluations, having detailed specific rules can help improve the reliability of the models. Lastly, referring to the first row of the last section of Table 3, we observe that when a reference is provided, GPT-4-TURBO performs much better but there are still a notable number of failures. The evaluator, despite being presented with the gold answer marked as a reference answer, fails to recognize the perturbations in many cases, except for reasoning tasks where it performs very well. *Our overall verdict is that* GPT-4-TURBO *is not a good evaluator as it fails to detect perturbations which cause a drop in the quality of the answer.*

5.2 How do other popular Evaluator LLMs perform?

We extend our evaluation to other models and compare their performance when using the 3 best strategies identified in Table 3. Table 4 shows that GPT-4-TURBO consistently outperforms other models in both the reference-less paradigms. Due to the high API cost of using the CLAUDE-3-OPUS model, we restrict its evaluation to only the Vanilla strategy, and note that it performed poorly as an Evaluator LLM.

In the reference-based paradigm, LLAMA-3-70B-INSTRUCT model surprisingly outperforms all others. Upon manually reviewing few instances, we observe that LLAMA-3-70B-INSTRUCT is a stringent evaluator and rarely awards perfect scores to even very well-formed answers when presented with a reference answer. While this may suggest that LLAMA-3-70B-INSTRUCT has a high evaluation standard, it also raises concerns about overlyrelying on the reference answer, which is typically not available in most practical scenarios. To further investigate this, we evaluate all the models on Score Invariant perturbations (Section §3.5) using the Reference evaluation strategy. Consistent with our prior observations, LLAMA-3-70B-INSTRUCT seldom awards perfect scores, doing so only in 13% of the cases as shown in Table 4. Lastly, looking at the last row of Table 4, we observe that even trained Evaluator LLMs like PROMETHEUS 2 are

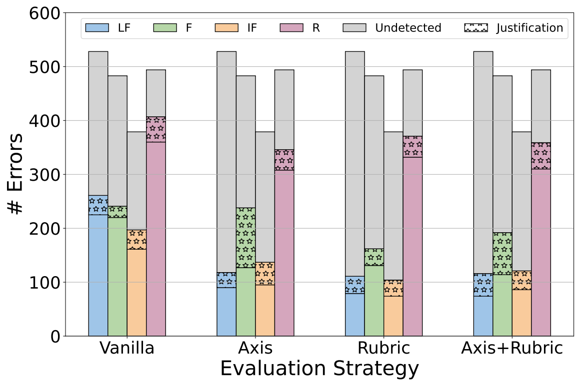

Figure 2: Comparison of perturbations detected solely by score analysis versus those identified with explanations. The highlighted region marked with stars denotes perturbations detected in explanations but not reflected in scores. Despite this, a significant proportion of perturbations remain undetected.

	LF↓		F↓		IF↓		R↓	
	1-3	1-5	1-3	1-5	1-3	1-5	1-3	1-5
R	0.85	**0.76**	0.73	**0.69**	0.80	**0.72**	0.33	**0.30**
A+R	0.86	**0.73**	0.76	**0.74**	0.77	**0.74**	**0.37**	0.38

Table 5: Comparing performance of Rubrics and Axis+Rubrics strategies with score range of 1-3 and 1-5. The numbers indicate the percentage of instances where the score generated by the LLM evaluator is not affected by the perturbation. Lower values (↓) indicate better performance in all categories.

worse than other general Evaluator LLMs.

5.3 Does it help to look beyond scores?

In addition to scoring, our evaluators also generate explanations that provide a justification for each score. We investigate whether these explanations detect the perturbations, even though this is not reflected in the scores. We prompt GPT-3.5-TURBO model with explanations from the instances where the evaluator rated the perturbed answer as equal to the gold answer, asking it to identify if any mistake or error has been reported in the explanation. Figure 2 reveals that explanations are only marginally helpful. Although perturbations are sometimes identified, they are overlooked or not considered significant enough to penalize the score. It is important to note that all the perturbations here were intended to incur a scoring penalty. Thus, while explicitly considering the explanations offers a slight improvement in the evaluator's performance, the overall performance is still poor.

5.4 What about score-invariant perturbations?

We evaluate different Evaluator LLMs using score-invariant perturbations (§ 3.5). Ideally, the evaluator should not reduce its score for these perturbations in score-based evaluations and should deem both responses correct in pairwise evaluations. Referring to Table 3 , in reference-less scoring, GPT-4-TURBO performs better when using non-vanilla evaluating strategies, while in pairwise comparison, it performs better when using simpler evaluation strategies. Similarly, as shown in Table 4, we observe that other Evaluator LLMs also perform well in a majority of cases. However, there is still a significant number of responses with score-invariant perturbations that they rate poorly.

5.5 Does increasing the range help in scoring?

Based on recommendations from Hada et al. (2023), our initial set-up for the Rubrics and Axis+Rubrics evaluators used a scoring range of 1 to 3. To explore whether a wider scoring range could enhance the evaluators' ability to identify and account for the perturbations, we extended the range to 1 to 5. Results presented in Table 5 suggest that this broader range slightly improves the evaluators' performance, perhaps due to the availability of more flexibility in scoring decisions.

6 Conclusion

We propose FBI, a novel framework designed to evaluate the proficiency of Evaluator LLMs in assessing four critical abilities: factual accuracy, instruction adherence, coherence in long-form writing, and reasoning proficiency, through targeted perturbations. Our comprehensive study, involving 2400 perturbed answers across 22 categories and using three evaluation paradigms (single-answer, pairwise, and reference-guided evaluation), reveals significant shortcomings in current Evaluator LLMs. Our findings show that even the most advanced models failed to identify quality drops in over 50% of cases on average. While reference-based evaluations performed relatively better, single-answer and pairwise evaluations demonstrated notable limitations. These results underscore the unreliable nature of current Evaluator LLMs and advocate for cautious implementation in practical applications. We hope that the FBI framework will be further extended and used for continued meta-evaluation of Evaluator LLMs.

Limitations

In our evaluation setup, detailed in Section 4, we concentrate on three primary evaluation paradigms: single-answer assessment, pairwise comparison, and reference-guided evaluation within a single model context and leave out multi-agent meta-evaluation and for future work. While we have compiled a list of perturbation categories, we believe it is not exhaustive and there is room room for further expansion. Our evaluation framework encompasses four fundamental task abilities, with plans to explore more advanced capabilities such as multilingual generation, tool usage, and planning in future work.

Ethics

All annotations described in Section 3 were done by students from our research group, all of whom hold at least a bachelor's or master's degree. This annotation was done as a part of their routine research work. The datasets used in this paper are all available under permissible licenses, and we adhere strictly to their intended usage, maintaining compliance with licensing requirements. Additionally, the code used for our evaluations and perturbation generation will be made publicly available under the MIT License[3]. We only used ChatGPT[4] for assistance purely with the language of the paper, e.g., paraphrasing, spell-checking, or polishing the author's original content, without suggesting new content.

Acknowledgements

We would like to thank EkStep Foundation and Nilekani Philanthropies for their generous grant, which supported this research. We extend our gratitude to Ananth, Devilal, Niharika, Nikhil, Sakshi, Sparsh, and Suriya for their invaluable assistance with manual audits. We also thank Raj Dabre and Anoop Kunchukuttan for their insightful discussions. We thank Google for supporting Sumanth's work through the Google Ph.D. Fellowship.

References

Anthropic. 2024. Introducing the next generation of claude. https://www.anthropic.com/news/claude-3-family. Accessed: 2024-06-14.

[3]https://opensource.org/licenses/MIT
[4]https://chatgpt.com

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

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: A preliminary empirical study. *CoRR*, abs/2304.00723.

Yiming Chen, Chen Zhang, Danqing Luo, Luis Fernando D'Haro, Robby T. Tan, and Haizhou Li. 2024. Unveiling the achilles' heel of nlg evaluators: A unified adversarial framework driven by large language models. *arXiv preprint arXiv: 2405.14646*.

Cheng-Han Chiang and Hung yi Lee. 2023. Can large language models be an alternative to human evaluations? *Annual Meeting of the Association for Computational Linguistics*.

David Cheng-Han Chiang and Hung-yi Lee. 2023. Can large language models be an alternative to human evaluations? In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 15607–15631. Association for Computational Linguistics.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. 2021. Training verifiers to solve math word problems. *CoRR*, abs/2110.14168.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Ning Ding, Yulin Chen, Bokai Xu, Yujia Qin, Shengding Hu, Zhiyuan Liu, Maosong Sun, and Bowen Zhou. 2023. Enhancing chat language models by scaling high-quality instructional conversations. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pages 3029–3051. Association for Computational Linguistics.

Yann Dubois, Chen Xuechen Li, Rohan Taori, Tianyi Zhang, Ishaan Gulrajani, Jimmy Ba, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2023. Alpacafarm: A simulation framework for methods that learn from human feedback. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

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

Rishav Hada, Varun Gumma, Mohamed Ahmed, Kalika Bali, and Sunayana Sitaram. 2024. Metal: Towards multilingual meta-evaluation. *arXiv preprint arXiv: 2404.01667*.

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

Hosein Hasanbeig, Hiteshi Sharma, Leo Betthauser, Felipe Vieira Frujeri, and Ida Momennejad. 2023. Allure: Auditing and improving llm-based evaluation of text using iterative in-context-learning. *arXiv preprint arXiv: 2309.13701*.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. 2021. Measuring mathematical problem solving with the MATH dataset. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks 1, NeurIPS Datasets and Benchmarks 2021, December 2021, virtual*.

Xuming Hu, Junzhe Chen, Xiaochuan Li, Yufei Guo, Lijie Wen, Philip S. Yu, and Zhijiang Guo. 2024. Towards understanding factual knowledge of large language models. In *The Twelfth International Conference on Learning Representations*.

Ryo Kamoi, Sarkar Snigdha Sarathi Das, Renze Lou, Jihyun Janice Ahn, Yilun Zhao, Xiaoxin Lu, Nan Zhang, Yusen Zhang, Ranran Haoran Zhang, Sujeeth Reddy Vummanthala, Salika Dave, Shaobo Qin, Arman Cohan, Wenpeng Yin, and Rui Zhang. 2024. Evaluating llms at detecting errors in llm responses. *arXiv preprint arXiv: 2404.03602*.

Seungone Kim, Jamin Shin, Yejin Cho, Joel Jang, Shayne Longpre, Hwaran Lee, Sangdoo Yun, Seongjin Shin, Sungdong Kim, James Thorne, and Minjoon Seo. 2023. Prometheus: Inducing fine-grained evaluation capability in language models. *CoRR*, abs/2310.08491.

Seungone Kim, Juyoung Suk, Ji Yong Cho, Shayne Longpre, Chaeeun Kim, Dongkeun Yoon, Guijin Son, Yejin Cho, Sheikh Shafayat, Jinheon Baek, Sue Hyun Park, Hyeonbin Hwang, Jinkyung Jo, Hyowon Cho, Haebin Shin, Seongyun Lee, Hanseok Oh, Noah Lee, Namgyu Ho, Se June Joo, Miyoung Ko, Yoonjoo Lee, Hyungjoo Chae, Jamin Shin, Joel Jang, Seonghyeon Ye, Bill Yuchen Lin, Sean Welleck, Graham Neubig, Moontae Lee, Kyungjae Lee, and Minjoon Seo. 2024a. The biggen bench: A principled benchmark for fine-grained evaluation of language models with language models.

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

Tom Kocmi and C. Federmann. 2023. Large language models are state-of-the-art evaluators of translation quality. *European Association for Machine Translation Conferences/Workshops*.

Zekun Li, Baolin Peng, Pengcheng He, and Xifeng Yan. 2023. Evaluating the instruction-following robustness of large language models to prompt injection. *arXiv preprint arXiv: 2308.10819*.

Yang Liu, Dan Iter, Yichong Xu, Shuo Wang, Ruochen Xu, and Chenguang Zhu. 2023. G-eval: Nlg evaluation using gpt-4 with better human alignment. *Conference on Empirical Methods in Natural Language Processing*.

Adian Liusie, Potsawee Manakul, and Mark J. F. Gales. 2023. Llm comparative assessment: Zero-shot nlg evaluation through pairwise comparisons using large language models. *arXiv preprint arXiv: 2307.07889*.

Nitika Mathur, Timothy Baldwin, and Trevor Cohn. 2020. Tangled up in BLEU: Reevaluating the evaluation of automatic machine translation evaluation metrics. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4984–4997, Online. Association for Computational Linguistics.

Meta. 2024. Introducing meta llama 3: The most capable openly available llm to date. `https://ai.meta.com/blog/meta-llama-3/`. Accessed: 2024-06-14.

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. *arXiv preprint arXiv: 2305.14251*.

Ben Naismith, Phoebe Mulcaire, and Jill Burstein. 2023. Automated evaluation of written discourse coherence using GPT-4. In *Proceedings of the 18th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2023)*, pages 394–403, Toronto, Canada. Association for Computational Linguistics.

Arjun Panickssery, Samuel R. Bowman, and Shi Feng. 2024. Llm evaluators recognize and favor their own generations. *arXiv preprint arXiv: 2404.13076*.

Marco Tulio Ribeiro, Tongshuang Wu, Carlos Guestrin, and Sameer Singh. 2020. Beyond accuracy: Behavioral testing of NLP models with CheckList. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4902–4912, Online. Association for Computational Linguistics.

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

Ananya B. Sai, Tanay Dixit, Dev Yashpal Sheth, Sreyas Mohan, and Mitesh M. Khapra. 2021. Perturbation CheckLists for evaluating NLG evaluation metrics. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7219–7234, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Ananya Sai B, Tanay Dixit, Vignesh Nagarajan, Anoop Kunchukuttan, Pratyush Kumar, Mitesh M. Khapra, and Raj Dabre. 2023. IndicMT eval: A dataset to meta-evaluate machine translation metrics for Indian languages. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14210–14228, Toronto, Canada. Association for Computational Linguistics.

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

Gemini Team, Machel Reid, Nikolay Savinov, Denis Teplyashin, Dmitry, Lepikhin, Timothy Lillicrap, Jean baptiste Alayrac, Radu Soricut, Angeliki Lazaridou, Orhan Firat, Julian Schrittwieser, Ioannis Antonoglou, Rohan Anil, Sebastian Borgeaud, Andrew Dai, Katie Millican, Ethan Dyer, Mia Glaese, Thibault Sottiaux, Benjamin Lee, Fabio Viola, Malcolm Reynolds, Yuanzhong Xu, James Molloy, Jilin Chen, Michael Isard, Paul Barham, Tom Hennigan, Ross McIlroy, Melvin Johnson, Johan Schalkwyk, Eli Collins, Eliza Rutherford, Erica Moreira, Kareem Ayoub, Megha Goel, Clemens Meyer, Gregory Thornton, Zhen Yang, Henryk Michalewski, Zaheer Abbas, Nathan Schucher, Ankesh Anand, Richard Ives, James Keeling, Karel Lenc, Salem Haykal, Siamak Shakeri, Pranav Shyam, Aakanksha Chowdhery, Roman Ring, Stephen Spencer, Eren Sezener, Luke Vilnis, Oscar Chang, Nobuyuki Morioka, George Tucker, Ce Zheng, Oliver Woodman, Nithya Attaluri, Tomas Kocisky, Evgenii Eltyshev, Xi Chen, Timothy Chung, Vittorio Selo, Siddhartha Brahma, Petko Georgiev, Ambrose Slone, Zhenkai Zhu, James Lottes, Siyuan Qiao, Ben Caine, Sebastian Riedel, Alex Tomala, Martin Chadwick, Juliette Love, Peter Choy, Sid Mittal, Neil Houlsby, Yunhao Tang, Matthew Lamm, Libin Bai, Qiao Zhang, Luheng He, Yong Cheng, Peter Humphreys, Yujia Li, Sergey Brin, Albin Cassirer, Yingjie Miao, Lukas Zilka, Taylor Tobin, Kelvin Xu, Lev Proleev, Daniel Sohn, Alberto Magni, Lisa Anne Hendricks, Isabel Gao, Santiago Ontanon, Oskar Bunyan, Nathan Byrd, Abhanshu Sharma, Biao Zhang, Mario Pinto, Rishika Sinha, Harsh Mehta, Dawei Jia, Sergi Caelles, Albert Webson, Alex Morris, Becca Roelofs, Yifan Ding, Robin Strudel, Xuehan Xiong, Marvin Ritter, Mostafa Dehghani, Rahma Chaabouni, Abhijit Karmarkar, Guangda Lai, Fabian Mentzer, Bibo Xu, YaGuang Li, Yujing Zhang, Tom Le Paine, Alex Goldin, Behnam Neyshabur, Kate Baumli, Anselm Levskaya, Michael Laskin, Wenhao Jia, Jack W. Rae, Kefan Xiao, Antoine He, Skye Giordano, Lakshman Yagati, Jean-Baptiste Lespiau, Paul Natsev, Sanjay Ganapathy, Fangyu Liu, Danilo Martins, Nanxin Chen, Yunhan Xu, Megan Barnes, Rhys May, Arpi Vezer, Junhyuk Oh, Ken Franko, Sophie Bridgers, Ruizhe Zhao, Boxi Wu, Basil Mustafa, Sean Sechrist, Emilio Parisotto, Thanumalayan Sankaranarayana Pillai, Chris Larkin, Chenjie Gu, Christina Sorokin, Maxim Krikun, Alexey Guseynov, Jessica Landon, Romina Datta, Alexander Pritzel, Phoebe Thacker, Fan Yang, Kevin Hui, Anja Hauth, Chih-Kuan Yeh, David Barker, Justin Mao-Jones, Sophia Austin, Hannah Sheahan, Parker Schuh, James Svensson, Rohan Jain, Vinay Ramasesh, Anton Briukhov, Da-Woon Chung, Tamara von Glehn, Christina Butterfield, Priya Jhakra, Matthew Wiethoff, Justin Frye, Jordan Grimstad, Beer Changpinyo, Charline Le Lan, Anna Bortsova, Yonghui Wu, Paul Voigtlaender, Tara Sainath, Shane Gu, Charlotte Smith, Will Hawkins, Kris Cao, James Besley, Srivatsan Srinivasan, Mark Omernick, Colin Gaffney, Gabriela Surita, Ryan Burnell, Bogdan Damoc, Junwhan Ahn, Andrew Brock, Mantas Pajarskas, Anastasia Petrushkina, Seb Noury, Lorenzo Blanco, Kevin Swersky, Arun Ahuja, Thi Avrahami, Vedant Misra, Raoul de Liedekerke, Mariko Iinuma, Alex Polozov, Sarah York, George van den Driessche, Paul Michel, Justin Chiu, Rory Blevins, Zach Gleicher, Adrià Recasens, Alban Rrustemi, Elena Gribovskaya, Aurko Roy, Wiktor Gworek, Sébastien M. R. Arnold, Lisa Lee, James Lee-Thorp, Marcello Maggioni, Enrique Piqueras, Kartikeya Badola, Sharad Vikram, Lucas Gonzalez, Anirudh Baddepudi, Evan Senter, Jacob Devlin, James Qin, Michael Azzam, Maja Trebacz, Martin Polacek, Kashyap Krishnakumar, Shuo yiin Chang, Matthew Tung, Ivo Penchev, Rishabh Joshi, Kate Olszewska, Carrie Muir, Mateo Wirth, Ale Jakse Hartman, Josh Newlan, Sheleem Kashem, Vijay Bolina, Elahe Dabir, Joost van Amersfoort, Zafarali Ahmed, James Cobon-Kerr, Aishwarya Kamath, Arnar Mar Hrafnkelsson, Le Hou, Ian Mackinnon, Alexandre Frechette, Eric Noland, Xiance Si, Emanuel Taropa, Dong Li, Phil Crone, Anmol Gulati, Sébastien Cevey, Jonas Adler, Ada Ma, David Silver, Simon Tokumine, Richard Powell, Stephan Lee, Kiran Vodrahalli, Samer Hassan, Diana Mincu, Antoine Yang, Nir Levine, Jenny Brennan, Mingqiu Wang, Sarah Hodkinson, Jeffrey Zhao, Josh Lipschultz, Aedan Pope, Michael B. Chang, Cheng Li, Laurent El Shafey, Michela Paganini, Sholto Douglas, Bernd Bohnet, Fabio Pardo, Seth Odoom, Mihaela Rosca, Cicero Nogueira dos Santos, Kedar Soparkar, Arthur Guez, Tom Hudson, Steven Hansen, Chulayuth Asawaroengchai, Ravi Addanki, Tianhe Yu, Wojciech Stokowiec, Mina Khan, Justin Gilmer, Jaehoon Lee, Carrie Grimes Bostock, Keran Rong, Jonathan Caton, Pedram Pejman, Filip Pavetic, Geoff Brown, Vivek Sharma, Mario Lučić, Rajkumar Samuel, Josip Djolonga, Amol Mandhane, Lars Lowe Sjösund, Elena Buchatskaya, Elspeth White, Natalie Clay, Jiepu Jiang, Hyeontaek Lim, Ross Hemsley, Zeyncep Cankara, Jane Labanowski, Nicola De Cao, David

Steiner, Sayed Hadi Hashemi, Jacob Austin, Anita Gergely, Tim Blyth, Joe Stanton, Kaushik Shivakumar, Aditya Siddhant, Anders Andreassen, Carlos Araya, Nikhil Sethi, Rakesh Shivanna, Steven Hand, Ankur Bapna, Ali Khodaei, Antoine Miech, Garrett Tanzer, Andy Swing, Shantanu Thakoor, Lora Aroyo, Zhufeng Pan, Zachary Nado, Jakub Sygnowski, Stephanie Winkler, Dian Yu, Mohammad Saleh, Loren Maggiore, Yamini Bansal, Xavier Garcia, Mehran Kazemi, Piyush Patil, Ishita Dasgupta, Iain Barr, Minh Giang, Thais Kagohara, Ivo Danihelka, Amit Marathe, Vladimir Feinberg, Mohamed Elhawaty, Nimesh Ghelani, Dan Horgan, Helen Miller, Lexi Walker, Richard Tanburn, Mukarram Tariq, Disha Shrivastava, Fei Xia, Qingze Wang, Chung-Cheng Chiu, Zoe Ashwood, Khuslen Baatarsukh, Sina Samangooei, Raphaël Lopez Kaufman, Fred Alcober, Axel Stjerngren, Paul Komarek, Katerina Tsihlas, Anudhyan Boral, Ramona Comanescu, Jeremy Chen, Ruibo Liu, Chris Welty, Dawn Bloxwich, Charlie Chen, Yanhua Sun, Fangxiaoyu Feng, Matthew Mauger, Xerxes Dotiwalla, Vincent Hellendoorn, Michael Sharman, Ivy Zheng, Krishna Haridasan, Gabe Barth-Maron, Craig Swanson, Dominika Rogozińska, Alek Andreev, Paul Kishan Rubenstein, Ruoxin Sang, Dan Hurt, Gamaleldin Elsayed, Renshen Wang, Dave Lacey, Anastasija Ilić, Yao Zhao, Adam Iwanicki, Alejandro Lince, Alexander Chen, Christina Lyu, Carl Lebsack, Jordan Griffith, Meenu Gaba, Paramjit Sandhu, Phil Chen, Anna Koop, Ravi Rajwar, Soheil Hassas Yeganeh, Solomon Chang, Rui Zhu, Soroush Radpour, Elnaz Davoodi, Ving Ian Lei, Yang Xu, Daniel Toyama, Constant Segal, Martin Wicke, Hanzhao Lin, Anna Bulanova, Adrià Puigdomènech Badia, Nemanja Rakićević, Pablo Sprechmann, Angelos Filos, Shaobo Hou, Víctor Campos, Nora Kassner, Devendra Sachan, Meire Fortunato, Chimezie Iwuanyanwu, Vitaly Nikolaev, Balaji Lakshminarayanan, Sadegh Jazayeri, Mani Varadarajan, Chetan Tekur, Doug Fritz, Misha Khalman, David Reitter, Kingshuk Dasgupta, Shourya Sarcar, Tina Ornduff, Javier Snaider, Fantine Huot, Johnson Jia, Rupert Kemp, Nejc Trdin, Anitha Vijayakumar, Lucy Kim, Christof Angermueller, Li Lao, Tianqi Liu, Haibin Zhang, David Engel, Somer Greene, Anaïs White, Jessica Austin, Lilly Taylor, Shereen Ashraf, Dangyi Liu, Maria Georgaki, Irene Cai, Yana Kulizhskaya, Sonam Goenka, Brennan Saeta, Ying Xu, Christian Frank, Dario de Cesare, Brona Robenek, Harry Richardson, Mahmoud Alnahlawi, Christopher Yew, Priya Ponnapalli, Marco Tagliasacchi, Alex Korchemniy, Yelin Kim, Dinghua Li, Bill Rosgen, Kyle Levin, Jeremy Wiesner, Praseem Banzal, Praveen Srinivasan, Hongkun Yu, Çağlar Ünlü, David Reid, Zora Tung, Daniel Finchelstein, Ravin Kumar, Andre Elisseeff, Jin Huang, Ming Zhang, Ricardo Aguilar, Mai Giménez, Jiawei Xia, Olivier Dousse, Willi Gierke, Damion Yates, Komal Jalan, Lu Li, Eri Latorre-Chimoto, Duc Dung Nguyen, Ken Durden, Praveen Kallakuri, Yaxin Liu, Matthew Johnson, Tomy Tsai, Alice Talbert, Jasmine Liu, Alexander Neitz, Chen Elkind, Marco Selvi, Mimi Jasarevic, Livio Baldini Soares, Albert Cui, Pidong Wang, Alek Wenjiao Wang, Xinyu Ye, Krystal Kallarackal, Lucia Loher, Hoi Lam, Josef Broder, Dan Holtmann-Rice, Nina Martin, Bramandia Ramadhana, Mrinal Shukla, Sujoy Basu, Abhi Mohan, Nick Fernando, Noah Fiedel, Kim Paterson, Hui Li, Ankush Garg, Jane Park, DongHyun Choi, Diane Wu, Sankalp Singh, Zhishuai Zhang, Amir Globerson, Lily Yu, John Carpenter, Félix de Chaumont Quitry, Carey Radebaugh, Chu-Cheng Lin, Alex Tudor, Prakash Shroff, Drew Garmon, Dayou Du, Neera Vats, Han Lu, Shariq Iqbal, Alex Yakubovich, Nilesh Tripuraneni, James Manyika, Haroon Qureshi, Nan Hua, Christel Ngani, Maria Abi Raad, Hannah Forbes, Jeff Stanway, Mukund Sundararajan, Victor Ungureanu, Colton Bishop, Yunjie Li, Balaji Venkatraman, Bo Li, Chloe Thornton, Salvatore Scellato, Nishesh Gupta, Yicheng Wang, Ian Tenney, Xihui Wu, Ashish Shenoy, Gabriel Carvajal, Diana Gage Wright, Ben Bariach, Zhuyun Xiao, Peter Hawkins, Sid Dalmia, Clement Farabet, Pedro Valenzuela, Quan Yuan, Ananth Agarwal, Mia Chen, Wooyeol Kim, Brice Hulse, Nandita Dukkipati, Adam Paszke, Andrew Bolt, Kiam Choo, Jennifer Beattie, Jennifer Prendki, Harsha Vashisht, Rebeca Santamaria-Fernandez, Luis C. Cobo, Jarek Wilkiewicz, David Madras, Ali Elqursh, Grant Uy, Kevin Ramirez, Matt Harvey, Tyler Liechty, Heiga Zen, Jeff Seibert, Clara Huiyi Hu, Andrey Khorlin, Maigo Le, Asaf Aharoni, Megan Li, Lily Wang, Sandeep Kumar, Norman Casagrande, Jay Hoover, Dalia El Badawy, David Soergel, Denis Vnukov, Matt Miecnikowski, Jiri Simsa, Praveen Kumar, Thibault Sellam, Daniel Vlasic, Samira Daruki, Nir Shabat, John Zhang, Guolong Su, Jiageng Zhang, Jeremiah Liu, Yi Sun, Evan Palmer, Alireza Ghaffarkhah, Xi Xiong, Victor Cotruta, Michael Fink, Lucas Dixon, Ashwin Sreevatsa, Adrian Goedeckemeyer, Alek Dimitriev, Mohsen Jafari, Remi Crocker, Nicholas FitzGerald, Aviral Kumar, Sanjay Ghemawat, Ivan Philips, Frederick Liu, Yannie Liang, Rachel Sterneck, Alena Repina, Marcus Wu, Laura Knight, Marin Georgiev, Hyo Lee, Harry Askham, Abhishek Chakladar, Annie Louis, Carl Crous, Hardie Cate, Dessie Petrova, Michael Quinn, Denese Owusu-Afriyie, Achintya Singhal, Nan Wei, Solomon Kim, Damien Vincent, Milad Nasr, Christopher A. Choquette-Choo, Reiko Tojo, Shawn Lu, Diego de Las Casas, Yuchung Cheng, Tolga Bolukbasi, Katherine Lee, Saaber Fatehi, Rajagopal Ananthanarayanan, Miteyan Patel, Charbel Kaed, Jing Li, Shreyas Rammohan Belle, Zhe Chen, Jaclyn Konzelmann, Siim Põder, Roopal Garg, Vinod Koverkathu, Adam Brown, Chris Dyer, Rosanne Liu, Azade Nova, Jun Xu, Alanna Walton, Alicia Parrish, Mark Epstein, Sara McCarthy, Slav Petrov, Demis Hassabis, Koray Kavukcuoglu, Jeffrey Dean, and Oriol Vinyals. 2024. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *arXiv preprint arXiv: 2403.05530*.

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

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

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

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

Ishaan Watts, Varun Gumma, Aditya Yadavalli, Vivek Seshadri, Swami Manohar, and Sunayana Sitaram. 2024. Pariksha: A scalable, democratic, transparent evaluation platform for assessing indic large language models.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Ed Huai hsin Chi, F. Xia, Quoc Le, and Denny Zhou. 2022. Chain of thought prompting elicits reasoning in large language models. *ArXiv*, abs/2201.11903.

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

Zhaofeng Wu, Linlu Qiu, Alexis Ross, Ekin Akyürek, Boyuan Chen, Bailin Wang, Najoung Kim, Jacob Andreas, and Yoon Kim. 2023. Reasoning or reciting? exploring the capabilities and limitations of language models through counterfactual tasks. *arXiv preprint arXiv: 2307.02477*.

Can Xu, Qingfeng Sun, Kai Zheng, Xiubo Geng, Pu Zhao, Jiazhan Feng, Chongyang Tao, and Daxin Jiang. 2023. Wizardlm: Empowering large language models to follow complex instructions. *CoRR*, abs/2304.12244.

Seonghyeon Ye, Doyoung Kim, Sungdong Kim, Hyeonbin Hwang, Seungone Kim, Yongrae Jo, James Thorne, Juho Kim, and Minjoon Seo. 2023. FLASK: fine-grained language model evaluation based on alignment skill sets. *CoRR*, abs/2307.10928.

Zhiyuan Zeng, Jiatong Yu, Tianyu Gao, Yu Meng, Tanya Goyal, and Danqi Chen. 2023. Evaluating large language models at evaluating instruction following. *CoRR*, abs/2310.07641.

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

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

Chunting Zhou, Pengfei Liu, Puxin Xu, Srini Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, Lili Yu, Susan Zhang, Gargi Ghosh, Mike Lewis, Luke Zettlemoyer, and Omer Levy. 2023a. LIMA: less is more for alignment. *CoRR*, abs/2305.11206.

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

Lianghui Zhu, Xinggang Wang, and Xinlong Wang. 2023. Judgelm: Fine-tuned large language models are scalable judges. *arXiv preprint arXiv: 2310.17631*.

A Manual Verication Process of the Perturbations

We engaged 17 graduate student volunteers with a good understanding of Large Language Models to manually verify the perturbations. Each annotator was provided with the instruction, the original gold answer, and the GPT-4-TURBO generated perturbed answer. They were tasked with classifying each perturbation into one of five categories: (i) Valid Perturbation, (ii) Invalid Perturbation, (iii) Score Invariant Perturbation, (iv) Not Relevant, and (v) Not Sure. Additionally, annotators were given explanations of the expected perturbations and the reasons why GPT-4-TURBO considered them valid.

To facilitate this process, we developed a straightforward application, the interface of which is depicted in Figure 3. This tool highlights the differences between the original and perturbed answers to aid easy identification.

Annotators were instructed to label an answer as "Valid Perturbation" only if they believed the perturbation warranted a score penalty relative to the gold answer. Perturbations not affecting the score were to be labeled "Score Invariant". If a perturbation was deemed incorrect or not reflected in the perturbed answer, annotators were asked to adjust the perturbation manually. Perturbations irrelevant to the category were to be marked as "Not Relevant".

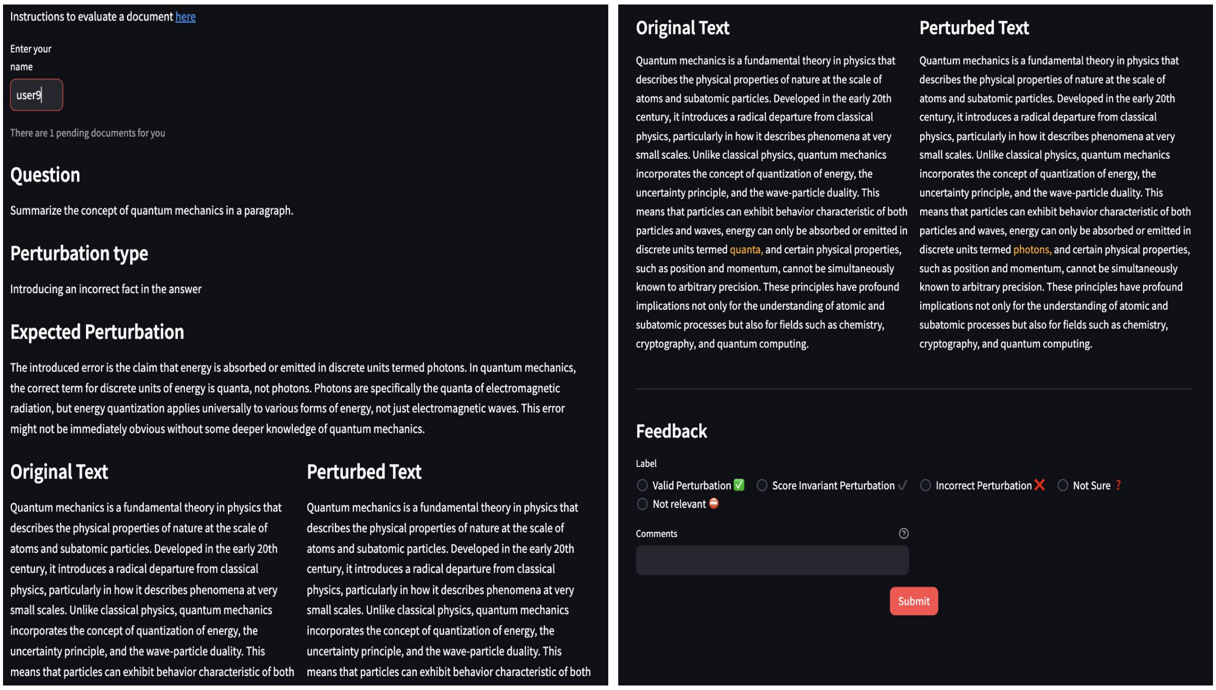

Figure 3: Screenshot of the User Application developed for validating perturbations.

B Detailed Results of Single Answer Evaluators

Detailed results of Single Answer evaluators can be found in Table 6, 7, 8, 9, 10.

C Detailed Results of Pairwise Evaluators

Detailed results of Pairwise Evaluators can be found in Table 11, 12, 13, 14, 15.

D Detailed Results of Reference-Guided Evaluators

Detailed results of Reference-guided Evaluators can be found in Table 16, 17

	Perturbation Type	TOTAL Errors	Detected Errors	Undetected Errors	% Undetected Errors
LF	COHERENCE	91	78	13	0.14
	COMPREHENSIVENESS	90	9	82	0.91
	CONSISTENCY	84	16	68	0.81
	GRAMMAR	92	25	67	0.73
	CHRONOLOGY	71	7	64	0.90
	SPELLING	100	11	89	0.89
	TOTAL	**528**	**146**	**383**	**0.73**
F	CONTEXTUAL	94	41	53	0.56
	ENTITY	87	29	58	0.67
	INCORRECT FACT	68	24	44	0.65
	NUMBER ERRORSS	74	22	52	0.70
	OPPOSITE FACT	91	39	52	0.57
	REMOVE FACT	69	4	65	0.94
	TOTAL	**483**	**159**	**324**	**0.67**
IF	ASSUMPTIONS	81	4	77	0.95
	DO LESS	100	32	68	0.68
	DO MORE	50	34	16	0.32
	IGNORE FORMAT	99	36	63	0.64
	SEQUENCE ERRORS	49	4	45	0.92
	TOTAL	**379**	**110**	**269**	**0.71**
R	CALCULATIONS	149	121	28	0.19
	COPYING NUMBERS	83	69	14	0.17
	FINAL ERRORS	97	54	43	0.44
	INCORRECT UNITS	77	66	11	0.14
	WRONG FORMULA	88	73	15	0.17
	TOTAL	**494**	**383**	**111**	**0.22**

Table 6: Results from evaluating FBI using **Vanilla*** evaluator. An error is said to be detected if the evaluator penalizes the score of the perturbed answer.

	Perturbation Type	TOTAL Errors	Detected Errors	Undetected Errors	% Undetected Errors
LF	COHERENCE	91	82	9	0.10
	COMPREHENSIVENESS	90	30	60	0.67
	CONSISTENCY	84	35	49	0.58
	GRAMMAR	92	40	52	0.57
	CHRONOLOGY	71	18	53	0.75
	SPELLING	100	20	80	0.80
	TOTAL	**528**	**225**	**303**	**0.57**
F	CONTEXTUAL	94	45	48	0.51
	ENTITY	87	43	44	0.51
	INCORRECT FACT	68	29	38	0.56
	NUMBER ERRORS	74	30	44	0.59
	OPPOSITE FACT	91	48	42	0.46
	REMOVE FACT	69	25	44	0.64
	TOTAL	**483**	**220**	**260**	**0.54**
IF	ASSUMPTIONS	81	12	69	0.85
	DO LESS	100	57	43	0.43
	DO MORE	50	31	19	0.38
	IGNORE FORMAT	99	41	57	0.58
	SEQUENCE ERRORS	49	20	29	0.59
	TOTAL	**379**	**161**	**217**	**0.57**
R	CALCULATIONS	149	112	34	0.23
	COPYING NUMBERS	83	69	12	0.14
	FINAL ERRORS	97	53	43	0.44
	INCORRECT UNITS	77	60	16	0.21
	WRONG FORMULA	88	66	19	0.22
	TOTAL	**494**	**360**	**124**	**0.25**

Table 7: Results from evaluating FBI using **Vanilla** evaluator. An error is said to be detected if the evaluator penalizes the score of the perturbed answer.

	Perturbation Type	Total Errors	Detected Errors	Undetected Errors	% Undetected Errors
LF	Coherence	91	47	44	0.48
	Comprehensiveness	90	2	88	0.98
	Consistency	84	11	73	0.87
	Grammar	92	15	77	0.84
	Chronology	71	0	71	1.00
	Spelling	100	4	96	0.96
	Total	**528**	**79**	**449**	**0.85**
F	Contextual	94	34	60	0.64
	Entity	87	29	58	0.67
	Incorrect Fact	68	18	50	0.74
	Number Errors	74	17	57	0.77
	Opposite Fact	91	32	59	0.65
	Remove Fact	69	1	68	0.99
	Total	**483**	**131**	**352**	**0.73**
IF	Assumptions	81	1	80	0.99
	Do Less	100	8	92	0.92
	Do More	50	39	11	0.22
	Ignore Format	99	26	73	0.74
	Sequence Errors	49	0	49	1.00
	Total	**379**	**74**	**305**	**0.80**
R	Calculations	149	102	47	0.32
	Copying Numbers	83	64	19	0.23
	Final Errors	97	49	48	0.49
	Incorrect Units	77	56	21	0.27
	Wrong Formula	88	61	27	0.31
	Total	**494**	**332**	**162**	**0.33**

Table 8: Results from evaluating FBI using **Rubrics** evaluator. An error is said to be detected if the evaluator penalizes the score of the perturbed answer.

	Perturbation Type	Total Errors	Detected Errors	Undetected Errors	% Undetected Errors
LF	Coherence	91	58	33	0.36
	Comprehensiveness	90	1	89	0.99
	Consistency	84	8	76	0.90
	Grammar	92	17	75	0.82
	Chronology	71	0	71	1.00
	Spelling	100	6	94	0.94
	Total	**528**	**90**	**438**	**0.83**
F	Contextual	94	29	65	0.69
	Entity	87	30	57	0.66
	Incorrect Fact	68	17	51	0.75
	Number Errors	74	18	56	0.76
	Opposite Fact	91	32	59	0.65
	Remove Fact	69	1	68	0.99
	Total	**483**	**127**	**356**	**0.74**
IF	Assumptions	81	5	76	0.94
	Do Less	100	20	80	0.80
	Do More	50	40	10	0.20
	Ignore Format	99	25	74	0.75
	Sequence Errors	49	5	44	0.90
	Total	**379**	**95**	**284**	**0.75**
R	Calculations	149	100	49	0.53
	Copying Numbers	83	57	26	0.31
	Final Errors	97	46	51	0.53
	Incorrect Units	77	42	35	0.45
	Wrong Formula	88	63	25	0.28
	Total	**494**	**308**	**186**	**0.43**

Table 9: Results from evaluating FBI using **Axis** evaluator. An error is said to be detected if the evaluator penalizes the score of the perturbed answer.

	Perturbation Type	Total Errors	Detected Errors	Undetected Errors	% Undetected Errors
LF	Coherence	91	45	46	**0.51**
	Comprehensiveness	90	0	90	1.00
	Consistency	84	6	78	0.93
	Grammar	92	16	76	0.83
	Chronology	71	0	71	1.00
	Spelling	100	7	93	0.93
	Total	**528**	**74**	**454**	**0.86**
F	Contextual	94	28	66	**0.70**
	Entity	87	27	60	0.69
	Incorrect Fact	68	15	53	0.78
	Number Errors	74	15	59	0.80
	Opposite Fact	91	28	63	0.69
	Remove Fact	69	1	68	0.99
	Total	**483**	**114**	**369**	**0.76**
IF	Assumptions	81	2	79	0.98
	Do Less	100	17	83	0.83
	Do More	50	39	11	0.22
	Ignore Format	99	24	75	0.76
	Sequence Errors	49	4	45	0.92
	Total	**379**	**86**	**293**	**0.77**
R	Calculations	149	97	52	0.35
	Copying Numbers	83	58	25	0.30
	Final Errors	97	48	49	0.51
	Incorrect Units	77	44	33	0.43
	Wrong Formula	88	63	25	0.37
	Total	**494**	**310**	**184**	**0.37**

Table 10: Results from evaluating FBI using **Axis+Rubrics** evaluator. An error is said to be detected if the evaluator penalizes the score of the perturbed answer.

	Perturbation Type	Total Errors	G	P	Both ✓	Both ✗	≠	% Undetected Errors
LF	Coherence	91	73	0	11	0	7	0.20
	Comprehensiveness	90	11	0	57	0	22	0.88
	Consistency	84	12	0	59	0	13	0.86
	Grammar	92	32	0	46	0	14	0.65
	Chronology	71	1	0	68	0	2	0.99
	Spelling	100	12	0	77	0	11	0.88
	Total	**528**	**141**	**0**	**318**	**0**	**69**	0.73
F	Contextual	94	55	0	12	0	27	0.41
	Entity	87	51	0	16	0	20	0.41
	Incorrect Fact	68	32	0	12	0	24	0.53
	Number Errors	74	29	1	22	0	22	0.61
	Opposite Fact	91	55	0	12	0	24	0.40
	Remove Fact	69	12	0	42	0	15	0.83
	Total	**483**	**234**	**1**	**116**	**0**	**132**	0.52
IF	Assumptions	81	6	25	34	0	16	0.93
	Do Less	100	40	0	22	0	38	0.60
	Do More	50	7	1	17	0	25	0.86
	Ignore Format	99	13	0	56	0	30	0.87
	Sequence Errors	49	0	0	49	0	0	1.00
	Total	**379**	**66**	**26**	**178**	**0**	**109**	0.83
R	Calculations	149	96	1	18	1	32	0.35
	Copying Numbers	83	58	0	7	1	17	0.30
	Final Errors	97	58	1	6	0	32	0.40
	Incorrect Units	77	48	0	17	1	11	0.38
	Wrong Formula	88	56	1	15	3	13	0.36
	Total	**494**	**316**	**3**	**63**	**6**	**105**	0.36

Table 11: Results from evaluating FBI using the **Pairwise**$*$ evaluator. An error is said to be detected if the evaluator chooses the Gold Answer. **G** indicates the number of times the evaluator has chosen the Gold Answer, **P** for the Perturbed Answer, **Both ✓** when both answers are correct, **Both ✗** when both are incorrect, and $\neq$ for verdict inconsistencies.

	Perturbation Type	Total Errors	G	P	Both ✓	Both ✗	$\neq$	% Undetected Errors
LF	Coherence	91	69	0	2	0	18	0.22
	Comprehensiveness	90	25	0	18	0	47	0.72
	Consistency	84	10	0	40	0	33	0.88
	Grammar	92	12	0	24	0	54	0.87
	Chronology	71	0	0	50	0	19	1.00
	Spelling	100	5	0	56	0	38	0.95
	Total	**528**	**121**	**0**	**190**	**0**	**209**	0.77
F	Contextual	94	76	0	5	0	13	0.19
	Entity	87	44	0	11	0	28	0.47
	Incorrect Fact	68	36	0	3	0	27	0.45
	Number Errors	74	34	0	9	0	28	0.52
	Opposite Fact	91	39	0	3	0	48	0.57
	Remove Fact	69	24	0	16	0	28	0.65
	Total	**483**	**253**	**0**	**47**	**0**	**172**	0.46
IF	Assumptions	81	4	43	3	0	31	0.95
	Do Less	100	58	0	11	0	30	0.41
	Do More	50	24	2	0	0	24	0.52
	Ignore Format	99	35	0	27	0	23	0.59
	Sequence Errors	49	0	0	23	0	26	1.00
	Total	**379**	**121**	**45**	**64**	**0**	**134**	0.67
R	Calculations	149	77	0	6	1	38	0.37
	Copying Numbers	83	40	0	1	1	18	0.33
	Final Errors	97	59	0	0	0	18	0.23
	Incorrect Units	77	38	0	7	0	20	0.42
	Wrong Formula	88	39	0	4	1	23	0.42
	Total	**494**	**253**	**0**	**18**	**3**	**117**	0.35

Table 12: Results from evaluating FBI using the **Pairwise** evaluator. An error is said to be detected if the evaluator chooses the Gold Answer. **G** indicates the number of times the evaluator has chosen the Gold Answer, **P** for the Perturbed Answer, **Both ✓** when both answers are correct, **Both ✗** when both are incorrect, and $\neq$ for verdict inconsistencies.

	Perturbation Type	Total Errors	G	P	Both ✓	Both ✗	≠	% Undetected Errors
LF	Coherence	91	82	0	2	0	7	0.10
	Comprehensiveness	90	28	0	25	0	37	0.69
	Consistency	84	10	0	46	0	28	0.88
	Grammar	92	8	0	24	0	60	0.91
	Chronology	71	0	0	51	0	20	1.00
	Spelling	100	4	0	48	0	48	0.96
	Total	**528**	**132**	**0**	**196**	**0**	**200**	0.75
F	Contextual	94	36	0	9	0	48	0.61
	Entity	87	37	0	14	0	34	0.56
	Incorrect Fact	68	27	0	4	0	36	0.60
	Number Errors	74	27	0	13	0	32	0.63
	Opposite Fact	91	32	0	6	0	53	0.65
	Remove Fact	69	19	0	18	0	32	0.72
	Total	**483**	**178**	**0**	**64**	**0**	**235**	0.63
IF	Assumptions	81	3	57	5	0	16	0.96
	Do Less	100	60	2	15	0	23	0.40
	Do More	50	25	3	0	0	22	0.50
	Ignore Format	99	33	0	29	0	37	0.67
	Sequence Errors	49	1	0	24	0	24	0.98
	Total	**379**	**122**	**62**	**73**	**0**	**122**	0.68
R	Calculations	149	82	1	12	0	46	0.42
	Copying Numbers	83	55	0	6	0	18	0.30
	Final Errors	97	47	1	0	0	42	0.48
	Incorrect Units	77	47	0	10	1	19	0.39
	Wrong Formula	88	46	1	7	0	27	0.43
	Total	**494**	**277**	**3**	**35**	**1**	**152**	0.41

Table 13: Results from evaluating FBI using the **Rules** evaluator. An error is said to be detected if the evaluator chooses the Gold Answer. **G** indicates the number of times the evaluator has chosen the Gold Answer, **P** for the Perturbed Answer, **Both ✓** when both answers are correct, **Both ✗** when both are incorrect, and ≠ for verdict inconsistencies.

	Perturbation Type	Total Errors	G	P	Both ✓	Both ✗	$\neq$	% Undetected Errors
LF	Coherence	91	82	0	1	0	8	0.10
	Comprehensiveness	90	49	0	9	0	32	0.46
	Consistency	84	16	0	50	0	18	0.81
	Grammar	92	34	0	26	0	32	0.63
	Chronology	71	0	0	57	0	14	1.00
	Spelling	100	11	0	58	0	31	0.89
	Total	**528**	**192**	**0**	**201**	**0**	**135**	0.64
F	Contextual	94	60	0	8	0	26	0.36
	Entity	87	60	0	11	0	16	0.31
	Incorrect Fact	68	41	0	4	0	23	0.40
	Number Errors	74	45	0	10	0	19	0.39
	Opposite Fact	91	61	0	7	0	23	0.33
	Remove Fact	69	5	0	58	0	6	0.93
	Total	**483**	**272**	**0**	**98**	**0**	**113**	0.44
IF	Assumptions	81	2	62	4	0	13	0.98
	Do Less	100	57	0	11	0	32	0.43
	Do More	50	40	2	3	0	5	0.20
	Ignore Format	99	53	0	13	0	33	0.46
	Sequence Errors	49	5	0	23	0	21	0.9
	Total	**379**	**157**	**64**	**54**	**0**	**104**	0.59
R	Calculations	149	108	1	16	0	23	0.27
	Copying Numbers	83	69	1	7	0	6	0.17
	Final Errors	97	75	1	2	0	19	0.23
	Incorrect Units	77	42	0	20	0	15	0.45
	Wrong Formula	88	64	0	12	0	12	0.27
	Total	**494**	**358**	**3**	**57**	**0**	**75**	0.27

Table 14: Results from evaluating FBI using the **Axis** evaluator. An error is said to be detected if the evaluator chooses the Gold Answer. **G** indicates the number of times the evaluator has chosen the Gold Answer, **P** for the Perturbed Answer, **Both ✓** when both answers are correct, **Both ✗** when both are incorrect, and $\neq$ for verdict inconsistencies.

	Perturbation Type	Total Errors	G	P	Both ✓	Both ✗	≠	% Undetected Errors
LF	Coherence	91	84	0	2	0	5	0.08
	Comprehensiveness	90	47	0	13	0	29	0.47
	Consistency	84	16	0	52	0	16	0.81
	Grammar	92	33	0	27	0	32	0.64
	Chronology	71	2	0	61	0	8	0.97
	Spelling	100	7	0	53	0	40	0.93
	Total	**528**	**189**	**0**	**208**	**0**	**130**	0.64
F	Contextual	94	56	0	8	0	28	0.39
	Entity	87	61	0	11	0	13	0.28
	Incorrect Fact	68	43	2	2	0	20	0.36
	Number Errors	74	43	0	8	0	21	0.40
	Opposite Fact	91	66	0	5	0	20	0.27
	Remove Fact	69	9	0	34	0	26	0.87
	Total	**483**	**278**	**2**	**68**	**0**	**128**	0.42
IF	Assumptions	81	2	65	2	0	12	0.98
	Do Less	100	59	0	8	0	33	0.41
	Do More	50	35	2	0	0	13	0.30
	Ignore Format	99	51	0	18	0	30	0.48
	Sequence Errors	49	1	0	29	0	19	0.98
	Total	**379**	**148**	**67**	**57**	**0**	**107**	0.61
R	Calculations	149	93	0	12	0	23	0.27
	Copying Numbers	83	58	0	6	0	12	0.24
	Final Errors	97	57	2	2	0	26	0.34
	Incorrect Units	77	38	0	19	0	16	0.48
	Wrong Formula	88	54	0	10	0	16	0.33
	Total	**494**	**300**	**2**	**49**	**0**	**93**	0.32

Table 15: Results from evaluating FBI using the **Axis+Rules** evaluator. An error is said to be detected if the evaluator chooses the Gold Answer. **G** indicates the number of times the evaluator has chosen the Gold Answer, **P** for the Perturbed Answer, **Both ✓** when both answers are correct, **Both ✗** when both are incorrect, and ≠ for verdict inconsistencies.

	Perturbation Type	Total Errors	10	9	8	<8	% Undetected Errors
LF	Coherence	91	2	5	9	75	0.02
	Comprehensiveness	90	32	39	6	13	0.36
	Consistency	84	31	27	4	22	0.37
	Grammar	92	10	51	9	22	0.11
	Chronology	71	47	21	2	1	0.66
	Spelling	100	14	72	4	10	0.14
	Total	**528**	**136**	**215**	**34**	**143**	0.26
F	Contextual	94	1	27	11	55	0.01
	Entity	87	8	16	16	47	0.09
	Incorrect Fact	68	2	15	12	39	0.03
	Number Errors	74	6	21	15	32	0.08
	Opposite Fact	91	0	11	4	76	0.00
	Remove Fact	69	36	18	10	5	0.52
	Total	**483**	**53**	**108**	**68**	**254**	0.11
IF	Assumptions	81	50	17	4	10	0.62
	Do Less	100	32	6	15	47	0.32
	Do More	50	22	10	12	6	0.44
	Ignore Format	99	43	18	7	30	0.43
	Sequence Errors	49	39	8	2	0	0.80
	Total	**379**	**186**	**59**	**40**	**93**	0.49
R	Calculations	149	6	6	6	131	0.04
	Copying Numbers	83	4	4	3	72	0.05
	Final Errors	97	1	2	4	89	0.01
	Incorrect Units	77	10	10	4	53	0.13
	Wrong Formula	88	1	12	1	74	0.01
	Total	**494**	**22**	**34**	**18**	**419**	0.04

Table 16: Results from evaluating FBI using the **Reference** evaluator. An error is said to be detected if the evaluator gives a perfect score of 10 to the perturbed answer. **10** indicates the number of times the evaluator has given the score of 10, **9** for the score of 9, **8** for the score of 8 and <**8** for scores less than 8.

	Perturbation Type	# Errs	Generic 5	Generic 4	Generic <4	Generic % Errors	Specific 5	Specific 4	Specific <4	Specific % Errors
LF	COHERENCE	91	9	33	49	0.10	17	18	56	0.19
	COMPREHENSIVENESS	90	40	42	8	0.44	40	46	4	0.44
	CONSISTENCY	84	36	36	12	0.43	50	26	8	0.60
	GRAMMAR	92	44	39	9	0.48	56	31	5	0.61
	CHRONOLOGY	71	43	24	4	0.61	42	23	6	0.59
	SPELLING	100	46	49	5	0.46	65	28	7	0.65
	TOTAL	**528**	**218**	**223**	**87**	0.41	**270**	**172**	**86**	0.51
F	CONTEXTUAL	94	46	39	9	0.49	56	25	13	0.60
	ENTITY	87	34	41	12	0.39	51	22	14	0.59
	INCORRECT FACT	68	29	30	9	0.43	45	18	5	0.66
	NUMBER ERRORS	74	36	32	6	0.49	47	18	9	0.64
	OPPOSITE FACT	91	37	41	13	0.41	52	28	11	0.57
	REMOVE FACT	69	41	27	1	0.59	50	18	1	0.72
	TOTAL	**483**	**223**	**210**	**50**	0.46	**301**	**129**	**53**	0.62
IF	ASSUMPTIONS	81	38	41	2	0.47	56	25	0	0.69
	DO LESS	100	53	44	3	0.53	54	44	2	0.54
	DO MORE	50	17	24	9	0.34	16	28	6	0.32
	IGNORE FORMAT	99	49	43	7	0.49	53	35	11	0.54
	SEQUENCE ERRORS	49	18	28	3	0.37	21	21	7	0.43
	TOTAL	**379**	**175**	**180**	**24**	0.46	**200**	**153**	**26**	0.53
R	CALCULATIONS	149	23	67	59	0.15	14	75	60	0.09
	COPYING NUMBERS	83	11	38	34	0.13	9	42	32	0.11
	FINAL ERRORS	97	22	46	29	0.23	10	54	33	0.10
	INCORRECT UNITS	77	16	23	38	0.21	8	34	35	0.10
	WRONG FORMULA	88	17	48	23	0.19	17	38	33	0.19
	TOTAL	**494**	**89**	**222**	**183**	0.18	**58**	**243**	**193**	0.12

Table 17: Results from evaluating FBI using the **Prometheus** evaluator. An error is said to be detected if the evaluator gives a perfect score of 5 to the perturbed answer. **5** indicates the number of times the evaluator has given the score of 5, **4** for the score of 4, and <**4** for scores less than 4. **Generic** indicates evaluating with general scoring rubrics and **Specific** indicates evaluating with task-specific rubrics.

	Perturbation Type	# Errs	Llama-3-70B-Instruct # DE	# UE	% UE	Claude-3-Opus # DE	# UE	% UE	Gemini-1.5-Pro # DE	# UE	% UE
LF	Coherence	91	61	21	0.29	72	18	0.20	83	8	0.09
	Comprehensiveness	90	22	59	0.82	19	71	0.79	29	60	0.67
	Consistency	84	9	65	1.00	18	65	0.78	29	55	0.65
	Grammar	92	8	80	0.95	12	80	0.87	29	63	0.68
	Chronology	71	1	60	1.15	6	64	0.91	18	53	0.75
	Spelling	100	7	85	1.01	13	80	0.92	18	82	0.82
	Total	**528**	**108**	**370**	0.86	**140**	**378**	0.74	**206**	**321**	0.61
F	Contextual	94	5	82	1.03	14	80	0.85	28	66	0.70
	Entity	87	13	64	0.94	22	65	0.75	32	55	0.63
	Incorrect Fact	68	6	55	1.02	10	58	0.85	15	53	0.78
	Number Errors	74	6	61	1.00	8	66	0.89	11	63	0.85
	Opposite Fact	91	10	74	0.96	17	74	0.81	32	59	0.65
	Remove Fact	69	18	49	0.74	8	61	0.88	13	56	0.81
	Total	**483**	**58**	**385**	0.95	**79**	**404**	0.84	**131**	**352**	0.73
IF	Assumptions	81	10	55	1.10	10	71	0.88	25	56	0.69
	Do Less	100	34	60	0.68	45	54	0.55	59	41	0.41
	Do More	50	11	35	0.81	11	39	0.78	26	24	0.48
	Ignore Format	99	12	53	1.71	25	74	0.75	49	49	0.51
	Sequence Errors	49	16	33	0.67	4	45	0.92	17	32	0.65
	Total	**379**	**83**	**236**	0.90	**95**	**283**	0.75	**176**	**202**	0.54
R	Calculations	149	55	82	0.65	90	59	0.40	81	64	0.43
	Copying Numbers	83	27	47	0.71	42	41	0.49	54	28	0.34
	Final Errors	97	18	70	0.88	35	62	0.64	36	60	0.63
	Incorrect Units	77	34	37	0.56	50	27	0.35	59	17	0.22
	Wrong Formula	88	25	54	0.77	43	44	0.51	55	32	0.37
	Total	**494**	**159**	**290**	0.71	**260**	**233**	0.47	**285**	**201**	0.41

Table 18: Results from evaluating FBI using **Vanilla**-Llama-3-70B-Instruct,Claude-3-Opus and Gemini-1.5-Pro evaluators. An error is said to be detected if the evaluator penalizes the score of the perturbed answer.

			Llama-3-70B-Instruct							Gemini-1.5-Pro					
	Perturbation Type	**# Errs**	**G**	**P**	**Both ✓**	**Both ✗**	**≠**	**% Errs**	**G**	**P**	**Both ✓**	**Both ✗**	**≠**	**% Errs**
LF	COHERENCE	91	59	0	0	0	18	0.23	77	0	2	0	12	0.15
	COMPREHENSIVENESS	90	40	0	0	0	34	0.46	40	0	24	0	26	0.56
	CONSISTENCY	84	8	0	2	0	69	0.90	13	0	49	0	22	0.85
	GRAMMAR	92	6	0	9	0	66	0.93	11	0	42	0	39	0.88
	CHRONOLOGY	71	0	0	1	0	63	1.00	3	0	52	0	16	0.96
	SPELLING	100	4	0	18	0	64	0.95	3	0	76	0	21	0.97
	TOTAL	**528**	**117**	**0**	**30**	**0**	**314**	0.75	**147**	**0**	**245**	**0**	**136**	0.72
F	CONTEXTUAL	94	20	0	5	0	40	0.69	43	1	8	0	42	0.54
	ENTITY	87	24	0	5	0	34	0.62	43	0	14	0	30	0.51
	INCORRECT FACT	68	11	1	2	0	34	0.77	30	0	2	0	36	0.56
	NUMBER ERRORS	74	16	0	2	0	38	0.71	33	0	10	0	31	0.55
	OPPOSITE FACT	91	14	0	4	0	46	0.78	41	0	5	0	45	0.55
	REMOVE FACT	69	24	0	4	0	33	0.61	12	0	35	0	22	0.83
	TOTAL	**483**	**109**	**1**	**22**	**0**	**225**	0.69	**202**	**1**	**74**	**0**	**206**	0.58
IF	ASSUMPTIONS	81	2	21	0	0	12	0.94	25	20	1	0	35	0.69
	DO LESS	100	44	1	1	0	37	0.47	38	0	11	2	49	0.62
	DO MORE	50	12	9	1	0	14	0.67	14	3	0	1	31	0.71
	IGNORE FORMAT	99	17	0	10	0	28	0.69	33	0	24	10	31	0.66
	SEQUENCE ERRORS	49	0	0	0	0	41	1.00	4	0	18	0	27	0.92
	TOTAL	**379**	**75**	**31**	**12**	**0**	**132**	0.70	**114**	**23**	**54**	**13**	**173**	0.70
R	CALCULATIONS	149	48	0	30	0	44	0.61	89	1	12	0	47	0.40
	COPYING NUMBERS	83	30	0	6	1	28	0.54	57	1	5	1	19	0.31
	FINAL ERRORS	97	30	0	3	0	41	0.59	59	2	0	1	35	0.39
	INCORRECT UNITS	77	27	0	11	0	25	0.57	40	0	12	1	24	0.48
	WRONG FORMULA	88	25	0	23	0	27	0.67	55	1	8	2	22	0.38
	TOTAL	**494**	**160**	**0**	**73**	**1**	**165**	0.60	**300**	**5**	**37**	**5**	**147**	0.39

Table 19: Results from evaluating FBI using the **Axis+Rules**-LLAMA-3-70B-INSTRUCT,CLAUDE-3-OPUS and GEMINI-1.5-PRO evaluators. An error is said to be detected if the evaluator chooses the Gold Answer. **G** indicates the number of times the evaluator has chosen the Gold Answer, **P** for the Perturbed Answer, **Both ✓** when both answers are correct, **Both ✗** when both are incorrect, and ≠ for verdict inconsistencies.

	Perturbation Type	# Errs	Llama-3-70B-Instruct 10	9	8	<8	% Errs	Gemini-1.5-Pro 10	9	8	<8	% Errs
LF	COHERENCE	91	1	2	3	75	0.02	1	1	2	87	0.01
	COMPREHENSIVENESS	90	1	33	23	20	0.01	13	29	26	21	0.14
	CONSISTENCY	84	2	41	19	6	0.03	5	48	18	12	0.06
	GRAMMAR	92	1	55	11	5	0.01	31	38	17	5	0.34
	CHRONOLOGY	71	3	34	13	1	0.06	15	41	12	3	0.21
	SPELLING	100	4	52	9	4	0.06	65	28	5	1	0.66
	TOTAL	**528**	**12**	**217**	**78**	**111**	0.03	**130**	**185**	**80**	**129**	0.25
F	CONTEXTUAL	94	0	49	19	19	0.00	4	34	24	29	0.04
	ENTITY	87	0	50	17	16	0.00	7	29	26	20	0.08
	INCORRECT FACT	68	0	38	18	9	0.00	2	31	20	13	0.03
	NUMBER ERRORS	74	2	53	10	4	0.03	3	34	22	9	0.04
	OPPOSITE FACT	91	0	37	19	30	0.00	4	18	30	38	0.04
	REMOVE FACT	69	4	29	22	13	0.06	13	18	26	12	0.19
	TOTAL	**483**	**6**	**256**	**105**	**91**	0.01	**33**	**164**	**148**	**121**	0.07
IF	ASSUMPTIONS	81	0	31	23	19	0.00	0	12	20	48	0.00
	DO LESS	100	1	23	35	32	0.01	26	15	27	32	0.26
	DO MORE	50	1	31	15	1	0.02	1	13	28	7	0.02
	IGNORE FORMAT	99	11	29	14	16	0.16	32	17	15	33	0.33
	SEQUENCE ERRORS	49	5	28	13	0	0.11	6	26	14	3	0.12
	TOTAL	**379**	**18**	**142**	**100**	**68**	0.05	**65**	**83**	**104**	**123**	0.17
R	CALCULATIONS	149	10	35	41	49	0.07	5	17	36	82	0.03
	COPYING NUMBERS	83	2	16	23	36	0.03	3	13	14	52	0.04
	FINAL ERRORS	97	0	20	56	13	0.00	0	28	44	23	0.00
	INCORRECT UNITS	77	2	22	11	34	0.03	3	17	12	45	0.04
	WRONG FORMULA	88	7	26	25	25	0.08	2	12	20	53	0.02
	TOTAL	**494**	**21**	**119**	**156**	**157**	0.05	**13**	**87**	**126**	**255**	0.03

Table 20: Results from evaluating FBI using the **Reference**-LLAMA-3-70B-INSTRUCT,CLAUDE-3-OPUS and GEMINI-1.5-PRO evaluators. An error is said to be detected if the evaluator gives a perfect score of 10 to the perturbed answer. **10** indicates the number of times the evaluator has given the score of 10, **9** for the score of 9, **8** for the score of 8 and <**8** for scores less than 8.

	Perturbation Type	# Errs	Detected Errors	Undetected Errors	Detected in Explanation	% Undetected Errors
LF	Coherence	91	82	9	1	**0.09**
	Comprehensiveness	90	30	60	5	**0.61**
	Consistency	84	35	49	7	**0.50**
	Grammar	92	40	52	9	**0.47**
	Chronology	71	18	53	3	**0.70**
	Spelling	100	20	80	11	**0.69**
	Total	**528**	**225**	**303**	**36**	**0.51**
F	Contextual	94	45	48	5	**0.47**
	Entity	87	43	44	3	**0.47**
	Incorrect Fact	68	29	38	4	**0.51**
	Number Errors	74	30	44	3	**0.55**
	Opposite Fact	91	48	42	6	**0.41**
	Remove Fact	69	25	44	0	**0.64**
	Total	**483**	**220**	**260**	**21**	**0.50**
IF	Assumptions	81	12	69	7	**0.77**
	Do Less	100	57	43	6	**0.37**
	Do More	50	31	19	12	**0.14**
	Ignore Format	99	41	57	10	**0.48**
	Sequence Errors	49	20	29	1	**0.57**
	Total	**379**	**161**	**217**	**36**	**0.48**
R	Calculations	149	112	34	15	**0.15**
	Copying Numbers	83	69	12	3	**0.13**
	Final Errors	97	53	43	16	**0.29**
	Incorrect Units	77	60	16	7	**0.13**
	Wrong Formula	88	66	19	6	**0.18**
	Total	**494**	**360**	**124**	**47**	**0.18**

Table 21: Results from looking at the explanation of the **Vanilla** evaluator to determine the presence of the error in the response. Detected in Explanation shows the number of “additional” errors detected by looking at the explanation in addition to the score.

	Perturbation Type	# Errs	Detected Errors	Undetected Errors	Detected in Justification	% Undetected Errors
LF	Coherence	91	58	33	1	**0.35**
	Comprehensiveness	90	1	89	5	**0.93**
	Consistency	84	8	76	3	**0.87**
	Grammar	92	17	75	7	**0.74**
	Chronology	71	0	71	9	**0.87**
	Spelling	100	6	94	3	**0.91**
	Total	**528**	**90**	**438**	**28**	**0.78**
F	Contextual	94	29	65	23	**0.45**
	Entity	87	30	57	15	**0.48**
	Incorrect Fact	68	17	51	14	**0.54**
	Number Errors	74	18	56	16	**0.54**
	Opposite Fact	91	32	59	23	**0.40**
	Remove Fact	69	1	68	20	**0.70**
	Total	**483**	**127**	**356**	**111**	**0.51**
IF	Assumptions	81	5	76	8	**0.84**
	Do Less	100	20	80	0	**0.80**
	Do More	50	40	10	6	**0.08**
	Ignore Format	99	25	74	12	**0.63**
	Sequence Errors	49	5	44	16	**0.57**
	Total	**379**	**95**	**284**	**42**	**0.64**
R	Calculations	149	100	49	9	**0.27**
	Copying Numbers	83	57	26	9	**0.20**
	Final Errors	97	46	51	7	**0.45**
	Incorrect Units	77	42	35	7	**0.36**
	Wrong Formula	88	63	25	6	**0.22**
	Total	**494**	**308**	**186**	**38**	**0.30**

Table 22: Results from looking at the explanation of the **Axis** evaluator to determine the presence of the error in the response. Detected in Explanation shows the number of "additional" errors detected by looking at the explanation in addition to the score.

	Perturbation Type	# Errs	Detected Errors	Undetected Errors	Detected in Justification	% Undetected Errors
LF	Coherence	91	47	44	2	**0.46**
	Comprehensiveness	90	2	88	5	**0.92**
	Consistency	84	11	73	6	**0.80**
	Grammar	92	15	77	6	**0.77**
	Chronology	71	0	71	5	**0.93**
	Spelling	100	4	96	8	**0.88**
	Total	**528**	**79**	**449**	**32**	**0.79**
F	Contextual	94	34	60	3	**0.61**
	Entity	87	29	58	3	**0.63**
	Incorrect Fact	68	18	50	2	**0.71**
	Number Errors	74	17	57	7	**0.68**
	Opposite Fact	91	32	59	6	**0.58**
	Remove Fact	69	1	68	10	**0.84**
	Total	**483**	**131**	**352**	**31**	**0.66**
IF	Assumptions	81	1	80	1	**0.98**
	Do Less	100	8	92	8	**0.84**
	Do More	50	39	11	2	**0.18**
	Ignore Format	99	26	73	14	**0.60**
	Sequence Errors	49	0	49	5	**0.90**
	Total	**379**	**74**	**305**	**30**	**0.73**
R	Calculations	149	102	47	10	**0.25**
	Copying Numbers	83	64	19	3	**0.19**
	Final Errors	97	49	48	9	**0.40**
	Incorrect Units	77	56	21	4	**0.22**
	Wrong Formula	88	61	27	13	**0.16**
	Total	**494**	**332**	**162**	**39**	**0.25**

Table 23: Results from looking at the explanation of the **Rubrics** evaluator to determine the presence of the error in the response. Detected in Explanation shows the number of "additional" errors detected by looking at the explanation in addition to the score.

	Perturbation Type	# Errs	Detected Errors	Undetected Errors	Detected in Justification	% Undetected Errors
LF	Coherence	91	45	46	0	**0.51**
	Comprehensiveness	90	0	90	11	**0.88**
	Consistency	84	6	78	8	**0.83**
	Grammar	92	16	76	5	**0.77**
	Chronology	71	0	71	12	**0.83**
	Spelling	100	7	93	6	**0.87**
	Total	**528**	**74**	**454**	**42**	**0.78**
F	Contextual	94	28	66	19	**0.50**
	Entity	87	27	60	9	**0.59**
	Incorrect Fact	68	15	53	10	**0.63**
	Number Errors	74	15	59	12	**0.64**
	Opposite Fact	91	28	63	12	**0.56**
	Remove Fact	69	1	68	16	**0.75**
	Total	**483**	**114**	**369**	**78**	**0.60**
IF	Assumptions	81	2	79	6	**0.90**
	Do Less	100	17	83	9	**0.74**
	Do More	50	39	11	1	**0.20**
	Ignore Format	99	24	75	14	**0.62**
	Sequence Errors	49	4	45	5	**0.82**
	Total	**379**	**86**	**293**	**35**	**0.68**
R	Calculations	149	97	52	14	**0.26**
	Copying Numbers	83	58	25	7	**0.22**
	Final Errors	97	48	49	12	**0.38**
	Incorrect Units	77	44	33	7	**0.34**
	Wrong Formula	88	63	25	9	**0.18**
	Total	**494**	**310**	**184**	**49**	**0.27**

Table 24: Results from looking at the explanation of the **Axis+Rubrics** evaluator to determine the presence of the error in the response. Detected in Explanation shows the number of "additional" errors detected by looking at the explanation in addition to the score.