Micro-Perturb: An Automated Perturbation-Based Framework for Evaluating LLMs on Table-Text Weakness Detection

Anonymous ACL submission

Abstract

Scholarly peer-review is a critical but time-consuming task. Existing work has studied how language models could assist in the review process, with one of the primary areas of study being on scientific error or weakness detection. Motivated by the existing work on scientific claim verification, we introduce the task of *table-text weakness detection* in scientific manuscripts. Through an inductive qualitative investigation over peer-review data, we identify five main categories of table-text weaknesses. We then introduce a guided perturbation pipeline for introducing such weaknesses into scientific manuscripts, and evaluate models on their ability to detect the weakness. We perform extensive human validations of our pipeline and find that it is effective at introducing most types of table-text weaknesses into scientific manuscripts. Finally, we evaluate models on a full generated dataset and human validated subset of the data. We find that models perform well at identifying certain types of table-text weaknesses, but struggle with others. Our data and code are available at http://anonymous.for.review

1 Introduction

Scholarly peer-review is a critical for ensuring the quality and reproducibility of scientific findings, but it is a time-consuming task that requires extensive domain expertise. With the reasoning performance demonstrated by recent language models, there has been increasing interest in incorporating them into peer-review workflows to reduce time and improve quality (Kuznetsov et al., 2024).

Recent work has explored a variety of peer-review related tasks and the ability for models to address them. This includes tasks such as novelty detection (Afzal et al., 2026), literature review (Tang et al., 2025), and assessing the utility of peer-reviews for authors (Sadallah et al., 2025).

While such tasks are important, one of the main objectives of peer-review is ensuring that the research findings are true and reproducible. As such, detecting errors and weaknesses in manuscripts is a critical aspect of the peer-review process [1].

Existing work has taken a few different approaches to constructing benchmarks for assessing the capabilities of models in detecting and understanding errors and weaknesses in manuscripts. One approach is to extract relevant content from peer-reviews and align it with manuscript content (Selch et al., 2026). While this ensures that the data reflects the actual types of issues identified by peer-reviewers, the sizes of such datasets can be limited, and can only be applied to areas where open peer-reviews are available.

Another approach is to introduce errors or weaknesses into manuscripts through perturbations. These perturbations can be performed by domain experts that are guided by what type of error to introduce (Xu et al., 2025), which can help maintain the quality and diversity of the dataset but at the cost of scalability. Alternatively, the perturbations can be performed by LLMs (Xi et al., 2025), which provides a highly scalable approach for generating data but which may not reflect the types of errors actually encountered by peer-reviewers.

Combining these approaches for any type of scientific error or weakness is challenging due to the diversity of issues that can arise in a scientific manuscript. To resolve this, we choose to take a more targeted approach in this work to assessing model capabilities. We focus on the task of *table-text weakness detection*.

Our choice is motivated by the extensive work on scientific claim verification with tabular evidence (Lu et al., 2023; Ho et al., 2026; Wang et al.,

[1] In this work, we refer to errors when there exists a clear logical contradiction with an established truth (e.g. evidence). Weaknesses are broader and refer to any negative aspect of a manuscript.

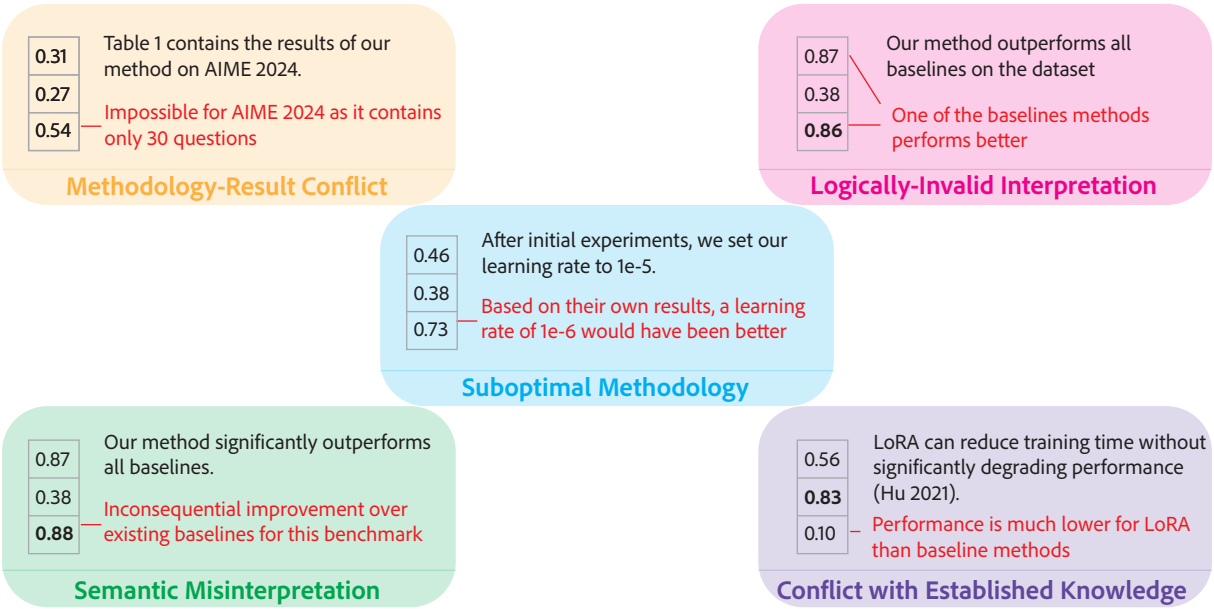

Figure 1: Overview of our table-text weakness categories, with a simple illustrative example for each category.

2025) which suggests that models may struggle to reason about whether tabular evidence supports a claim. During our own study, we found that human annotators struggled to detect weaknesses for some cases, even when provided with the relevant extracted textual and tabular context.

To understand the types of table-text weaknesses that peer-reviewers detect in scientific manuscripts, we begin with a rigorous qualitative investigation over peer-review data. We use an inductive approach to extract insight into the types of table-text weaknesses that actually exist in scientific manuscripts submitted for peer-review. We uncover five main categories of weaknesses, an overview of which is presented in Figure 1.

We then introduce a guided perturbation pipeline for automatically inserting table-text weaknesses into manuscripts. We perform extensive human validation of our pipeline's output to understand its performance. We find that for most types of table-text weaknesses, the pipeline is highly successful at introducing the desired weakness into a manuscript.

Finally, we perform extensive evaluations to understand model performance on this task. We find that models display variability in their ability to detect table-text weaknesses based on the category. However, models appear to be highly adept at detecting certain types of weaknesses, which suggests that they could be incorporated into the peer-review workflow.

Our contributions are as follows:

- We introduce a categorization of five table-text weaknesses based on a rigorous qualitative investigation performed over real peer-review data.
- We introduce a guided perturbation pipeline for automatically inserting table-text weaknesses into scientific manuscripts, and evaluate its performance through extensive human validations.
- We perform extensive evaluations on the task of *table-text weakness detection* using various models, and find that model performance can vary by the category of error. However, results suggest that models could be incorporated into the peer-review workflow for detecting weaknesses in high-performing categories.

2 Related Work

In this section, we describe related work on scientific claim verification with tabular evidence and scientific error and weakness detection benchmarks. We provide an overview comparing our work and key related work in Table 1.

2.1 Scientific Claim Verification

Claim verification is the task of determining whether a textual claim is supported, refuted, or neutral with respect to evidence (Nakov et al., 2021). Claim verification is an essential part of the fact-checking pipeline, and we can consider the

Citation	Task	Peer-Review Grounded	Perturbation Source	Perturbation Target
(Lu et al., 2023)	Claim Verification	No	Human	Claim
(Wang et al., 2025)	Claim Verification	No	Human	Claim
(Ho et al., 2026)	Claim Verification	No	Automated	Evidence (Figures, Tables)
(Selch et al., 2026)	MCQA	Yes	NA	NA
(Xi et al., 2025)	Error Localization	No	Automated	Manuscript Text
(Xu et al., 2025)	Limitation Detection	Yes	Human	Manuscript Text
Ours	Table-Text Weakness Detection	Yes	Automated	Manuscript Table

Table 1: **Related Work**. Comparison of our work and some key related work.

task of detecting errors in scientific manuscripts as identifying claims that are refuted by scientific evidence. Scientific claim verification is a variation of this task that requires verifying scientific claims with evidence that is usually extracted from scientific manuscripts, such as abstracts (Wadden et al., 2020) and tables (Lu et al., 2023).

Claim verification with tabular data has been of particular interest to researchers due to the challenges of working with such structured data. Earlier work in this area explored automated claim verification primarily using tables and claims from Wikipedia (Chen et al., 2020; Gupta et al., 2020). Recent work has extended tabular claim verification to the scientific domain. SEM-TAB-FACTS (Wang et al., 2021) was one of the earlier benchmarks based on real scientific tables and crowdsourced claims. SciTab (Lu et al., 2023) extracted real scientific claims from papers and filtered for those "highlighting and commenting on key data". SciVer extends this task to cases where reasoning is required over multiple figures and tables in the evidence (Wang et al., 2025). Finally, MuSciClaim (Lal et al., 2025) is constructed from perturbed scientific claims and multimodal evidence.

While these works are relevant as some table-text weaknesses can be viewed as scientific claim verification tasks, most of these works extract the relevant claims and evidence from the document, rather than requiring the model to detect and localize errors itself. SciClaimEval (Ho et al., 2026) is one of the most similar to our work, as it uses original sentences from the manuscript while perturbing the evidence. However, it still focuses on strict errors while ours includes weaknesses that are not direct contradictions between claim and evidence. Furthermore, like most claim verification benchmarks, it provides the claim extracted from the manuscript, while the full document is only provided as evidence in 13% of cases. Finally, they perform structural perturbations over table data, which may introduce specific types of semantic errors more frequently, and not fully capture the complexity of the types of errors actually encountered by peer-reviewers.

2.2 Scientific Paper Understanding Benchmarks

Another line of related work is on benchmarking model performance on scientific understanding tasks. The most relevant to our work are those that focus on error, weakness, or limitation identification. Unlike scientific claim verification, such tasks require models to identify issues using the full manuscript context, rather than being provided with specific claims and evidence.

Benchmarks such as QASA (Lee et al., 2023) require models to understand scientific manuscripts and evaluate them on question-answering tasks, while SPIQA (Pramanick et al., 2024) extends this to multimodal settings. PRISMM-Bench (Selch et al., 2026) assesses models on real peer-reviewer identified multimodal weaknesses. Their dataset provides reviewer extracted comments alongside relevant multimodal content extracted from the scientific manuscript. While highly relevant to our work as we use it as the starting point of our qualitative investigation into the types of real table-text weaknesses identified by peer-reviewers, it focuses on MCQA and is limited in size and coverage of rare weaknesses due to its reliance on real peer-review data.

The most relevant benchmarks on scientific understand to our work introduce errors and limitations into scientific manuscripts through perturbations. LimitGen (Xu et al., 2025) is a benchmark for identifying critical limitations in scientific manuscripts. Furthermore, the benchmark is grounded in real limitations that could be encountered by peer-reviewers. However, unlike our work, the perturbations are manually introduced into the manuscript, which makes the method difficult to scale. Furthermore, the work focuses primarily on textual limitations, and perturbations are also text-

based. This means that modifying one part of the text can introduce unintended cross-text contradictions, while our approach performs edits only on the tables, ensuring that such cross-text contradictions do not arise unless related to the table.

Another highly relevant benchmark is (Xi et al., 2025). In this work the authors automate the process of introducing perturbations into scientific manuscripts. However, these perturbations may not necessarily reflect the type of errors actually encountered by peer-reviewers. Furthermore, their work focuses primarily on the task of localizing errors in the manuscript. Finally, their approach introduces strict errors into the text of scientific manuscripts, while we focus on multimodal table-text weaknesses, which includes errors but also negative arguments against particular aspects of a scholarly work (Kuznetsov et al., 2022; Hua et al., 2019).

3 Categories of Table-Text Weaknesses

To ensure that our method is grounded in the real-world needs of peer-reviewers, we begin with a qualitative investigation to understand the types of table-text weaknesses that are actually identified in scientific manuscripts. In this section, we describe our methodology for this investigation and the final categories that we discovered.

We employ an inductive, bottom-up approach to naturally derive categories from real peer-review data. We discovered five main types of table-text weaknesses: *methodology-result conflict*, *logically-invalid interpretation*, *suboptimal methodology*, *semantic misinterpretation*, and *conflict with established knowledge*. An overview of our final categories with a simple illustrative example for each category is provided in Figure 1.

3.1 Categorization Process

As the purpose of our qualitative investigation was to extract meaningful categories of table-text weaknesses from peer-review data, we employed an inductive approach based on Braun & Clarke's thematic analysis (Braun and Clarke, 2006). We analyzed data from PRISMM-bench (Selch et al., 2026), which contains reviewer comments along with relevant sentences, figures, and tables extracted from the manuscript.

As PRISMM-bench was constructed from sourcing reviews from OpenReview, the researchers conducting thematic analysis required sufficient knowledge and expertise to understand the extracted content. Thus, this process was conducted by two authors of this paper; a second-year PhD student in NLP and a postdoctoral researcher in NLP.

In the first stage of thematic analysis, the annotators are required to familiarize themselves with the data. Both annotators read the PRISMM-bench paper, while one annotator also reviewed all relevant instances from PRISMM-bench. In total, there were 46 instances in PRISMM-bench that were related to table-text weaknesses.

In the second stage, codes are generated that describe specific instances of data. For each instance of table-text data in PRISMM-bench, one annotator summarized the weakness identified by the reviewer with one or more short codes. Codes could be generated for an instance or old codes could be reused, depending on what the annotator believed was appropriate. For each instance, the annotator could also choose to describe the weakness in one or two sentences if there was important context that needed to be preserved for later stages of the analysis, or if additional explanation was required beyond the codes. Our annotation interface is shown in Appendix C.

Once the initial codes were generated, the annotators analyzed the codes to uncover recurring patterns and themes. Each annotator reviewed the entire list of generated codes and associated comments from the previous stage and formed an initial grouping of codes. The annotators then discussed the similarities and differences between their findings and collaboratively generated a new set of groups. During this process, the meaning of the initial codes was discussed, and any ambiguity was resolved by referencing the original context from PRISMM-bench. By the end of this process, the annotators had identified and defined five themes, which formed the categories used in the rest of this work.

Finally, during our pilot annotations for validating the quality of our guided perturbation pipeline (described in detail in Section 4), we identified and resolved ambiguities in the definitions for the categories. In one case, we expanded the scope of a category as the initial definition only applied to a small subset of the manuscripts we examined. All changes were taken only after carefully reviewing the initial codes for the category and with agreement from both annotators on the new categorization.

3.2 Final Categories

Below, we describe the five table–text weakness categories that we discovered through our inductive qualitative analysis of peer-review data.

Methodology-Result Conflict. The manuscript text describes how data was collected, filtered, and processed before being presented in the table. However, the data in the table contradicts the described methodology. The table may be missing values, contain values that are impossible if the methodology were correctly described, or have additional unexplained values. It is important to note that the manuscript text does not claim to analyze, compare, or interpret the meaning of the values in the table; rather, based on the values of the table, it must be the case that either the table or the methodology is incorrect or incomplete.

Logically Invalid Interpretation. The manuscript text makes a specific statement based on the values in the table. The analysis of the values in the table is mathematically, computationally, or logically incorrect. There is a clear contradiction between the statement and the values. This includes claims that fail to match the exact values in the table, comparisons that are mathematically incorrect, or more complex reasoning over the table values that can be demonstrated to be false. It does not include claims that are debatable due to the imprecision in language or ambiguity in how the statement maps to the values in the table.

Semantic Misinterpretation. The manuscript text makes a vague qualitative statement based on the values in the table. This statement may be weakly supported or weakly contradicted by the values in the table. The statement may be misleading or exaggerated based on the values presented in the table, or be a biased interpretation that supports the author's views. However, due to imprecision or ambiguity in the manuscript text or the values of the table, it is not possible to formally prove that the statement is incorrect.

Suboptimal Methodology. The manuscript text describes the research methodology that is followed. Based on the text description and the values in the table, the methodology is suboptimal and alternative steps could be better. This includes cases where the authors make an experimental design decision based on partial data, but do not select the best option based on the data. It also includes cases such as evaluating on a partial dataset or performing comparisons between methods that are misleading or unfair due to differences between them. Crucially, the values in the table do not contradict the methodology, but imply that improvements could have been made to the methodology.

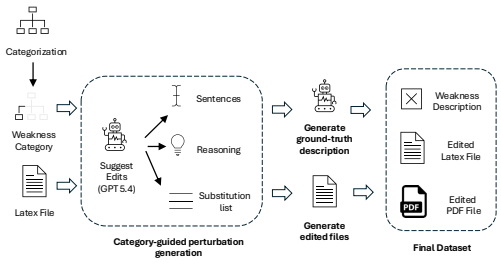

Figure 2: Overview of our perturbation pipeline for introducing errors into scientific manuscripts.

Contradiction with Established Knowledge. The manuscript text makes a statement that is assumed to be true as it is supported by reliable sources of evidence instead of the data collected for this particular work. Sources of evidence may include common domain knowledge, prior work, or mathematical proofs. The values in the table, however, appear to contradict this statement. That is, there would appear to be a contradiction between the data in this work and the established knowledge that was assumed to be correct. This may indicate that there is an issue with the statement or with the methodology and execution of the work.

4 Guided Perturbation Pipeline

In this section, we describe our pipeline for introducing guided weaknesses into scientific manuscripts through perturbations.

4.1 Pipeline Stages

Our pipeline accepts LaTeX files and a weakness category as inputs, and produces LaTeX and PDF files with the specified weakness inserted into them.

Source Data. Our pipeline performs perturbations over the LaTeX source code of a manuscript. As human evaluation would be required to validate the performance of our pipeline, we selected manuscripts based on the expertise of our annotators; in this case, we chose to collect papers accepted to EMNLP 2025, which was the latest ACL conference at the time of data collection.

We identified papers that were accepted to EMNLP 2025 then downloaded the LaTeX source from arXiv. Projects that successfully compiled were retained while others were discarded. We

use a total of 60 manuscripts for our dataset, while additional manuscripts were used during pilot annotations and pipeline development. All data was in English.

We do not distribute the original LaTeX and PDF files to comply with intended use and avoid violating licensing terms, but we provide an identifier for the original manuscript, our perturbations, judgements, and pipeline code to reproduce results. We did not discover any personally-identifiable information in our human-validated perturbations as our edits are constrained to tables.

Category guided perturbation generation. To generate scientific manuscripts with specific table-text weaknesses, we use GPT-5.4 to suggest perturbations that can be applied to the tables of a manuscript to introduce the specified weakness.

We prompt the model with the full list of categories and definitions of table-text weaknesses identified earlier (as described in Section 2.1), the specific type of weakness we want to generate, the full (flattened) LaTeX source of the manuscript, and images of each page of the corresponding PDF file. We provide the PDF as a grid of images, similar to the evaluation procedure in PRISMM-Bench (Selch et al., 2026) and MMLongBench-Doc (Ma et al., 2024). This allows the model to employ visual information from the manuscript, such as figures, as well as the unstructured visual image of the table to suggest its edits. Additionally, we provide the model with one to two in-context learning examples demonstrating table-text weaknesses in the specified category.

The model is instructed to output up to three sentences from the manuscript that can be used to verify the existence of the weakness, a list of substitutions that must be performed on the LaTeX to introduce the weakness, and reasoning that demonstrates why performing these edits would introduce the required type of weakness. The prompt also instructs the model to avoid introducing other types of weaknesses, as well as exact match discrepancies with the manuscript text, figures, or tables.

Generating edited files. After the model has generated suggested edits, we perform a simple validation to ensure that the sentences are actually from the manuscript and not hallucinated, and that the suggested edits are performed over one or more tables in the manuscript. If all suggested edits pass this validation step, we perform the substitutions in the original LaTeX and then recompile the LaTeX source. If compilation fails, we consider it a failure and do not include it in the dataset.

Generating ground-truth description. After the model suggests edits, and the edits have been validated, we generate a ground-truth description of the weakness in the edited manuscript. We provide the model with all of the output generated in the perturbation step (the relevant manuscript context, edit list, and reasoning) as well as the original weakness category and LaTeX source code, and prompt it to generate a description of the weakness that was inserted into the manuscript. The output description serves as our gold-standard weakness description.

Final Dataset Our final dataset consists of edited LaTeX files, edited PDF files, and a ground-truth description of the weakness in the manuscript. Each original manuscript is processed through our pipeline for each category - the resulting dataset has 270 instances which passed all automatic validation steps.

4.2 Human Validation

Category	Instances	Percent Agreement	Gwet's AC1
ConEK	4	1.00	1.00
LogII	3	0.67	0.54
MetRC	4	1.00	1.00
SemM	4	0.75	0.68
SubM	4	1.00	1.00
Overall	19	0.90	0.88

Table 2: **Inter-Annotator Agreement**. We provide inter-annotator agreement scores for each category and the overall agreement. We provide agreement as percentage agreement and Gwet's AC1.

Category	Instances	Skipped	Successful	Success Rate	(excl. Skipped)
ConEK	33	9	23	0.70	0.96
LogII	33	5	27	0.82	0.96
MetRC	33	3	29	0.88	0.97
SemM	33	2	29	0.88	0.94
SubM	33	4	13	0.39	0.45
Overall	165	23	121	0.73	0.85

Table 3: **Human Validated Subset**. We provide inter-annotator agreement scores for each category and the overall agreement. We provide agreement as percentage agreement and Gwet's AC1.

To ensure that our guided perturbation pipeline was performing correctly, we performed extensive human validations. The purpose of this validation was to ensure that the perturbations suggested by the model were introducing a table-text weakness into the manuscript, and that the weakness met the

required definition. This process was conducted by the same two annotators that collaboratively discovered the categorization defined in Section 2.1.

The annotators were presented with the output produced by the edit model, the original manuscript, and the edited manuscript. The annotators then had to label each instance they received with a binary label, indicating whether the weakness was correctly introduced or not. We conducted our validation in three main stages, the full details of which can be found in Appendix A.

We provide our final agreement metrics on 20 tasks that were labeled by both annotators. We provide percentage agreement and Gwet's AC1, which provides a more accurate measure of agreement in for unbalanced cases. Our final IAA scores are provided in Table 2.

Finally, one annotator labeled an additional 33 instances to measure the guided perturbation pipeline's success rate. We present the results in Table 3. We provide the success rate when treating skipped examples as failed edits, as well as the success rate when we exclude these instances. Additional details and analysis on this can be found in Appendix A.

5 Experiments

5.1 Models

We restricted the models we evaluated on to large vision-language models with reasoning capabilities as these have demonstrated strong performance on scientific understanding tasks (Selch et al., 2026). We also required models to have a training cutoff date before November 4th, 2025 to minimize the impact of contamination on our dataset. Furthermore, we required models to have a context length of at least 256k to ensure that the LaTeX source and images can fit, as well as additional context for reasoning output.

Based on these restrictions, we chose to perform our evaluations on the GPT-5.4 families of models (which have a cutoff date of August 2025) using high reasoning settings. Furthermore, we evaluate on models from the Gemini family as they are frequently used in peer-review tasks due to their support for long context lengths (Chang et al., 2025). We evaluate on Gemini 2.5-flash and Gemini 2.5-flash-lite, which were released a few months prior to our required cutoff date. These were evaluated without reasoning to provide a comparison.

All models were accessed with the official API with default temperature and settings.

5.2 Evaluation Protocol

To measure the semantic similarity between the generated and ground truth weakness, we use LLM-as-a-judge (Zheng et al., 2023). We adapt the prompt that was used in (Xu et al., 2025) for comparing two limitations, and which demonstrated very strong correlation with human judgments during validation. As our approach is comparing a longer generated report on table-text weaknesses to a single ground-truth weakness, we adapt the prompt to not penalize the generated response for containing additional details beyond the ground truth. We additionally provide results on Rouge-L and BERTScore in Appendix B.

5.3 Inputs

We evaluate models under three input settings. In the first, models receive only the LaTeX source code. In the second, models are provided only with an image grid of the PDF, following the setup used in PRISMM-Bench and MMLongBench-Doc. Finally, in the third setting, models receive both the LaTeX source and PDF images. While this multimodal setting provides the richest information—combining visual cues for error localization (e.g., perturbation artifacts) with structured tabular information from the LaTeX source—it also results in the longest context length, which may negatively impact model performance due to long-context limitations (Liu et al., 2024).

5.4 Results

Our results on the full dataset are shown in Table 5, while LLM-as-a-judge results for the full dataset and the human validated subset are in Table 4. Results are for a single run and represent the mean judgements score. See Appendix B for additional metrics and results. We make a few observations:

Models rarely benefit from visual input. For both the full dataset and the human-validated subset, the best (or tied for best) performance on *conflict with established knowledge*, *logically-invalid interpretation*, and *methodology-result conflict* was achieved using only the LaTeX source as input. The main exception was the *suboptimal methodology* category, where visual input consistently improved performance for the GPT models.

Shorter context is superior to additional information. In most cases, we found that the best performance was achieved with only visual or LaTeX

Model	Full Dataset					Human Validated Subset				
	ConEK	LogII	MetRC	SemM	SubM	ConEK	LogII	MetRC	SemM	SubM
GPT 5.4										
Images	3.3	4.7	3.8	**3.6**	3.4	3.6	4.5	3.7	3.6	**3.8**
Latex	**3.7**	4.8	**4.3**	3.5	3.2	**4.0**	4.7	**4.1**	**3.8**	3.4
Images & Latex	3.6	4.8	4.2	3.5	3.4	3.8	4.7	3.9	3.6	3.5
GPT 5.4-mini										
Images	2.3	4.1	3.3	2.7	2.2	2.4	3.6	3.1	2.8	2.4
Latex	2.9	4.5	3.7	2.6	2.1	2.8	4.4	3.6	2.6	2.0
Images & Latex	2.9	4.2	3.7	2.9	2.4	3.0	3.9	3.6	3.1	2.8
GPT 5.4-nano										
Images	2.0	3.0	2.4	2.0	1.7	2.1	3.0	2.3	2.1	1.4
Latex	2.4	4.3	3.3	2.4	1.6	2.6	4.3	3.2	2.6	1.4
Images & Latex	2.2	4.2	3.1	2.6	1.6	2.1	4.0	3.0	2.8	1.4
Gemini 2.5 Flash										
Images	1.7	2.1	1.6	1.8	1.3	1.7	2.0	1.6	1.8	1.2
Latex	2.2	3.9	2.6	2.9	2.5	2.4	3.8	2.7	3.0	2.0
Images & Latex	2.2	4.0	2.8	2.8	2.4	2.5	3.8	2.9	3.1	2.5
Gemini 2.5 Flash-lite										
Images	1.7	2.0	1.9	1.9	1.5	1.8	1.9	2.1	1.9	1.8
Latex	1.9	2.8	2.0	2.5	2.1	2.0	2.7	2.0	2.8	2.2
Images & Latex	1.9	2.7	2.2	2.4	2.0	1.8	2.5	2.1	2.6	1.7

Table 4: **LLM-as-a-judge Evaluations on Automated Dataset and Human Validated Subset**. Results using GPT 5.4 as a judge on our entire synthetic dataset. **Bold** scores represent the best score for an error category across input types and models, while underlined scores indicate a tie for best score.

input. Providing both provided marginal improvements at best, and usually performed worse than either individually. This suggests that models are struggling to utilize the full context for these kinds of scientific reasoning tasks, and instead benefit from reasoning over smaller contexts.

Large variation in performance based on weakness type. We find that all models performed the best on detecting weaknesses in the *logically-invalid interpretation* category. Most of the models achieved a score of 4 or higher in this category, indicating that the model would detect the weakness, but perhaps not at the same level of detail as the ground-truth. These results would strongly indicate that models could be used in practice to provide writing assistance or aid peer-reviewers in detecting such weaknesses in manuscripts. While the scores for *methodology-result conflict* are lower, this was a weakness category that our annotators found difficult to detect even when provided with the text context and edits.

Results are mainly consistent across the full dataset and human validated subset. Most of the patterns that we identified were consistent across both the full dataset and human validated subset. Performance on identifying weaknesses in the *logically-invalid interpretation* category, for example, was the highest across all categories for all models. LaTeX source-only input still demonstrated strong performance, despite the potential for additional cues from visual artifacts.

The consistency of our results support our approach as a scalable method for extracting insight on model performance on scientific understanding tasks. As our method can be easily extended to additional domains, it provides an opportunity for researchers to explore the use of models on identifying table-text weaknesses in their own field.

6 Conclusion and Future Work

In this work, we have introduced the task of *table-text weakness detection* in scientific manuscripts. We identified five main types of weaknesses through our investigation, and introduced a pipeline for inserting specific types of weaknesses into scientific manuscripts. We evaluated models on their ability to detect these errors, and discovered that most models perform best at detecting weaknesses in the *logically-invalid interpretation* category. Our results suggest that models could be incorporated into peer-review workflows to help detect such weaknesses in scientific manuscripts.

In future work, we plan to extend our framework to additional scientific domains and multilingual settings, as well as studying human–AI collaborative peer-review workflows.

7 Limitations

Our work has several limitations. The first is that we do not use any open-source models in our experiments. While this was initially intended to allow researchers from other domains to easily replicated our methodology, even without significant technical expertise, we could have provided additional results on such open-source models.

Another limitation is that our dataset only uses data from EMNLP 2025. While this was to prevent contamination in model training data from influencing the results, it is possible that our findings may not generalize to other conferences.

8 Ethics and Broader Impact

Our work investigates whether large language models can assist scientific peer review by detecting table–text weaknesses in manuscripts. While such systems may help reviewers identify inconsistencies and reduce workload, they should be viewed strictly as assistive tools rather than replacements for human judgment. Our results show substantial variation across weakness categories, indicating that current models remain unreliable for fully automated review.

Our perturbation pipeline automatically introduces weaknesses into scientific manuscripts for benchmarking purposes. Although intended for controlled evaluation, similar techniques could potentially be misused to generate misleading scientific content. To reduce this risk, the framework is designed for research settings and pairs perturbations with explicit annotations and weakness descriptions.

The benchmark may also reflect biases from its source data and annotation process. Since the dataset is constructed from EMNLP 2025 papers and peer-review practices in NLP, the discovered weakness categories may not generalize to other scientific disciplines. Future work should expand the framework to broader domains and more diverse reviewer communities.

Finally, our experiments rely on proprietary API-based models, which raises concerns regarding reproducibility, accessibility, and environmental cost. We encourage future research on open-source and computationally efficient alternatives for scientific reasoning and peer-review assistance.

References

Osama Mohammed Afzal, Preslav Nakov, Tom Hope, and Iryna Gurevych. 2026. Beyond "Not Novel Enough": Enriching Scholarly Critique with LLM-Assisted Feedback. In *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2648–2671, Rabat, Morocco. Association for Computational Linguistics.

Virginia Braun and Victoria Clarke. 2006. Using thematic analysis in psychology. *Qualitative Research in Psychology*, 3(2):77–101.

Yuan Chang, Ziyue Li, Hengyuan Zhang, Yuanbo Kong, Yanru Wu, Hayden Kwok-Hay So, Zhijiang Guo, Liya Zhu, and Ngai Wong. 2025. TreeReview: A Dynamic Tree of Questions Framework for Deep and Efficient LLM-based Scientific Peer Review. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 15651–15682, Suzhou, China. Association for Computational Linguistics.

Wenhu Chen, Hongmin Wang, Jianshu Chen, Yunkai Zhang, Hong Wang, Shiyang Li, Xiyou Zhou, and William Yang Wang. 2020. TabFact: A Large-scale Dataset for Table-based Fact Verification. In *International Conference on Learning Representations*.

Vivek Gupta, Maitrey Mehta, Pegah Nokhiz, and Vivek Srikumar. 2020. INFOTABS: Inference on tables as semi-structured data. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2309–2324, Online. Association for Computational Linguistics.

Xanh Ho, Yun-Ang Wu, Sunisth Kumar, Tian Cheng Xia, Florian Boudin, Andre Greiner-Petter, and Akiko Aizawa. 2026. Sciclaimeval: Cross-modal claim verification in scientific papers. In *Proceedings of the 15th Language Resources and Evaluation Conference (LREC 2026)*, Palma de Mallorca, Spain. ELRA Language Resources Association.

Xinyu Hua, Mitko Nikolov, Nikhil Badugu, and Lu Wang. 2019. Argument mining for understanding peer reviews. 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 2131–2137, Minneapolis, Minnesota. Association for Computational Linguistics.

Ilia Kuznetsov, Osama Mohammed Afzal, Koen Dercksen, Nils Dycke, Alexander Goldberg, Tom Hope, Dirk Hovy, Jonathan K. Kummerfeld, Anne Lauscher, Kevin Leyton-Brown, Sheng Lu, Mausam, Margot Mieskes, Aurélie Névéol, Danish Pruthi, Lizhen Qu, Roy Schwartz, Noah A. Smith, Thamar Solorio, and 5 others. 2024. What can natural language processing do for peer review? *Preprint*, arXiv:2405.06563.

Ilia Kuznetsov, Jan Buchmann, Max Eichler, and Iryna Gurevych. 2022. Revise and resubmit: An intertextual model of text-based collaboration in peer review. *Computational Linguistics*, 48(4):949–986.

Yash Kumar Lal, Manikanta Bandham, Mohammad Saqib Hasan, Apoorva Kashi, Mahnaz Koupaee, and Niranjan Balasubramanian. 2025. MuSciClaims: Multimodal scientific claim verification. In *Proceedings of the 14th International Joint Conference on Natural Language Processing and the 4th Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics*, pages 3285–3307, Mumbai, India. The Asian Federation of Natural Language Processing and The Association for Computational Linguistics.

Yoonjoo Lee, Kyungjae Lee, Sunghyun Park, Dasol Hwang, Jaehyeon Kim, Hong-in Lee, and Moontae Lee. 2023. QASA: advanced question answering on scientific articles. In *Proceedings of the 40th International Conference on Machine Learning*, ICML'23. JMLR.org.

Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. 2024. Lost in the Middle: How Language Models Use Long Contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173.

Xinyuan Lu, Liangming Pan, Qian Liu, Preslav Nakov, and Min-Yen Kan. 2023. SCITAB: A Challenging Benchmark for Compositional Reasoning and Claim Verification on Scientific Tables. In *The 2023 Conference on Empirical Methods in Natural Language Processing*.

Yubo Ma, Yuhang Zang, Liangyu Chen, Meiqi Chen, Yizhu Jiao, Xinze Li, Xinyuan Lu, Ziyu Liu, Yan Ma, Xiaoyi Dong, Pan Zhang, Liangming Pan, Yu-Gang Jiang, Jiaqi Wang, Yixin Cao, and Aixin Sun. 2024. MMLongBench-Doc: Benchmarking Long-context Document Understanding with Visualizations. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

Preslav Nakov, David Corney, Maram Hasanain, Firoj Alam, Tamer Elsayed, Alberto Barrón-Cedeño, Paolo Papotti, Shaden Shaar, and Giovanni Da San Martino. 2021. Automated Fact-Checking for Assisting Human Fact-Checkers. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21*, pages 4551–4558. International Joint Conferences on Artificial Intelligence Organization. Survey Track.

Shraman Pramanick, Rama Chellappa, and Subhashini Venugopalan. 2024. SPIQA: A Dataset for Multimodal Question Answering on Scientific Papers. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

Abdelrahman Sadallah, Tim Baumgärtner, Iryna Gurevych, and Ted Briscoe. 2025. The good, the bad and the constructive: Automatically measuring peer review's utility for authors. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 28991–29021, Suzhou, China. Association for Computational Linguistics.

Lukas Selch, Yufang Hou, Muhammad Jehanzeb Mirza, Sivan Doveh, James R. Glass, Rogerio Feris, and Wei Lin. 2026. PRISMM-Bench: A Benchmark of Peer-Review Grounded Multimodal Inconsistencies. In *The Fourteenth International Conference on Learning Representations*.

Xuemei Tang, Xufeng Duan, and Zhenguang Cai. 2025. Large language models for automated literature review: An evaluation of reference generation, abstract writing, and review composition. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 1602–1617, Suzhou, China. Association for Computational Linguistics.

David Wadden, Shanchuan Lin, Kyle Lo, Lucy Lu Wang, Madeleine van Zuylen, Arman Cohan, and Hannaneh Hajishirzi. 2020. Fact or fiction: Verifying scientific claims. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7534–7550, Online. Association for Computational Linguistics.

Chengye Wang, Yifei Shen, Zexi Kuang, Arman Cohan, and Yilun Zhao. 2025. SciVer: Evaluating Foundation Models for Multimodal Scientific Claim Verification. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8562–8579, Vienna, Austria. Association for Computational Linguistics.

Nancy X. R. Wang, Diwakar Mahajan, Marina Danilevsky, and Sara Rosenthal. 2021. SemEval-2021 Task 9: Fact Verification and Evidence Finding for Tabular Data in Scientific Documents (SEM-TAB-FACTS). In *Proceedings of the 15th International Workshop on Semantic Evaluation (SemEval-2021)*, pages 317–326, Online. Association for Computational Linguistics.

Sarina Xi, Vishisht Rao, Justin Payan, and Nihar B. Shah. 2025. FLAWS: A Benchmark for Error Identification and Localization in Scientific Papers. *Preprint*, arXiv:2511.21843.

Zhijian Xu, Yilun Zhao, Manasi Patwardhan, Lovekesh Vig, and Arman Cohan. 2025. Can LLMs Identify Critical Limitations within Scientific Research? A Systematic Evaluation on AI Research Papers. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 20652–20706, Vienna, Austria. Association for Computational Linguistics.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, Hao Zhang, Joseph Gonzalez, and Ion Stoica. 2023. Judging LLM-as-a-Judge with MT-Bench and Chatbot Arena.

In *Advances in Neural Information Processing Systems*, volume 36, pages 46595–46623. Curran Associates, Inc.

A Human Validation

In this section, we provide the full details of our human validation process.

Pilot Annotations. The annotators focused on resolving any ambiguities in the task, category definitions, or input context through several rounds of small, pilot annotations. During this stage, any ambiguities with the category definitions were carefully resolved by referencing the original data, as mentioned in Section 3.1. The annotators also determined if additional information or context was required beyond what was originally provided.

During this stage, it was determined that providing the model reasoning was critical. Without the model reasoning, the annotators struggled to identify certain types of weaknesses. This was particularly important for the *methodology-result conflict* category, as even with the correct context and provided edits, the annotators were unable to determine when a small change in a cell value could make it an impossible value.

Finally, at the end of this stage a set of detailed annotation guidelines were produced, enabling our study to be extended to different domains and to reproduce our validation results. Annotation guidelines will be shared alongside our data and code.

IAA Study. The two annotators labeled a set of 20 tasks by following the annotation guidelines. The annotators received the edits produced by the model, the model reasoning, extracted sentences, the weakness category, and the original and edited PDF versions of the manuscript. The annotators had to determine whether the edit successfully introduced the specified type of weakness into the category. Our annotation interface is shown in Appendix C.

As our annotations were unbalanced; that is, the pipeline was frequently performing correctly, we provide our final agreement metrics as percentage agreement and Gwet's AC1, which provides a more accurate measure of agreement in for unbalanced cases. Our final IAA scores are provided in Table 2. We observed almost perfect agreement overall, and for three categories the two annotators were in complete agreement. For the remaining two categories, the agreement was moderate to substantial.

Pipeline Performance. Once the annotation guidelines were finalized and the inter-annotator agreement demonstrated that the task was well-defined, one annotator labeled an additional 33 instances to measure the guided perturbation pipeli-

nesuccess rate. We present the results of these annotations in 3. We provide the success rate when treating skipped examples as failed edits, as well as the success rate when we exclude these instances. Success rate was generally high across categories, but lower for *conflict with established knowledge* and *suboptimal methodology*.

Analysis. For *conflict with established knowledge*, incorrect edits were low, but the skip rate was high. The annotator noted that this was primarily due to visual artifacts that were introduced during the edit process; the *conflict with established knowledge*category would frequently introduce a new table that would overlap with the manuscript text. When such instances were encountered during pilot annotations, the specified weakness could still be introduced into the manuscript, but the visual artifacts could make it easier to localize weaknesses. In the end, we chose not to filter out such cases in our automated pipeline through a post-validation step as we believed they reflected real-world use cases, particularly for writing assistance.

The automated pipeline performed the worst at introducing the *suboptimal methodology* category into manuscripts. According to the annotator, this was primarily due to the weakness being introduced into ablation tables. It was unclear if a superior methodology uncovered during the ablation stage should be considered a weakness; while the annotator chose to record these as failures, they did conform to the requirements outlined in the weakness definition.

B Additional Results

We provide the results of token-level metrics on our full dataset in Table 5 and on our human validated subset in Table 6. For Rouge-L and BERTScore we use the HuggingFace Evaluations library.

Model	Rouge-L					BERTScore (Recall)				
	ConEK	LogII	MetRC	SemM	SubM	ConEK	LogII	MetRC	SemM	SubM
GPT 5.4										
Images	0.11	0.15	0.12	0.12	0.12	0.83	0.86	0.84	0.81	0.82
Latex	0.11	0.13	0.12	0.12	0.12	0.84	0.86	0.85	0.83	0.83
Images & Latex	0.11	0.13	0.11	0.12	0.12	0.83	0.86	0.84	0.83	0.82
GPT 5.4-mini										
Images	0.13	0.19	0.15	0.15	0.14	0.83	0.86	0.84	0.83	0.83
Latex	**0.14**	**0.20**	**0.16**	0.15	0.14	0.83	0.87	0.85	0.84	0.83
Images & Latex	0.13	0.17	0.15	0.14	0.13	0.83	0.86	0.84	0.83	0.82
GPT 5.4-nano										
Images	0.12	0.17	0.13	0.13	0.13	0.83	0.85	0.83	0.82	0.82
Latex	0.13	0.18	0.14	0.13	0.12	0.84	0.87	0.85	0.84	0.83
Images & Latex	0.12	0.19	0.14	0.13	0.12	0.84	0.87	0.84	0.84	0.82
Gemini 2.5 Flash										
Images	0.09	0.09	0.09	0.09	0.10	0.81	0.82	0.82	0.81	0.81
Latex	0.11	0.13	0.10	0.12	0.12	0.83	0.84	0.83	0.83	0.83
Images & Latex	0.10	0.13	0.11	0.12	0.11	0.83	0.84	0.83	0.83	0.82
Gemini 2.5 Flash-lite										
Images	0.08	0.07	0.07	0.07	0.08	0.81	0.81	0.81	0.80	0.81
Latex	0.07	0.07	0.07	0.07	0.07	0.82	0.83	0.83	0.83	0.82
Images & Latex	0.07	0.08	0.08	0.08	0.07	0.82	0.83	0.83	0.82	0.82

Table 5: **Token-Overlap Metrics**. Results on surface-level metrics that rely on token recall. We provide results for Rouge-L and BERTScore's recall score. **Bold** scores represent the best score for an error category across input types and models, while underlined scores indicate a tie for best score.

Model	Rouge-L					BERTScore (Recall)				
	ConEK	LogII	MetRC	SemM	SubM	ConEK	LogII	MetRC	SemM	SubM
GPT 5.4										
Images	0.11	0.15	0.11	0.13	0.12	0.83	0.85	0.84	0.83	0.82
Latex	0.11	0.14	0.12	0.12	0.12	0.84	0.86	0.85	0.83	0.83
Images & Latex	0.10	0.13	0.11	0.12	0.12	0.84	0.85	0.84	0.83	0.83
GPT 5.4-mini										
Images	0.12	0.19	0.14	0.15	0.13	0.83	0.85	0.83	0.83	0.82
Latex	0.13	0.20	0.15	0.15	0.13	0.83	0.87	0.85	0.84	0.82
Images & Latex	0.13	0.17	0.14	0.14	0.13	0.83	0.86	0.84	0.83	0.82
GPT 5.4-nano										
Images	0.12	0.17	0.13	0.13	0.12	0.83	0.85	0.83	0.82	0.82
Latex	0.12	0.18	0.13	0.13	0.11	0.84	0.87	0.85	0.84	0.82
Images & Latex	0.11	0.18	0.14	0.14	0.11	0.84	0.86	0.84	0.84	0.82
Gemini 2.5 Flash										
Images	0.08	0.09	0.09	0.09	0.10	0.81	0.82	0.82	0.81	0.81
Latex	0.10	0.12	0.09	0.11	0.10	0.83	0.84	0.83	0.83	0.82
Images & Latex	0.09	0.13	0.10	0.12	0.10	0.83	0.84	0.83	0.83	0.82
Gemini 2.5 Flash-lite										
Images	0.08	0.07	0.07	0.08	0.08	0.81	0.81	0.81	0.80	0.81
Latex	0.06	0.07	0.07	0.07	0.08	0.82	0.84	0.83	0.83	0.82
Images & Latex	0.07	0.08	0.07	0.08	0.06	0.82	0.83	0.83	0.83	0.82

Table 6: **Surface-Level Token Metrics on Human Subset**. Results on surface-level metrics that rely on token recall on the human-validated subset.

C Annotation Interface

In this section, we provide screenshots of our annotation interfaces. The interface used for the qualitative investigation is shown in Figure 3. The interface used during the human validation of the perturbation pipeline is shown in Figure 4 and Figure 5.

D Annotation Guidelines

SciTabFlaw
(English Annotation Guidelines)

Introduction

As interest has grown in using LLMs to help with or automate the scientific peer-review process, it is critical that we evaluate the ability for such models to reason about and identify scientific errors. The objective of this annotation task is to construct such a benchmark for evaluating the capabilities of models to identify errors and weaknesses across modalities; specifically, across both tables and text.

Overview of Annotation Task

To produce our benchmark, we edit existing manuscripts in order to introduce specific types of errors or weaknesses into them. Your task is to determine whether the edit successfully introduces the desired error or weakness into the manuscript.
You will be provided with two manuscripts: the original manuscripts and the edited manuscripts. You will also be provided with the error or weakness category that was introduced into the manuscript, a set of sentences extracted verbatim from the manuscript, and a set of edits. Based on the edits and the manuscripts, determine whether the specified type of error has been introduced into the manuscript.

Error Categories

Methodology-Result Conflict	The manuscript text describes how data was collected, filtered, and processed before being presented in the table. However, the data in the table contradicts the described methodology. The table may be missing values, contain values that are impossible if the methodology were correctly described, or have additional unexplained values. It is important to note that the manuscript text does not claim to analyze, compare, or interpret the meaning of the values in the table; rather, based on the values of the table, it must be the case that either the table or the methodology is incorrect or incomplete.
Logically Invalid Interpretation	The manuscript text makes a specific statement based on the values in the table. The analysis of the values in the table is mathematically, computationally, or logically incorrect. There is a clear contradiction between the statement and the values. This includes claims that fail to match the exact values in the table, comparisons that are mathematically incorrect, or more complex reasoning over the table values that can be demonstrated to be false. It does not include claims that are debatable due to the imprecision in language or ambiguity in how the statement maps to the values in the table.
Semantic Misinterpretation	The manuscript text makes a vague qualitative statement based on the values in the table. This statement may be weakly supported or weakly contradicted by the values in the table. The statement may be misleading or exaggerated based on the values presented in the table, or be a biased interpretation that supports the author's views. However, due to imprecision or ambiguity in the manuscript text or the values of the table, it is not possible to formally prove that the statement is incorrect.
Suboptimal Methodology	The manuscript text describes the research methodology that is followed. Based on the text description and the values in the table, the methodology is suboptimal and alternative steps could be better. This includes cases where the authors make an experimental design decision based on partial data, but do not select the best option based on the data. It also includes cases such as evaluating on a partial dataset or performing comparisons between methods that are misleading or unfair due to differences between them. Crucially, the values in the table do not contradict the methodology, but imply that improvements could have been made to the methodology.
Contradiction with Established Knowledge	The manuscript text makes a statement that is assumed to be true as it is supported by reliable sources of evidence instead of the data collected for this particular work. Sources of evidence may include common domain knowledge, prior work, or mathematical proofs. The values in the table, however, appear to contradict this statement. That is, there would appear to be a contradiction between the data in this work and the established knowledge that was assumed to be correct. This may indicate that there is an issue with the statement or with the methodology and execution of the work.

Annotation Guidelines

You will be provided with:
a) An original manuscript.
b) A manuscript where one or more tables have been edited.
c) A set of edits that were applied to the original manuscript. Each edit consists of a unique string from the original manuscript and a second string which was used to replace the original. Edits were performed sequentially.
d) A set of sentences from the manuscripts that were originally strongly supported by the table, but should only be weakly supported or contradicted after the edit. Since edits are only performed on table, these sentences should be common across both manuscripts.
e) An error or weakness category.

Your task is to determine whether the edits correctly introduce the type of error or weakness specified into the manuscript.

Detailed guidelines are provided below, and examples clarifying specific points or boundary cases are provided in the next section.

Note: It is recommended that you use Microsoft Edge for this task, as the browser's built-in search functionality will find and scroll both PDF files at the same time, making it easier to locate tables.

****
*** Please follow the annotation guidelines below carefully.
****

At any point in the annotation process, if there is missing data or edits that do not make any changes (replacement string is same as original) press **Skip**.

1. Read the error category.
2. Read the provided sentences from the manuscript.
3. Review each edit. The parts of the replacement string that are different from the original are in **red**. Identify key information from the edits that can be used for determining which tables are modified in the manuscripts.
4. For each edit, compare the relevant parts and tables of the original manuscript and the edited manuscript. Understand the relationship between the edit and each of the provided sentences; **in particular, be careful of sentences that are taken out of context or are referencing a different figure in the manuscript**. Consider how the support for each sentence changes based on the edit.
5. Based on the provided sentences and the relevant original tables, if the type of specified error appears to already exist prior to the edit, press **Skip.**
6. If there are obvious artifacts in the edited manuscript that could indicate an error, such as problems with table boundaries or overlapping text, press **Skip.**
7. Based on the relationships between the sentences and edits, determine whether the specified type of error was introduced into the manuscript.
8. Based on the relationships between the sentences and edits, determine whether any of the other defined types of errors were introduced into the manuscript.
9. If the specified type of error was introduced, and it is sufficiently distinct from any other introduced errors, select **yes.** Otherwise, select **no.**
   a. While we do not specify the criteria for "sufficiently distinct", as a general rule of thumb: if you were writing a peer-review, would you specify these errors separately or mention them together? If it is the former, then they are sufficiently distinct.
10. If the case is particularly confusing or challenging, include any additional notes that may be useful when reviewing, discussing, or using these annotations.
11. Click **Submit**

Examples

Methodology-Result Conflict

(DARS) Reasoner+Critic		
ACC†,*	F1†,*	QWK*
0.7274	**0.7315**	0.8100
0.7136	0.7303	0.8277
0.8645	**0.6303**	**0.8326**
0.8648	**0.5988**	**0.8016**
0.5709	**0.5653**	**0.8253**
0.6071	**0.6059**	**0.7705**
0.7247	**0.6437**	**0.8113**

(DARS) Reasoner+Critic		
ACC†,*	F1†,*	QWK*
0.7274	**0.7315**	0.8100
0.7136	0.7303	0.8277
0.8645	**0.6303**	**0.8326**
0.8648	**0.5988**	**0.8016**
0.5709	**0.5653**	**0.8253**
0.6071	**0.6059**	**0.7705**
0.7402	**0.6619**	**0.8294**

Sentence

The overall performance is calculated as the average across all datasets.

Edits

```
\textbf{Overall} & 0.7515 & 0.6656 & 0.8428 & 0.6966 & 0.6038 & 0.7877 & 0.6897 & 0.5763 & 0.7900 & 0.6925
& 0.6011 & 0.7839 & 0.6141 & 0.4792& 0.6280 & \textbf{0.7247} & \textbf{0.6437} & \textbf{0.8113} \\
=>
\textbf{Overall} & 0.7515 & 0.6656 & 0.8428 & 0.6966 & 0.6038 & 0.7877 & 0.6897 & 0.5763 & 0.7900 & 0.6925
& 0.6011 & 0.7839 & 0.6141 & 0.4792& 0.6280 & \textbf{0.7402} & \textbf{0.6619} & \textbf{0.8294} \\
```

Description: The sentence states that the overall scrore (last row of the table) is calculated as the average across all datasets (each row). However, the edit changes the values for the last row for one of the methods, meaning that it is no longer the mean. This introduces a conflict between the stated methodology and the results of the conflict, meaning the type of error is correctly introduced into the manuscript.

Logically Invalid Interpretation

HanFei	7B	7.56	1.55	8.08	5.73
Small-Scale LLM (<10B)					
Deepseek-R1-1.5B	1.5B	16.00	22.16	18.18	18.78
Deepseek-R1-7B	7B	32.22	36.08	26.26	31.52
GRPO-Base	1.5B	52.00	61.86	38.38	50.75
GRPO-Law	1.5B	59.11	66.49	46.46	57.36
LexPam	1.5B	**64.89**	**69.59**	**47.47**	**60.65**
Larger-Scale LLM (≥10B)					
Deepseek-R1-32B	32B	70.22	68.56	48.48	62.42
GPT-4o-mini	-	60.00	59.28	51.52	56.93
QwQ-32B-Preview	32B	73.33	79.90	59.60	70.94
Deepseek-R1	671B	74.67	81.96	63.64	73.42

Table 2: Performance Comparison. LexPam (1.5B) aoutperforms all legal LLMs and small-scale reasoning models, and rivals larger-scale models. Bold indicates the best score within the <10B and the legal LLM groups.

Deepseek-R1-1.5B	1.5B	16.00	22.16	18.18	18.78
Deepseek-R1-7B	7B	32.22	36.08	26.26	31.52
GRPO-Base	1.5B	52.00	61.86	38.38	50.75
GRPO-Law	1.5B	59.11	66.49	46.46	57.36
LexPam	1.5B	**64.89**	**69.59**	**47.47**	**60.65**
Larger-Scale LLM (≥10B)					
Deepseek-R1-32B	32B	70.22	68.56	48.48	62.42
GPT-4o-mini	-	60.00	59.28	51.52	56.93
QwQ-32B-Preview	32B	73.33	79.90	59.60	69.94
Deepseek-R1	671B	74.67	81.96	63.64	73.42

Table 2: Performance Comparison. LexPam (1.5B) aoutperforms all legal LLMs and small-scale reasoning models, and rivals larger-scale models. Bold indicates the best score within the <10B and the legal LLM groups.

Sentence

Extensive experiments on LexNum show that existing legal LLMs perform poorly, with average accuracy below 15\%. Our proposed LexPam achieves 60.65\% accuracy, surpassing DeepSeek-R1-7B (31.52\%) and GPT-4o-mini (56.93\%), and approaching the performance of much larger models such as QwQ-32B (70.94\%) and DeepSeek-R1-671B (73.42\%).

Edits

```
QwQ-32B-Preview &32B & 73.33 & 79.90 & 59.60 & 70.94 \\
=>
QwQ-32B-Preview &32B & 73.33 & 79.90 & 59.60 & 69.94 \\
---
```

Description: The sentence states that QwQ-32b achieves a performance of 70.94%, but the introduced edit changes this to 69.94%. There are no obvious artifacts introduced with this edit. Since there is a clear contradiction between these values, this is a logically invalid interpretation.

Semantic Misinterpretation

Dataset	Hallucination Rate↓ Conf./Overall	Conf.	Overall	Correct. Rate↑	
	before	after	after	before	after
TriviaQA	23.3	**19.0**	21.2	**71.3**	70.6
NQ-Open	40.2	**26.2**	32.7	**50.7**	47.7
PopQA	33.7	**21.6**	23.2	**23.5**	21.0
Average	32.4	**22.3**	25.7	**48.5**	46.4

Dataset	Hallucination Rate↓ Conf./Overall	Conf.	Overall	Correct. Rate↑	
	before	after	after	before	after
TriviaQA	23.3	22.6	23.0	**71.3**	71.0
NQ-Open	40.2	38.9	39.6	**50.7**	50.1
PopQA	33.7	32.5	33.1	**23.5**	23.2
Average	32.4	31.3	31.9	**48.5**	48.1

Sentence

As shown in Tab.~\ref{tab:detect_results_3models}, incorporating VU alongside SU improves detection performance for all models.

Overall, the application of MUC results in a substantial reduction of confident hallucinations at the cost of a small decrease in Correctness Rate.

The observed decrease in the VU/SU Disagreement Rate, along with an increase in the Correlation Coefficient, demonstrates improved calibration and alignment between verbal and semantic uncertainties across most models and datasets.

Edits

```
\multirow{-3}{*}{\textbf{TriviaQA}} & Combined & \textbf{79.71} & 80.8 & \textbf{72.99} & \textbf{74.6} & \textbf{74.71} & \textbf{72.7} \\ \midrule & Semantic &
65.29 & 70.7 & 64.47 & 62.1 & 56.74 & 53.6 \\ & Verbal & 54.04 & 71.2 & 65.26 & 62.5 & 61.85 & \textbf{60.7} \\ \multirow{-3}{*}{\textbf{NQ-Open}} & Combined &
\textbf{66.02} & 70.3 & \textbf{68.96} & \textbf{64.8} & \textbf{62.36} & 58.7 \\ \midrule & Semantic & 71.16 & \textbf{81.2} & 66.03 & 71.1 & 53.44 & 71.7 \\ &
Verbal & 62.30 & 81.0 & 71.13 & 71.7 & \textbf{75.66} & \textbf{76.2} \\ \multirow{-3}{*}{\textbf{PopQA}} & Combined & \textbf{75.66} & 81.1 & \textbf{73.82} &
\textbf{75.7} & 75.43 & 76.0 \\
=>
\multirow{-3}{*}{\textbf{TriviaQA}} & Combined & 78.60 & 80.0 & 72.30 & 74.0 & 73.90 & 72.4 \\ \midrule & Semantic & 65.29 & 70.7 & 64.47 & 62.1 & 56.74 & 53.6 \\
& Verbal & 54.04 & 71.2 & 65.26 & 62.5 & 61.85 & \textbf{60.7} \\ \multirow{-3}{*}{\textbf{NQ-Open}} & Combined & 63.50 & 68.9 & 66.40 & 63.9 & 60.10 & 57.2 \\
\midrule & Semantic & 71.16 & \textbf{81.2} & 66.03 & 71.1 & 53.44 & 71.7 \\ & Verbal & 62.30 & 81.0 & 71.13 & 71.7 & \textbf{75.66} & \textbf{76.2} \\
\multirow{-3}{*}{\textbf{PopQA}} & Combined & 69.80 & 78.4 & 70.20 & 73.9 & 69.10 & 73.4 \\
---
TriviaQA & 23.3 & \textbf{19.0} & 21.2 & \textbf{71.3} & 70.6 & 5.4 & 8.2 & 21.50 & \textbf{21.40} & 0.59 & \textbf{0.63} & 0.50 & \textbf{0.55} & 0.16 &
\textbf{0.16} \\ NQ-Open & 40.2 & \textbf{26.2} & 32.7 & \textbf{50.7} & 47.7 & 9.1 & 19.6 & 35.10 & \textbf{18.90} & 0.38 & \textbf{0.69} & 0.37 & \textbf{0.54} &
0.17 & \textbf{0.24} \\ PopQA & 33.7 & \textbf{21.6} & 23.2 & \textbf{23.5} & 21.0 & 42.8 & 55.8 & 50.70 & \textbf{44.70} & 0.05 & \textbf{0.34} & 0.61 &
\textbf{0.73} & 0.17 & \textbf{0.20} \\ \rowcolor[HTML]{D9D9D9} Average & 32.4 & \textbf{22.3} & 25.7 & \textbf{48.5} & 46.4 & 19.1 & 27.9 & 35.80 & \textbf{28.30}
& 0.34 & \textbf{0.55} & 0.49 & \textbf{0.61} & 0.17 & \textbf{0.20} \\ \hline
=>
TriviaQA & 23.3 & 22.6 & 23.0 & \textbf{71.3} & 71.0 & 5.4 & 6.0 & 21.50 & 22.10 & 0.59 & 0.58 & 0.50 & 0.51 & 0.16 & 0.16 \\ NQ-Open & 40.2 & 38.9 & 39.6 &
\textbf{50.7} & 50.1 & 9.1 & 10.2 & 35.10 & 34.30 & 0.38 & 0.40 & 0.37 & 0.39 & 0.17 & 0.18 \\ PopQA & 33.7 & 32.5 & 33.1 & \textbf{23.5} & 23.2 & 42.8 & 43.6 &
50.70 & 49.90 & 0.05 & 0.09 & 0.61 & 0.62 & 0.17 & 0.18 \\ \rowcolor[HTML]{D9D9D9} Average & 32.4 & 31.3 & 31.9 & \textbf{48.5} & 48.1 & 19.1 & 19.9 & 35.80 &
```

Description: The second sentence states that MUC substantially reduces hallucinations with a small decrease in correctness. The introduced edit makes the after hallucination rate slightly lower than the before rate, making the "substantial reduction" debatable. Note that in this example, the edits are introducing two different errors into two tables – one of which is not a semantic misinterpretation.

Suboptimal Methodology

Model	Scale	EC	WC	TC	Avg
Legal LLM					
DISC-LawLLM	13B	12.44	2.58	18.18	11.07
fuzi.mingcha	6B	14.00	4.12	10.10	9.41
LexiLaw	6B	4.67	2.06	5.05	3.93
Tailing	7B	18.89	10.31	15.15	14.78
zhihai	7B	6.00	3.09	5.05	4.71
LawGPT_zh	6B	5.11	3.61	5.05	4.59
HanFei	7B	7.56	1.55	8.08	5.73
Small-Scale LLM (<10B)					
Deepseek-R1-1.5B	1.5B	16.00	22.16	18.18	18.78
Deepseek-R1-7B	7B	32.22	36.08	26.26	31.52
GRPO-Base	1.5B	52.00	61.86	38.38	50.75
GRPO-Law	1.5B	59.11	66.49	46.46	57.36
LexPam	1.5B	**64.89**	**69.59**	**47.47**	**60.65**
Larger-Scale LLM (≥10B)					
Deepseek-R1-32B	32B	70.22	68.56	48.48	62.42
GPT-4o-mini	-	60.00	59.28	51.52	56.93
QwQ-32B-Preview	32B	73.33	79.90	59.60	70.94
Deepseek-R1	671B	74.67	81.96	63.64	73.42

Table 2: Performance Comparison. LexPam (1.5B) aoutperforms all legal LLMs and small-scale reasoning models, and rivals larger-scale models. Bold indicates the best score within the <10B and the legal LLM groups.

QwQ-32B-Preview	32B	73.33	79.90	59.60	70.94
Deepseek-R1	671B	74.67	81.96	63.64	73.42

Table 2: Performance Comparison. LexPam (1.5B) aoutperforms all legal LLMs and small-scale reasoning models, and rivals larger-scale models. Bold indicates the best score within the <10B and the legal LLM groups.

num_generations	Avg Acc	Max compl. len
1	62.10	768
2	64.80	768
4	60.65	768
8	66.40	768

Table 3: Sensitivity of LexPam to the number of generations used in GRPO training (held-out dev set).

Sentence

For GRPO training, we set the learning rate to 1e-6, the number of generations (num_generations) to 4, and the maximum completion length to 768.

Table~\ref{tab:main results} presents accuracy scores on the LexNum benchmark, categorized into (1) legal-domain LLMs, (2) reasoning LLMs below 10B, and (3) larger-scale reasoning LLMs.

Edits

```
\begin{table}[t] \centering \resizebox{0.48\textwidth}{!}{ \begin{tabular}{cc|ccc|c} \toprule Model & Scale & EC & WC & TC &\textbf{Avg}\\ \hline \multicolumn{6}
{c}{\textbf{Legal LLM}}\\ \hline DISC-LawLLM &13B & 12.44 & 2.58 & 18.18 & 11.07 \\ fuzi.mingcha &6B & 14.00 & 4.12 & 10.10 & 9.41 \\ LexiLaw &6B & 4.67 & 2.06 &
5.05 & 3.93\\ Tailing &7B & 18.89 & 10.31 & 15.15 & 14.78\\ zhihai &7B & 6.00 & 3.09 & 5.05 & 4.71 \\ LawGPT\_zh &6B & 5.11 & 3.61 & 5.05 & 4.59\\ HanFei &7B &
7.56 & 1.55 & 8.08 & 5.73 \\ \hline \hline \multicolumn{6}{c}{\textbf{Small-Scale LLM (<10B)}}\\ \hline Deepseek-R1-1.5B &1.5B & 16.00 & 22.16 & 18.18 & 18.78 \\
Deepseek-R1-7B &7B & 32.22 & 36.08 & 26.26 & 31.52 \\ GRPO-Base&1.5B & 52.00 & 61.86 & 38.38 & 50.75 \\ \cdashline{1-6} GRPO-Law&1.5B & 59.11 & 66.49 & 46.46 &
57.36 \\ LexPam&1.5B & \textbf{64.89} & \textbf{69.59} & \textbf{47.47} & \textbf{60.65} \\ \hline \hline \multicolumn{6}{c}{\textbf{Larger-Scale LLM
($\geq$10B)}}\\ \hline Deepseek-R1-32B &32B & 70.22 & 68.56 & 48.48 & 62.42 \\ GPT-4o-mini &- & 60.00 & 59.28 & 51.52 & 56.93 \\ QwQ-32B-Preview &32B & 73.33 &
79.90 & 59.60 & 70.94 \\ Deepseek-R1 &671B & 74.67 & 81.96 & 63.64 & 73.42 \\ \bottomrule \end{tabular} } \caption{Performance Comparison. LexPam (1.5B)
aoutperforms all legal LLMs and small-scale reasoning models, and rivals larger-scale models. Bold indicates the best score within the <10B and the legal LLM
groups.} \label{tab:main results} \end{table}
=>
\begin{table}[t] \centering \resizebox{0.48\textwidth}{!}{ \begin{tabular}{cc|ccc|c} \toprule Model & Scale & EC & WC & TC &\textbf{Avg}\\ \hline \multicolumn{6}
{c}{\textbf{Legal LLM}}\\ \hline DISC-LawLLM &13B & 12.44 & 2.58 & 18.18 & 11.07 \\ fuzi.mingcha &6B & 14.00 & 4.12 & 10.10 & 9.41 \\ LexiLaw &6B & 4.67 & 2.06 &
5.05 & 3.93\\ Tailing &7B & 18.89 & 10.31 & 15.15 & 14.78\\ zhihai &7B & 6.00 & 3.09 & 5.05 & 4.71 \\ LawGPT\_zh &6B & 5.11 & 3.61 & 5.05 & 4.59\\ HanFei &7B &
7.56 & 1.55 & 8.08 & 5.73 \\ \hline \hline \multicolumn{6}{c}{\textbf{Small-Scale LLM (<10B)}}\\ \hline Deepseek-R1-1.5B &1.5B & 16.00 & 22.16 & 18.18 & 18.78 \\
Deepseek-R1-7B &7B & 32.22 & 36.08 & 26.26 & 31.52 \\ GRPO-Base&1.5B & 52.00 & 61.86 & 38.38 & 50.75 \\ \cdashline{1-6} GRPO-Law&1.5B & 59.11 & 66.49 & 46.46 &
57.36 \\ LexPam&1.5B & \textbf{64.89} & \textbf{69.59} & \textbf{47.47} & \textbf{60.65} \\ \hline \hline \multicolumn{6}{c}{\textbf{Larger-Scale LLM
($\geq$10B)}}\\ \hline Deepseek-R1-32B &32B & 70.22 & 68.56 & 48.48 & 62.42 \\ GPT-4o-mini &- & 60.00 & 59.28 & 51.52 & 56.93 \\ QwQ-32B-Preview &32B & 73.33 &
79.90 & 59.60 & 70.94 \\ Deepseek-R1 &671B & 74.67 & 81.96 & 63.64 & 73.42 \\ \bottomrule \end{tabular} } \caption{Performance Comparison. LexPam (1.5B)
aoutperforms all legal LLMs and small-scale reasoning models, and rivals larger-scale models. Bold indicates the best score within the <10B and the legal LLM
groups.} \label{tab:main results} \end{table} \begin{table}[t] \centering \resizebox{0.46\textwidth}{!}{ \begin{tabular}{l|cc} \toprule \textbf{num\_generations} &
\textbf{Avg Acc} & \textbf{Max compl. len}\\ \midrule 1 & 62.10 & 768\\ 2 & 64.80 & 768\\ 4 & 60.65 & 768\\ 8 & 66.40 & 768\\ \bottomrule \end{tabular} }
\caption{Sensitivity of LexPam to the number of generations used in GRPO training (held-out dev set).} \label{tab:numgen_sensitivity} \end{table}
---
```

Description: The manuscript text states that they set the number of generations to 4. However, the table that is inserted (and never directly referenced in the text) shows that at 2 generations the performance is better and would be computationally less expensive, making it a better option.

Contradiction with Established Knowledge

Note: Due to a lack of examples for this category, we provide an example from the original PRISMM-bench dataset. This demonstrates an error or weakness that falls into this category, but because of the differences in datasets it does not match the format for the annotation task where an edit is introduced to an existing manuscript.

Method \ Architecture	WideResNet-40-1 (0.56M) Top1 (%) ↑	ECE (%) ↓	SCE (%) ↓	ACE (%) ↓	MobileNetV2 (2.25M) Top1 (%) ↑	ECE (%) ↓	SCE (%) ↓	ACE (%) ↓
NLL	70.04	11.16	0.30	11.19	66.09	7.76	0.25	7.80
LS Szegedy et al. (2015)	70.07	1.30	0.21	1.49	66.96	4.24	0.23	4.18
CE+TS Guo et al. (2017)	70.04	2.57	**0.19**	2.50	66.09	2.33	**0.19**	2.37
MMCE Kumar et al. (2018)	69.69	7.34	0.25	7.37	62.90	3.21	0.21	3.15
MixUp Thulasidasan et al. (2019)	72.04	2.57	0.21	2.52	67.53	8.69	0.28	9.73
CRL Moon et al. (2020)	65.80	13.91	0.37	13.91	67.05	12.06	0.33	12.06
PSKD Kim et al. (2021)	**72.56**	3.73	0.20	3.72	69.09	6.95	0.23	6.94
MDCA Hebbalaguppe et al. (2022b)	68.51	1.35	0.21	1.34	66.96	1.61	0.20	1.92
AdaFocal Ghosh et al. (2022)	67.36	2.10	0.21	1.97	65.34	1.83	0.20	1.53
CPC Cheng & Vasconcelos (2022)	69.99	7.61	0.23	7.55	67.30	4.17	0.22	4.07
MbLS Liu et al. (2022)	69.97	5.37	0.22	5.37	67.22	1.25	0.20	1.25
KD(UC)	69.60	15.18	0.37	15.18	66.82	5.40	0.22	5.36
Ours (KD with MixUp)	72.48	1.21	0.20	1.17	**69.92**	2.17	0.24	2.10
Ours (KD with AdaFocal)	71.70	1.19	**0.19**	1.34	66.64	1.55	0.20	1.43
Ours (KD with CPC)	70.00	9.02	0.26	9.01	67.83	**0.88**	**0.19**	**0.95**
Ours (KD with MDCA)	71.07	**0.98**	0.20	**1.10**	67.17	1.10	0.20	1.17

Table 2: **[Large-to-small]** Comparison of calibration performance of small student models calibrated using KD(C) framework vs. SOTA calibration techniques, employing a relatively larger calibrated teacher on CIFAR100 dataset. WRN-40-2 (Zagoruyko & Komodakis (2016)) and ResNeXt-18x4 (Xie et al. (2017)) were used as teachers for WRN-40-1 (Zagoruyko & Komodakis (2016)) and MobileNetV2 (Sandler et al. (2018)) respectively as student models. For ECE/SCE computation, 15 bins were used in accordance with prior work. ACE uses an adaptive binning strategy. For full results refer to the supplementary . **Numbers in bold:** best performance; underlined: second best. KD(C) gives best all around performance, in one instance PSKD is slightly better than KD(C) in accuracy, but lags much behind in calibration metric (the focus of this paper).

Part 1

since post-hoc methods are known to result in relatively inferior calibration performance

Description: The sentence states that post-hoc methods are known to result in relatively inferior calibration performance. However, the results demonstrate that CE+TS (a post-hoc method) outperforms many training-time calibration methods.

Row Content

Part 0

	CIFAR-10			
Methods	1 Epoch	10 Epochs	30 Epochs	1000 Epochs
SimCLR	0.282	0.565	0.663	0.910
BYOL	0.249	0.489	0.684	0.926
VICReg	0.406	0.697	0.781	0.921
SwAW	0.245	0.532	0.767	0.923
EMP-SSL(20 patches)	0.806	0.907	0.931	\
EMP-SMILE(20 patches)	0.315	0.242	0.325	\
EMP-InfoNCE(20 patches)	0.753	0.789	0.818	\
EMP-MILE(20 patches)	0.768	0.802	0.859	\

Table 1: Performance of the MI based SSL and standard self-supervised SOTA methods with different epochs. Accuracy is measured by training linear classifier on learned representation.

Review text

Table 1: The table presents results only for 300 epochs, while the text discusses performance at 1000 epochs and expects those results; 1000-epoch results are missing from the table.

Review description

Table 1 is supposed to show a performance comparison between the suggested method and existing methods, but while the existing methods go until 1000 epochs, the suggested methods only until 30 epochs, making a comparison at 1000 epochs not possible.

> View full PDF

Record response

labels

Add options

description

submit

Figure 3: Qualitative investigation interface.

Figure 4: Perturbation pipeline validation interface (top).

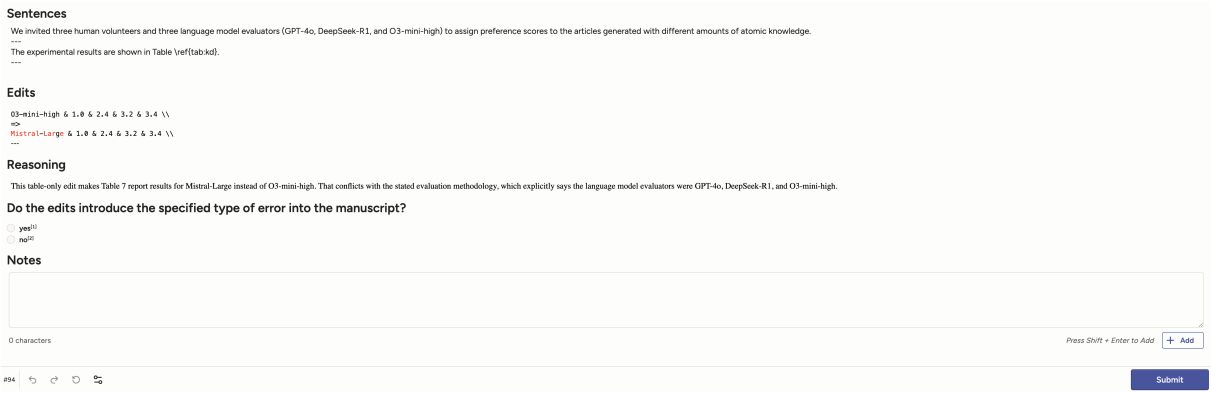

Figure 5: Perturbation pipeline validation interface (bottom).