SciAudit: Benchmarking MLLMs for Scientific Error Detection

Anonymous authors
Paper under double-blind review

Abstract

The rapid growth of scientific output makes it increasingly difficult for peer reviewers to examine every paper in depth. As a result, substantive errors or inconsistencies may be missed, leading to monetary, reputational, or scientific costs. Multimodal large language models (MLLMs) could assist scientific review, but existing error detection benchmarks remain limited in data scale, evaluation granularity, and task openness. We introduce SciAudit, a benchmark for open-ended scientific error detection in full papers. SciAudit comprises two complementary subsets spanning 16 scientific domains: Gold contains 2938 human-verified issues from 293 problematic papers, while Silver contains 5441 controlled errors constructed from 527 high-quality papers through a multi-model *Draft–Check–Refine* pipeline. Each error is annotated with its location, supporting evidence, reasoning, and type. These annotations support quality-aware metrics that assess error detection and the quality of the intermediate reasoning process. We evaluate 13 MLLMs under thinking and non-thinking settings. Results show that current MLLMs remain limited in open-ended error detection, with even the strongest model achieving less than 50% under our quality-aware F1 evaluation. Additional thinking generally improves performance, but errors requiring composite capabilities or multi-step reasoning remain challenging. Furthermore, we find that models can correctly verify 52.5%–81.3% of previously missed errors when explicitly prompted about them. This suggests that many detection failures stem from limitations in error discovery and reasoning, rather than insufficient knowledge alone. We hope SciAudit can provide a rigorous testbed for evaluating MLLMs and developing more reliable agent systems for scientific auditing in the future.

1 Introduction

As scientific output continues to grow, peer review plays an important role in finding errors and ensuring research quality (Aczel et al., 2025). However, limited reviewer capacity makes it difficult to thoroughly inspect every paper, so substantive errors may be missed (Hanson et al., 2024). These errors can cause monetary, reputational, or scientific costs (Lovering et al., 2025). Recent advances in multimodal large language models (MLLMs) offer new ways to assist this auditing process. This raises a key question: ***Can MLLMs autonomously and reliably detect scientific errors in papers?***

As shown in Figure 1, scientific error detection requires models to search the full paper for relevant evidence and connect information across text and figures to identify and verify potential errors. More importantly, in real-world peer review, potential errors are not specified in advance: reviewers must examine the paper without explicit guidance and independently identify inconsistencies, incorrect conclusions, and other substantive problems. Therefore, the challenge of scientific error detection lies not only in integrating distributed evidence to verify potential errors, but also in autonomously discovering potential issues without explicit guidance.

Recent benchmarks have made progress toward evaluating scientific error detection, but they typically address only part of the problem. A realistic benchmark should require models to detect errors openly from full papers, while also covering errors that reflect real scientific practice at sufficient scale. ScholScan (Li et al., 2026a) and PRISMM-Bench (Selch et al., 2026) evaluate error detection through question–answer pairs tied to specific sections or locations. This constrains annotations to well-posed questions and does not measure open-ended error detection in realistic reviewing

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

scenarios. FLAWS (Xi et al., 2025), FIND (Lovering et al., 2025), and PaperAudit-Bench (Tu et al., 2026) evaluate error detection using constructed or manually injected errors and inconsistencies, which may not fully reflect errors in real papers. SPOT (Son et al., 2025) collects 91 confirmed errors from real papers and evaluates whether models can rediscover them from the original papers, but its limited scale constrains broad and comprehensive evaluation. In addition, most existing benchmarks focus only on whether an error is detected, without evaluating the intermediate reasoning process.

To address these limitations, we introduce the **SciAudit** benchmark. It has three key features:

Open-ended Detection. Models examine the entire paper to identify multiple scientific errors that may occur across different sections or locations, rather than detecting errors within a well-posed part of the paper.

Comprehensive Coverage. SciAudit provides two complementary evaluation subsets spanning 16 scientific domains. Gold contains 2938 human-verified issues from 293 problematic papers, while Silver contains 5441 controlled errors across 527 high-quality papers. Together, they combine authenticity with sufficient scale and diversity for comprehensive evaluation.

Quality-Aware Evaluation. Each error is annotated with its location, supporting evidence, reasoning trace, and error type. Predictions are evaluated along four dimensions. This evaluation captures not only whether an error is detected, but also the reliability of its localization, evidence, reasoning, and type classification.

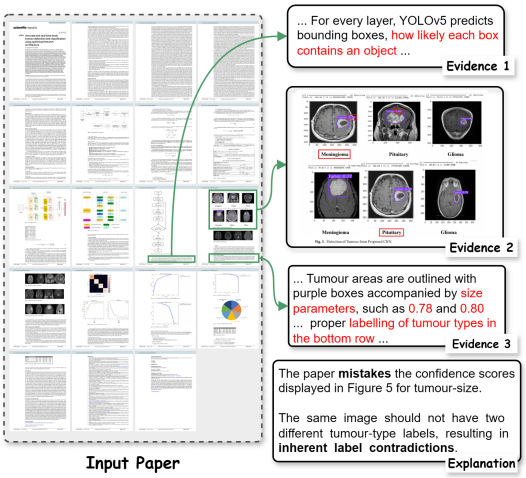


Figure 1: Example of scientific error detection in full papers through cross-section and multimodal evidence integration.

Our main contributions are as follows:

- **Benchmarking scientific error detection.** We construct SciAudit to evaluate paper-level scientific error detection without predefined inspection directions, combining human-verified issues with controlled injected errors across 16 scientific domains.
- **Quality-aware evaluation framework.** We replace binary true-positive counting with quality-weighted scoring across four dimensions: error localization, supporting evidence, reasoning coherence, and error type. The framework assesses both error detection and prediction reliability.
- **Systematic diagnosis of MLLMs.** We evaluate 13 representative MLLMs under both thinking and non-thinking settings. All models struggle to independently identify errors in full papers, yet many missed errors can be judged accurately when explicitly provided. For example, GPT-5.4 identifies only 35.5% of errors with high quality on average, but correctly verifies 81.3% of missed errors when given explicit candidates. This suggests that many missed errors are within the models' understanding, but they still struggle to autonomously discover and reason about them.

2 Related Work

2.1 Multimodal large language models

Multimodal large language models (MLLMs) integrate visual perception with language modeling, enabling joint understanding and reasoning over text and visual content. Early work such as LLaVA (Liu et al., 2023) established visual instruction tuning as a major paradigm for multimodal alignment. Recent models focus on long-context understanding and complex multimodal reasoning. Qwen3-VL (Bai et al., 2025) supports long interleaved multimodal contexts, while InternVL3.5 (Wang et al., 2025) strengthens reasoning through reinforcement learning and adaptive visual resolution. Document-oriented models such as DocThinker (Yu et al., 2025) and DocR1 (Xiong et al., 2026) further improve structured reasoning and multi-page evidence localization. These advances provide the foundation for scientific error detection, where evidence is often distributed across text, figures, tables, and other elements, demanding cross-modal and cross-section reasoning.

Table 1: Comparison with scientific error detection benchmarks. **#Source**, **#Doc**, **#Annotation**, and **#Domain**: The number of data sources, documents, annotations, and domains. **Open-ended Detection**: Models need to detect multiple errors across the full paper. **Fine-grained Evaluation**: Fine-grained evaluation of the intermediate process. **Real-world Sample**: Includes samples from real-world scientific practice.

Benchmark	Data Scale				Open-ended Detection	Fine-grained Evaluation	Real-world Sample
	#Source	#Doc	#Annotation	#Domain			
FLAWS	1	713	713	1	✓	✗	✗
FIND	5	500	500	–	✓	✓	✗
PRISMM-Bench	1	242	262	1	✗	✗	✓
ScholScan	2	715	1800	13	✗	✓	✓
SPOT	2	83	91	10	✓	✗	✓
PaperAudit-Bench	1	220	∼2200	1	✓	✗	✗
SciAudit	**7**	**820**	**8379**	**16**	✓	✓	✓

2.2 Scientific Paper Review Benchmarks

Scientific peer review assesses research validity by examining methods, evidence, and claims. Recent benchmarks evaluate LLMs on related tasks. LimitGen (Xu et al., 2025) evaluates the identification of critical limitations. TreeReview (Chang et al., 2025) assesses full-review and actionable-feedback generation. Mind the Blind Spots (Shin et al., 2025) evaluates whether LLM reviews focus on the same scientific aspects as expert reviews. ReviewBench (Li et al., 2026b) evaluates review quality using paper-specific rubrics grounded in review guidelines, paper content, and human reviews. These benchmarks mainly assess review generation, reviewer focus, or broad research limitations. Scientific error detection focuses more specifically on whether models can autonomously identify concrete and objectively verifiable errors from full papers.

2.3 Scientific Error Detection Benchmarks

Recent benchmarks explore different aspects of scientific error detection. FLAWS (Xi et al., 2025) evaluates whether models can locate constructed errors that invalidate key claims in full papers. FIND (Lovering et al., 2025) tests contradiction detection in long technical documents using expert-designed errors. PaperAudit-Bench (Tu et al., 2026) injects multiple error types into high-quality papers to evaluate open-ended detection. These benchmarks rely on constructed errors, which may not fully reflect real scientific errors. ScholScan (Li et al., 2026a) combines generated errors with issues sampled from real peer reviews, but restricts detection to specified sections and error categories. PRISMM-Bench (Selch et al., 2026) extracts multimodal inconsistencies from real peer reviews, mainly evaluating detection through multiple-choice questions on selected paper components. SPOT (Son et al., 2025) tests whether models can rediscover 91 confirmed errors from 83 original papers, but its scale remains limited.

As shown in Table 1, SciAudit is the only benchmark in the comparison that jointly provides open-ended detection, fine-grained evaluation, and real-world errors. It also offers the broadest coverage, with 8379 annotations from 820 papers spanning 16 domains and 7 data sources. Therefore, SciAudit can provide broad and comprehensive evaluation for scientific error detection.

3 Benchmark Construction

3.1 Scientific Error Definition

SciAudit focuses on objectively verifiable scientific errors that can be solidly established through evidence in the paper, contextual reasoning, or general domain knowledge. Such errors make scientific content inaccurate or internally inconsistent. Subjective or insufficiently supported concerns, study limitations, and minor spelling, grammar, or wording issues that don't affect meaning are excluded. Inspired by ScholScan (Li et al., 2026a), we organize errors along the research workflow while adapting to real-world errors and our open-ended task setting. We categorize errors into 9 types, covering stages from question definition to reporting, as illustrated in Figure 2.

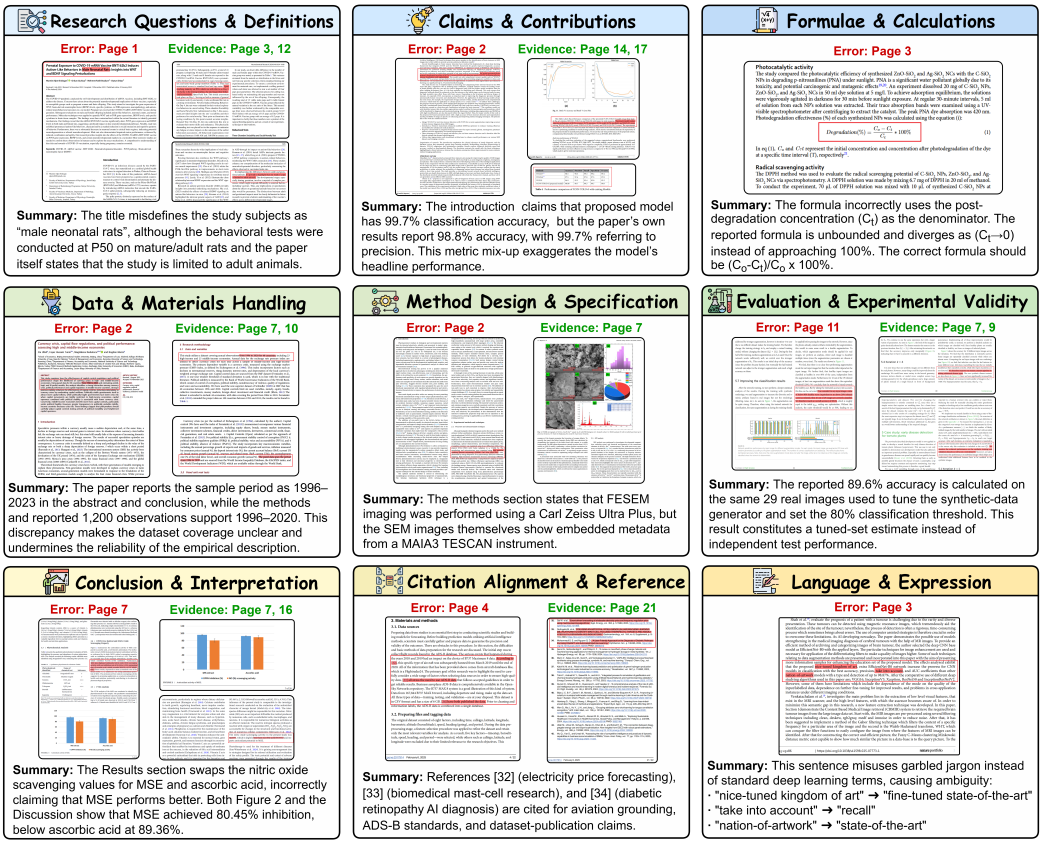


Figure 2: Representative examples of the 9 scientific error categories in SciAudit, showing the error and corresponding evidence from the paper.

3.2 Data Collection and Construction

Gold Construction. Gold is designed to capture scientific errors from real-world research practice. Public reviews provide useful signals about potential problems. We therefore jointly provide each paper and its reviews to strong MLLMs to mine potential errors with high coverage, followed by expert review for final verification. As shown in Figure 3, Gold is constructed in three stages.

(i) Problematic paper collection. We primarily collect papers published from 2024 to 2026 with associated public reviews from PubPeer (`https://pubpeer.com/`), Retraction Watch (`https://retractionwatch.com/`), and OpenReview (`https://openreview.net/`). These papers cover real-world cases such as retractions, corrections, public concerns, and rejections.

(ii) Candidate issues mining. To maximize coverage, we use three of the strongest MLLMs available at the time of construction, GPT-5.4, Gemini-3.1-Pro, and Qwen-3.6-Plus. Each model jointly analyzes the full paper and its public reviews, using the reviews as signals while independently mining the paper for potential errors. The models produce structured candidate annotations with locations, evidence, reasoning, and error types.

(iii) Expert review. After merging and deduplicating the model outputs, we obtain 3,482 candidate errors. 4 experts with peer-review experience then conduct the final review. AI tools may assist with paper reading, evidence retrieval, and cross-section comparison, but all decisions about error validity and final annotations are made by the human experts based on the paper. Each paper is independently reviewed by one expert, who verifies candidate issues and may identify missed errors during their review. If uncertainties arise, the expert consults other experts to reach a final decision.

Silver Construction. Gold provides authentic errors, but collecting and verifying real-world cases is difficult to scale. Silver is therefore designed as a scalable complement to Gold by injecting controlled errors into high-quality papers and applying multiple rounds of quality control. As shown in Figure 3, Silver is also constructed in three stages.

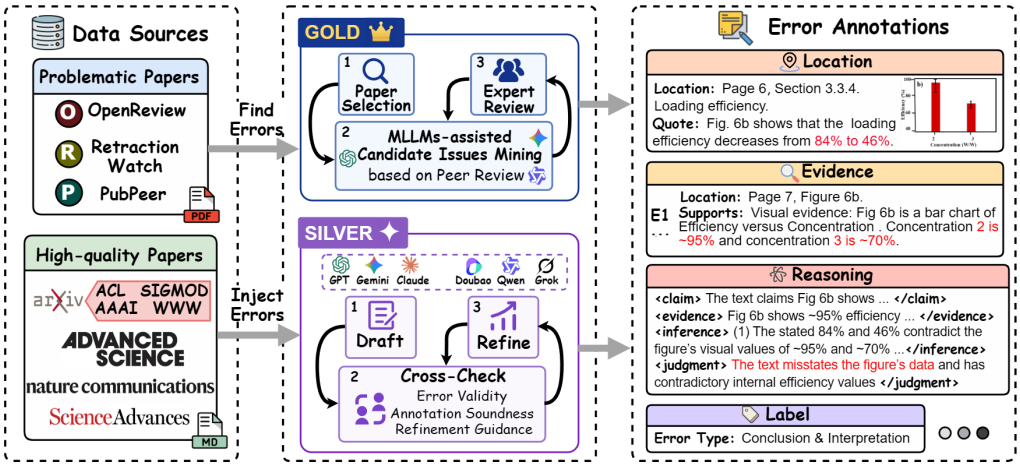


Figure 3: Overview of SciAudit data construction, covering Gold and Silver construction pipelines and the structured error annotations.

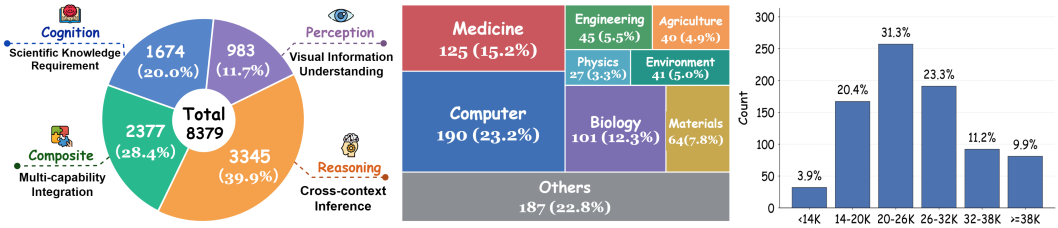


Figure 4: Dataset statistics of SciAudit, including capabilities required for error detection, domain coverage, and paper-length distribution.

(i) High-quality paper collection. To minimize pre-existing scientific errors before injecting controlled errors, we collect high-quality papers published or accepted in 2026. These include papers from *AAAI*, *ACL*, *SIGMOD*, and *WWW* available on arXiv, as well as papers from *Advanced Science*, *Nature Communications*, and *Science Advances*.

(ii) Multi-model errors generation. To reduce model-specific bias and increase the diversity of generated errors, we use six strong proprietary MLLMs from different model families: GPT-5.4, Gemini-3.1-Pro, Claude-Sonnet-4.6, Doubao-Seed-2.0-Pro, Qwen-3.6-Plus, and Grok-4.3. These models are used in a three-stage *Draft–Check–Refine* pipeline. In *Draft*, each model injects errors into the Markdown parsed from the PDF by modifying content and produces structured annotations. In *Check*, the models are paired for reciprocal assessment of injected error validity and annotation soundness, providing refinement guidance. In *Refine*, the drafting model discards or revises injected errors until no further guidance is given or two rounds are completed.

(iii) Quality validation. To validate Silver quality, we randomly sample 50 papers (∼10% of Silver) and manually examine all 541 generated errors. Each error is independently evaluated by two experts on three dimensions (*Error Validity*, *Error Value*, *Annotation Correctness*) using a 0–4 scale. Across the two experts, 93.3–94.1% of errors score at least 3 for validity, 74.3–80.0% for value, and 90.0–94.1% for annotation correctness, with strong inter-evaluator agreement. These results indicate that most generated errors are valid, meaningful, and accurately annotated. See Appendix B.1 for details.

Overall, Gold contains 2,938 verified errors across 293 papers after expert review, which discards 939 candidate errors, revises 1,154, and adds 395 new ones. More details are provided in Appendix B.2. Meanwhile, the Silver pipeline produces 5,441 controlled errors across 527 papers.

Capability Categorization. To support fine-grained analysis, we categorize each error by the main capability required for detection. As shown in Figure 4, *Cognition*, *Perception*, *Reasoning*, and *Composite* require scientific knowledge, visual understanding, cross-context inference, and multi-capability integration, respectively. The four categories contain 1674, 983, 3345, and 2377 errors. The figure also summarizes the domain and paper-length distributions of SciAudit.

3.3 Quality-aware Evaluation Framework

Recall, Precision, and F1 are commonly used to evaluate the scientific error detection task (Xi et al., 2025; Son et al., 2025; Tu et al., 2026). However, these metrics treat all correct predictions as identical binary true positives, ignoring differences in prediction quality. Therefore, we propose a quality-aware evaluation framework that replaces binary counts with quality-weighted scores: a prediction no longer receives full credit merely for being correctly identified, but instead receives a weighted score based on its quality across different dimensions.

For a paper, let the annotated error set be $G = \{g_i\}_{i=1}^{m}$ and the prediction set be $P = \{p_j\}_{j=1}^{n}$. Each error instance is represented as $x = (t, c, l, e, r, s)$, where t, c, l, e, r, and s denote the error text, category, location, evidence, reasoning, and summary, respectively. Because the same error may be expressed differently, we use GPT-5.4 to semantically match each g_i and p_j based on their t, l, and s fields, yielding a one-to-one matching set $M \subseteq G \times P$. Each matched pair is then evaluated along the following four quality dimensions.

(i) ***Where*** *is the error?* **Where** measures whether the prediction identifies the correct erroneous span and its document location. We average their normalized longest-common-substring (LCStr) similarities over the error text (t) and document location (l). The metric penalizes incomplete, overextended, and misplaced predictions.

$$S_1(g,p) = \frac{\mathrm{sim}(g.t, p.t) + \mathrm{sim}(g.l, p.l)}{2}, \quad \mathrm{sim}(g.t, p.t) = \frac{|\mathrm{LCStr}(g.t, p.t)|}{\sqrt{|g.t| \cdot |p.t|}}. \tag{1}$$

(ii) ***What*** *is the evidence?* **What** measures whether the prediction provides sufficient evidence without unnecessary redundancy. GPT-5.4 performs many-to-many semantic alignment, allowing evidence items to be split or merged. Let $m_E(g,p)$ denote the number of gold evidence items covered by the prediction: $m_E(g,p) = \left|\{e_g \in g.e : \exists\, e_p \in p.e,\ e_g \sim e_p\}\right|$. The score balances evidence completeness against prediction conciseness by rewarding complete gold-evidence coverage and penalizing redundant predicted evidence. We then define the evidence score:

$$S_2(g,p) = \sqrt{\frac{m_E(g,p)}{|g.e|} \cdot \frac{m_E(g,p)}{\max\left(|p.e|, m_E(g,p)\right)}}. \tag{2}$$

(iii) ***Why*** *is it wrong?* **Why** measures whether the prediction reproduces the key reasoning process rather than merely stating the conclusion. GPT-5.4 normalizes both explanations into ordered atomic steps, which are compared using a semantic longest-common-subsequence (LCS) normalized by IoU. This penalizes omitted, extraneous, and misordered reasoning steps.

$$S_3(g,p) = \frac{\ell_r(g,p)}{|g.r| + |p.r| - \ell_r(g,p)}, \quad \ell_r(g,p) = \left|\mathrm{LCS}(g.r, p.r)\right|. \tag{3}$$

(iv) ***Which*** *error is it?* **Which** measures whether the model correctly identifies the nature of the error. For each matched gold–prediction pair, the two errors have already been semantically aligned and therefore share the same underlying error nature, earning a base score of 0.5. An exact match between the predicted and gold categories contributes the remaining 0.5.

$$S_4(g,p) = \frac{1 + \mathbb{I}[g.c = p.c]}{2}. \tag{4}$$

(v) ***Overall*** *quality-aware score.* We average the four dimension scores for each matched pair and sum the resulting scores as the quality-weighted true-positive score:

$$S(g,p) = \frac{1}{4}\sum_{k=1}^{4} S_k(g,p), \qquad \text{Q-TP} = \sum_{(g,p)\in M} S(g,p). \tag{5}$$

Finally, we replace binary true positives with Q-TP and define the Paper-level metrics as:

$$\text{Q-Recall} = \frac{\text{Q-TP}}{|G|}, \quad \text{Q-Precision} = \frac{\text{Q-TP}}{|P|}, \quad \text{Q-F1} = \frac{2 \cdot \text{Q-Precision} \cdot \text{Q-Recall}}{\text{Q-Precision} + \text{Q-Recall}}. \tag{6}$$

The above metrics require semantic matching between differently expressed errors, evidence, and reasoning steps. We use GPT-5.4 (OpenAI, 2026), the strongest model in our evaluation, for this matching. GPT-5.4 does not judge whether an error is valid or assign final scores; it only aligns semantically equivalent content and normalizes reasoning steps. The final scores are computed using predefined rules. Appendix C.1.3 shows that the results are stable across different evaluator models.

Table 2: Paper-level performance (%) comparison on the Gold and Silver sets. Each metric is reported as **thinking / non-thinking** mode. **Best** and second-best scores are identified separately within proprietary and open-source MLLMs for each metric and mode.

Models	Gold			Silver		
	Q-F1	Q-Recall	Q-Precision	Q-F1	Q-Recall	Q-Precision
Proprietary MLLMs						
Grok-4.3	15.7 / 6.3	9.8 / 4.9	48.2 / 12.6	23.2 / 4.6	14.6 / 3.3	69.4 / 10.3
Claude-Sonnet-4.6	16.6 / 12.2	10.5 / 10.2	47.7 / 18.5	24.4 / 18.8	15.6 / 12.9	**71.4** / 45.6
Gemini-3.1-Pro	22.2 / 19.6	16.3 / 13.9	40.5 / **38.5**	31.8 / 28.1	23.5 / 20.1	54.8 / 53.9
Qwen-3.6-Plus	24.4 / 14.5	17.9 / 9.9	43.1 / 31.9	33.6 / 19.7	24.3 / 12.3	59.9 / 60.2
Doubao-Seed-2.0-Pro	24.2 / 21.2	17.3 / 16.2	47.9 / 34.5	37.2 / 37.8	26.1 / 28.3	71.0 / **62.2**
GPT-5.4	**38.7** / **25.5**	**32.5** / **22.6**	**52.6** / 31.8	**47.6** / **41.2**	**38.9** / **35.5**	64.6 / 52.1
Open-source MLLMs						
Llama-4-Maverick	- / 1.9	- / 1.3	- / 5.6	- / 5.4	- / 3.4	- / 19.8
GLM-4.6V	5.2 / 4.5	3.2 / 2.8	20.0 / 17.6	8.4 / 6.5	5.1 / 3.8	35.2 / 31.8
Mistral-Small-4	10.6 / 6.9	11.3 / 8.1	11.4 / 7.0	12.6 / 8.7	11.3 / 8.5	16.8 / 10.1
Intern-S2-Preview	16.6 / 7.4	11.7 / 4.9	32.1 / 17.8	26.8 / 10.0	18.8 / 6.4	52.1 / 28.5
Gemma-4-31B-it	20.0 / 11.0	14.5 / 7.0	36.8 / **30.8**	28.1 / 17.1	19.4 / 10.4	**57.3** / **59.1**
Qwen3.5-122B-A10B	21.6 / 13.4	15.7 / 9.8	**39.3** / 24.5	31.0 / 18.7	22.2 / 12.5	55.5 / 45.8
Qwen3.5-27B	**23.3** / **14.4**	**17.8** / **10.1**	37.6 / 28.0	**31.3** / **24.4**	**23.2** / **16.1**	52.5 / 57.0

4 Experiments

4.1 Experimental Setup

Models. We evaluate 13 representative MLLMs, including 6 proprietary and 7 open-source models.

(i) Proprietary models: Grok-4.3 (Amazon Web Services, 2026), Claude-Sonnet-4.6 (Anthropic, 2026), Gemini-3.1-Pro (Google DeepMind, 2026), Qwen-3.6-Plus (Qwen Team, 2026b), Doubao-Seed-2.0-Pro (ByteDance Seed, 2026), and GPT-5.4 (OpenAI, 2026).

(ii) Open-source models: Llama-4-Maverick (Meta AI, 2025), GLM-4.6V (Team et al., 2025), Mistral-Small-4 (Mistral AI, 2026), Intern-S2-Preview (Bai et al., 2026), Gemma-4-31B-it (Gemma Team et al., 2026), Qwen3.5-122B-A10B and Qwen3.5-27B (Qwen Team, 2026a).

Input and Output Paradigm. To ensure consistent inputs and a fair comparison, we follow prior works (Tu et al., 2026; Son et al., 2025) and use dots.mocr (Zheng et al., 2026) to parse each PDF into interleaved text and figures for multimodal input, prompting MLLMs to freely identify scientific errors in a structured format.

We test both thinking and non-thinking settings where supported. To reduce generation randomness and improve reproducibility, we set temperature=0 to minimize sampling randomness, top_p=1.0 to avoid additional probability-mass truncation, and seed=42 to control random sampling where supported, while using the same task instructions for all models.

4.2 Overall Performance

We first evaluate the overall scientific error detection performance of MLLMs on Gold and Silver and examine the effect of thinking mode, with the main results summarized in Table 2 as follows:

End-to-end scientific error detection remains limited. On both Gold and Silver, no model achieves a Q-Recall above 40%. Even the strongest model, GPT-5.4, reaches only 32.5% and 38.9%, respectively. Except for GPT-5.4, closed-source models do not consistently outperform open-source models. These results show that current MLLMs still struggle to reliably identify and substantiate errors in complete scientific papers.

Table 3: Error-level Q-Recall (%) across capability categories. Errors are grouped into ***Cognition***, ***Perception***, ***Reasoning***, and ***Composite*** based on the primary capability required for detection. Values in parentheses indicate the deviation of each category score from the model's average score across the four capability categories.

Models	Cognition	Perception	Reasoning	Composite	AVG.
Proprietary MLLMs					
Grok-4.3	10.1 (−2.0)	12.6 (+0.4)	14.4 (+2.3)	11.5 (−0.7)	12.2
Claude-Sonnet-4.6	11.6 (−1.5)	13.6 (+0.5)	15.7 (+2.5)	11.6 (−1.5)	13.1
Gemini-3.1-Pro	20.8 (−1.3)	27.6 (+5.5)	17.3 (−4.7)	22.5 (+0.4)	22.0
Qwen-3.6-Plus	23.7 (+1.6)	23.0 (+0.8)	20.8 (−1.4)	21.2 (−1.0)	22.2
Doubao-Seed-2.0-Pro	22.7 (+1.0)	19.4 (−2.4)	24.4 (+2.7)	20.5 (−1.3)	21.8
GPT-5.4	**36.0** (+0.5)	**32.7** (−2.8)	**36.6** (+1.1)	**36.6** (+1.1)	**35.5**
Open-source MLLMs					
GLM-4.6V	2.2 (−1.9)	4.7 (+0.7)	5.4 (+1.4)	3.8 (−0.2)	4.0
Mistral-Small-4	8.8 (−1.5)	7.8 (−2.5)	12.5 (+2.3)	11.9 (+1.7)	10.3
Intern-S2-Preview	14.7 (−1.8)	20.9 (+4.3)	15.7 (−0.8)	14.9 (−1.7)	16.5
Gemma-4-31B-it	13.9 (−4.0)	20.9 (+3.0)	18.3 (+0.3)	18.6 (+0.7)	17.9
Qwen3.5-122B-A10B	21.2 (+0.9)	**22.6** (+2.3)	19.7 (−0.6)	17.7 (−2.6)	20.3
Qwen3.5-27B	**22.2** (+1.0)	21.6 (+0.4)	**21.3** (+0.1)	**19.7** (−1.5)	**21.2**

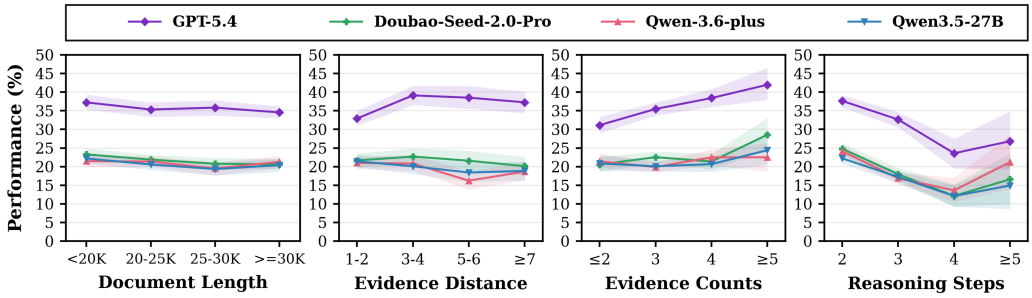


Figure 5: Performance trends across varying complexities. Shaded regions show 95% percentile-bootstrap confidence intervals from 1,000 paper-level resamples. **Evidence Distance**: Average page distance between an error and its supporting evidence.

Thinking generally improves detection. Most models perform better in thinking mode. Q-Recall improves for almost all models, suggesting fewer missed errors, while Q-Precision also increases in most cases, indicating fewer invalid predictions. Overall, additional reasoning improves both error coverage and prediction reliability.

Gold and Silver show consistent model trends. Model performance is highly consistent across the two subsets. GPT-5.4 performs best among proprietary models on both Gold and Silver, while Qwen3.5-27B performs best among open-source models. The relative rankings of other models are also largely stable. This consistency supports Silver as a scalable complement to Gold.

Additional results are provided in Appendix C.2. First, standard Recall, Precision, and F1 show trends consistent with our quality-aware metrics. Second, the *Where*, *What*, *Why*, and *Which* scores confirm that matched predictions are generally well localized, supported, reasoned, and classified. Third, model rankings remain stable across Silver subsets generated by different Draft models, with no clear self-generator advantage.

4.3 Fine-Grained Analysis

We conduct analysis to understand model failures. We examine model capability limitations, what factors affect detection, and the main causes of false positives. Finally, we ask models to re-verify missed errors to identify whether such errors are beyond their understanding or merely undiscovered.

Capability Limitation Analysis. Different errors require different model capabilities. We group them into four types: *Cognition*, *Perception*, *Reasoning*, and *Composite*, corresponding to scientific knowledge, visual understanding, cross-context reasoning, and multi-capability integration. We report Q-Recall for the 12 evaluated MLLMs in thinking mode in Table 3. The main findings are:

(i) Composite errors are challenging for many MLLMs. 8 models perform below their overall average on Composite errors. One possible reason is that such errors often lack a single clue that can directly expose the problem, with anomalies emerging only after combining multiple pieces of information, making them more hidden and difficult to actively discover.

(ii) Different models show different preferences. The relative performance differences under different capability requirements do not change simultaneously with the overall performance of the model. For example, Gemini-3.1-Pro, Gemma-4-31B-it and Intern-S2-Preview are 5.5%, 3.0% and 4.3% higher than their respective averages on perception errors.

Detection Factor Analysis. We further examine factors associated with detection difficulty. As shown in Figure 5, performance changes only modestly with document length and evidence distance, suggesting that long contexts and distant evidence are not the primary bottlenecks.

In contrast, evidence quantity and reasoning depth show clearer trends. Errors with more supporting evidence are generally easier to verify, while performance declines as the required reasoning steps increase, indicating that multi-step reasoning remains a major challenge. The rebound in the final tier is likely due to its smaller sample size and wider confidence interval.

False Positives Analysis. We randomly sample 30 papers and analyze 57 false positives predictions produced by the two best-performing models, GPT-5.4 and Doubao-Seed-2.0-Pro. Manual verification confirms that 50 are genuine false positives, while only 3 are valid errors missed by the annotations and the remaining 4 are attributable to OCR noise. Representative examples are provided in Appendix C.3. We identify two major sources of false positives.

(i) Factual or reasoning failures. Among these, 11 are hallucinations, involving misreading or incorrectly interpreting evidence from the paper or domain knowledge; another 8 arise from invalid reasoning, where the cited facts are largely correct but do not support the claimed scientific error.

(ii) Misclassifying non-errors as scientific errors. Among the false positives, 2 rely on subjective assumptions or debatable interpretations, 13 describe ordinary study limitations that don't compromise scientific correctness, and 16 involve minor wording issues without substantive impact. Despite explicit prompt instructions to exclude such cases, models still frequently misclassify them as scientific errors, revealing a tendency toward being overly strict.

Missed Error Analysis. We randomly sample 50 papers and evaluate GPT-5.4, Doubao-Seed-2.0-Pro, Qwen-3.6-Plus, and Qwen3.5-27B. For each model, we gather previously missed errors. We then provide the model with each missed error's location and ask it to verify whether it is an error.

Table 4 reports verification accuracy and reasoning quality. Results show that when the locations of missed errors are explicitly provided, models achieve 52.5%–81.3% verification accuracy and maintain high reasoning quality across *Which*, *What*, and *Why*. This finding indicates that many missed errors are not beyond the models' ability to judge correctly, but are instead overlooked during autonomous detection.

Table 4: Verification performance on model-specific errors missed during initial detection.

Model	Accuracy	Reasoning Quality		
		Which	What	Why
GPT-5.4	81.3	84.1	77.7	85.4
Doubao-Seed-2.0-Pro	62.9	86.0	78.0	83.4
Qwen-3.6-Plus	52.5	83.0	67.6	72.2
Qwen3.5-27B	72.3	83.3	73.9	76.5

5 Conclusion

We introduce SciAudit, a large-scale, fine-grained benchmark for detecting errors in scientific papers. Evaluation across 13 MLLMs shows that autonomous end-to-end detection remains limited, even for leading models: thinking improves performance, yet composite errors and multi-step evidence checking remain challenging. Meanwhile, when given candidate issues, models can correctly verify many errors they fail to detect independently, suggesting that these failures are more often due to limitations in error discovery and reasoning than to a lack of knowledge. We hope SciAudit can provide a rigorous testbed for developing more reliable scientific auditing agent systems in the future.

AI Use Statement

In this work, we used generative AI tools to aid and polish the writing of the manuscript and to assist benchmark construction. For the Gold subset, MLLMs were used to mine candidate issues from papers and associated public reviews, while all final annotation decisions were made by human experts. For the Silver subset, multiple MLLMs were used to generate, cross-check, and refine controlled errors, with their quality further assessed through manual validation on a sampled subset.

We reviewed all AI-assisted writing. The authors independently determined the research design, experimental methodology, analysis, and conclusions. We take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

Ethics Statement

The benchmark is constructed from publicly accessible scholarly papers. All error annotations, modified instances, and evaluation labels were created by the authors, and no proprietary or non-public datasets were used. Data collection, processing, and release comply with the ethical and legal requirements applicable at the time of acquisition.

The annotations are benchmark labels defined under our operational criteria and may remain subject to uncertainty or legitimate scientific disagreement. They are intended solely to evaluate MLLMs' ability to detect potential scientific errors and do not constitute formal findings of research misconduct. No inference should be drawn regarding the intent, integrity, competence, or professional conduct of the original authors or their institutions.

The annotations must not be used by any individual or organization to accuse, allege misconduct against, or otherwise make adverse claims about any author, researcher, or institution, nor may they be used for reputational, disciplinary, punitive, or other adverse purposes.

The benchmark, code, and documentation will be publicly released for academic and non-commercial research upon acceptance.

All 13 evaluated models were accessed through legally authorized APIs provided by VolcEngine, Alibaba Cloud, OpenRouter, BigModel, and InternLM.

Reproducibility Statement

To facilitate verification and future extension, we will release the complete dataset, source code, and detailed documentation upon acceptance. The repository will also include step-by-step instructions and the exact hyperparameter configurations used in our experiments, enabling other researchers to reproduce our results and build upon our work.

References

Balazs Aczel, Ann-Sophie Barwich, Amanda B. Diekman, Ayelet Fishbach, Robert L. Goldstone, Pablo Gomez, Odd Erik Gundersen, Paul T. von Hippel, Alex O. Holcombe, Stephan Lewandowsky, Nazbanou Nozari, Franco Pestilli, and John P. A. Ioannidis. The present and future of peer review: Ideas, interventions, and evidence. *Proceedings of the National Academy of Sciences*, 122(5):e2401232121, 2025. doi: 10.1073/pnas.2401232121.

Amazon Web Services. Grok 4.3 – amazon bedrock, June 2026. URL `https://docs.aws.amazon.com/bedrock/latest/userguide/model-card-xai-grok-4-3.html`. Model card.

Anthropic. Introducing Claude Sonnet 4.6, February 2026. URL `https://www.anthropic.com/news/claude-sonnet-4-6`.

Lei Bai, Jiaqi Cao, Chiyu Chen, Guanzhou Chen, Kai Chen, Guangran Cheng, Erfei Cui, Xuanlang Dai, Shengyuan Ding, Shangheng Du, Yanhui Duan, Yue Fan, Youqing Fang, Quan Gan, Yuanyuan Gao, Jiaye Ge, Lixin Gu, Yuzhe Gu, Qipeng Guo, Junjun He, Xin Hong, Ming Hu, Zhouqi Hua,

Haian Huang, Junhao Huang, Zixian Huang, Minxi Jin, Lingkai Kong, Alexander Lam, Zehao Li, Zonglin Li, Tianhao Liang, Dahua Lin, Junyao Lin, Tianyang Lin, Zhouhan Lin, Jiangning Liu, Jin Liu, Kuikun Liu, Wenran Liu, Yifei Liu, Yuhong Liu, Yuhong Liu, Zhoumianze Liu, Ziyan Liu, Ziyu Liu, Haijun Lv, Han Lv, Chengqi Lyu, Le Ma, Ningsheng Ma, Zerun Ma, Haoyang Peng, Runyu Peng, Jifei Shan, Zixin Shang, Kou Shi, Xiang Shi, Qisheng Su, Xuerui Su, Hao Sun, Xiao Sun, Yanan Sun, Yu Sun, Huanze Tang, Yinghao Tang, Wenhui Tian, Zhongbo Tian, Bingli Wang, Haomin Wang, Jiarui Wang, Jingzhi Wang, Rui Wang, Xiquan Wang, Yi Wang, Zhecan Wang, Ziyi Wang, Zun Wang, Rubin Wei, Lianyi Wu, Wen Wu, Yue Wu, Yuhan Wu, Zhenyu Wu, Zijian Wu, Shuhao Xing, Jun Xu, Xingle Xu, Xuenan Xu, Xiangchao Yan, Ziang Yan, Bowen Yang, Danni Yang, Lin Yang, Zhiqi Yang, Qian Yao, Haochen Ye, Peng Ye, Jinhui Yin, Jiashuo Yu, Dingbo Yuan, Fei Yuan, Yuhang Zang, Bo Zhang, Chao Zhang, Chen Zhang, Hongjie Zhang, Junming Zhang, Wenlong Zhang, Wenwei Zhang, Yiming Zhang, Zhuo Zhang, Ziyang Zhang, Haiteng Zhao, Penghao Zhao, Yibo Zhao, Zhonghan Zhao, Zhihang Zhong, Bowen Zhou, Peiheng Zhou, Xin Zhou, Xinyu Zhou, Yunhua Zhou, Dongsheng Zhu, and Yicheng Zou. Intern-s2-preview: Scientific agentic foundation model, 2026. URL https://arxiv.org/abs/2608.13505.

Shuai Bai, Yuxuan Cai, Ruizhe Chen, et al. Qwen3-vl technical report. *arXiv preprint arXiv:2511.21631*, 2025.

ByteDance Seed. Seed2.0 model card: Towards intelligence frontier for real-world complexity, 2026. URL https://arxiv.org/abs/2607.00248.

Yuan Chang, Ziyue Li, Hengyuan Zhang, Yuanbo Kong, Yanru Wu, Hayden Kwok-Hay So, Zhijiang Guo, Liya Zhu, and Ngai Wong. 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*, pp. 15651–15682. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.emnlp-main.790.

Gemma Team et al. Gemma 4 technical report, 2026. URL https://arxiv.org/abs/2607.02770.

Google DeepMind. Gemini 3.1 Pro model card, February 2026. URL https://deepmind.google/models/model-cards/gemini-3-1-pro/.

Mark A. Hanson, Pablo Gómez Barreiro, Paolo Crosetto, and Dan Brockington. The strain on scientific publishing. *Quantitative Science Studies*, 5(4):823–843, 11 2024. ISSN 2641-3337. doi: 10.1162/qss_a_00327. URL https://doi.org/10.1162/qss_a_00327.

Rongjin Li, Zichen Tang, Xianghe Wang, Xinyi Hu, Zhengyu Wang, Zhengyu Lu, Yiling Huang, Jiayuan Chen, Weisheng Tan, Jiacheng Liu, Zhongjun Yang, and Haihong E. Not search, but scan: Benchmarking mllms on scan-oriented academic paper reasoning. In *The Fourteenth International Conference on Learning Representations*, 2026a. URL https://openreview.net/forum?id=GDA1yB6yDP.

Zhuofeng Li, Yi Lu, Dongfu Jiang, Haoxiang Zhang, Yuyang Bai, Chuan Li, Yu Wang, Shuiwang Ji, Jianwen Xie, and Yu Zhang. ReviewGrounder: Improving review substantiveness with rubric-guided, tool-integrated agents. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 32015–32044. Association for Computational Linguistics, 2026b. doi: 10.18653/v1/2026.acl-long.1477.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *Advances in Neural Information Processing Systems*, volume 36, pp. 34892–34916, 2023. doi: 10.52202/075280-1516. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/6dcf277ea32ce3288914faf369fe6de0-Abstract-Conference.html.

Charles J. Lovering, Seth Ebner, Brandon Smock, Michael Krumdick, Saad Rabbani, Ahmed Muhammad, Varshini Reddy, and Chris Tanner. On finding inconsistencies in documents, 2025. URL https://arxiv.org/abs/2512.18601.

Meta AI. The Llama 4 herd: The beginning of a new era of natively multimodal AI innovation, April 2025. URL https://ai.meta.com/blog/llama-4-multimodal-intelligence/.

Mistral AI. Introducing Mistral Small 4, March 2026. URL https://mistral.ai/news/mistral-small-4/.

OpenAI. GPT-5.4 Thinking system card, March 2026. URL https://openai.com/index/gpt-5-4-thinking-system-card/.

Qwen Team. Qwen3.5: Towards native multimodal agents, February 2026a. URL https://qwen.ai/blog?id=qwen3.5.

Qwen Team. Qwen3.6-Plus: Towards real world agents, April 2026b. URL https://qwen.ai/blog?id=qwen3.6.

Lukas Selch, Yufang Hou, Muhammad Jehanzeb Mirza, Sivan Doveh, James R. Glass, Rogério Feris, and Wei Lin. PRISMM-bench: A benchmark of peer-review grounded multimodal inconsistencies. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=mjkGXdgm4T.

Hyungyu Shin, Jingyu Tang, Yoonjoo Lee, Nayoung Kim, Hyunseung Lim, Ji Yong Cho, Hwajung Hong, Moontae Lee, and Juho Kim. Mind the blind spots: A focus-level evaluation framework for LLM reviews. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 35630–35656. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.emnlp-main.1805.

Guijin Son, Jiwoo Hong, Honglu Fan, Heejeong Nam, Hyunwoo Ko, Seungwon Lim, Jinyeop Song, Jinha Choi, Gonçalo Paulo, Youngjae Yu, and Stella Biderman. When AI co-scientists fail: SPOT—a benchmark for automated verification of scientific research. *arXiv preprint arXiv:2505.11855*, 2025. doi: 10.48550/arXiv.2505.11855. URL https://arxiv.org/abs/2505.11855.

V Team, Wenyi Hong, Wenmeng Yu, Xiaotao Gu, Guo Wang, Guobing Gan, Haomiao Tang, Jiale Cheng, Ji Qi, Junhui Ji, Lihang Pan, Shuaiqi Duan, Weihan Wang, Yan Wang, Yean Cheng, Zehai He, Zhe Su, Zhen Yang, Ziyang Pan, Aohan Zeng, Baoxu Wang, Bin Chen, Boyan Shi, Changyu Pang, Chenhui Zhang, Da Yin, Fan Yang, Guoqing Chen, Jiazheng Xu, Jiale Zhu, Jiali Chen, Jing Chen, Jinhao Chen, Jinghao Lin, Jinjiang Wang, Junjie Chen, Leqi Lei, Letian Gong, Leyi Pan, Mingdao Liu, Mingde Xu, Mingzhi Zhang, Qinkai Zheng, Sheng Yang, Shi Zhong, Shiyu Huang, Shuyuan Zhao, Siyan Xue, Shangqin Tu, Shengbiao Meng, Tianshu Zhang, Tianwei Luo, Tianxiang Hao, Tianyu Tong, Wenkai Li, Wei Jia, Xiao Liu, Xiaohan Zhang, Xin Lyu, Xinyue Fan, Xuancheng Huang, Yanling Wang, Yadong Xue, Yanfeng Wang, Yanzi Wang, Yifan An, Yifan Du, Yiming Shi, Yiheng Huang, Yilin Niu, Yuan Wang, Yuanchang Yue, Yuchen Li, Yutao Zhang, Yuting Wang, Yu Wang, Yuxuan Zhang, Zhao Xue, Zhenyu Hou, Zhengxiao Du, Zihan Wang, Peng Zhang, Debing Liu, Bin Xu, Juanzi Li, Minlie Huang, Yuxiao Dong, and Jie Tang. Glm-4.5v and glm-4.1v-thinking: Towards versatile multimodal reasoning with scalable reinforcement learning, 2025. URL https://arxiv.org/abs/2507.01006.

Songjun Tu, Yiwen Ma, Jiahao Lin, Qichao Zhang, Xiangyuan Lan, Junfeng Li, Nan Xu, Linjing Li, and Dongbin Zhao. PaperAudit-Bench: Benchmarking error detection in research papers for critical automated peer review. *arXiv preprint arXiv:2601.19916*, 2026.

Weiyun Wang, Zhangwei Gao, Lixin Gu, Hengjun Pu, Long Cui, Xingguang Wei, Zhaoyang Liu, et al. Internvl3.5: Advancing open-source multimodal models in versatility, reasoning, and efficiency. *arXiv preprint arXiv:2508.18265*, 2025.

Sarina Xi, Vishisht Rao, Justin Payan, and Nihar B. Shah. Flaws: A benchmark for error identification and localization in scientific papers, 2025. URL https://arxiv.org/abs/2511.21843.

Junyu Xiong, Yonghui Wang, Weichao Zhao, Chenyu Liu, Bing Yin, Wengang Zhou, and Houqiang Li. Docr1: Evidence page-guided grpo for multi-page document understanding. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 11178–11186, 2026. doi: 10.1609/aaai.v40i13.38097.

Zhijian Xu, Yilun Zhao, Manasi Patwardhan, Lovekesh Vig, and Arman Cohan. 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)*, pp. 20652–20706. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.acl-long.1009.

Wenwen Yu, Zhibo Yang, Yuliang Liu, and Xiang Bai. Docthinker: Explainable multimodal large language models with rule-based reinforcement learning for document understanding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 837–847, 2025.

Handong Zheng, Yumeng Li, Kaile Zhang, Liang Xin, Guangwei Zhao, Hao Liu, Jiayu Chen, Jie Lou, Qi Fu, Rui Yang, Shuo Jiang, Weijian Luo, Weijie Su, Weijun Zhang, Xingyu Zhu, Yabin Li, Yiwei ma, Yu Chen, Yuqiu Ji, Zhaohui Yu, Guang Yang, Colin Zhang, Lei Zhang, Yuliang Liu, and Xiang Bai. Multimodal ocr: Parse anything from documents, 2026. URL https://arxiv.org/abs/2603.13032.

A System Prompts

Prompt A.1: General Error Types

```
# [Error Types]

Each error must be assigned to exactly one of the following 9 types:

1. **Research Questions & Definitions**

   - **Definition**: Errors where the research object, research
   question, study scope, or core concepts are defined incorrectly or
   described inconsistently within the paper.

2. **Claims & Contributions**

   - **Definition**: Errors where the paper's statements about its
   contributions, novelty, capabilities, applicability, or scope are
   objectively incorrect or inconsistent with what the work actually
   introduces, implements, or evaluates.

3. **Formulae & Calculations**

   - **Definition**: Errors in formulae, calculations, derivations,
   proofs, formal reasoning, or quantitative analysis where equations,
   logical steps, or numerical results are mathematically incorrect,
   or internally inconsistent.

4. **Data & Materials Handling**

   - **Definition**: Errors where data, samples, materials,
   measurements, or statistics are collected, selected, described,
   processed, partitioned, or used incorrectly or inconsistently, such
   as data contamination, leakage, duplication, mislabeling, or
   improper reuse.

5. **Method Design & Specification**

   - **Definition**: Errors where a model, algorithm, analytical
   framework, procedure, measurement, or workflow is incorrectly
   designed, specified, or described inconsistently.

6. **Evaluation & Experimental Validity**

   - **Definition**: Errors where an evaluation, comparison,
   experiment, statistical analysis, case study, qualitative
   assessment, or theoretical validation is designed, conducted, or
   reported incorrectly or inconsistently.

7. **Conclusion & Interpretation**

   - **Definition**: Errors where a conclusion or interpretation is
   objectively incorrect, contradicts the reported evidence or
   results.

8. **Citation Alignment & Reference**

   * **Definition**: Errors in the consistency of citations and
   references that can be verified from information within the paper,
   such as incorrect references to figures or tables, mismatches
   between in-text citations and reference entries, and duplicate or
   internally inconsistent reference entries.

9. **Language & Expression**

   - **Definition**: Errors where a domain-critical term is used
   incorrectly or inconsistently. Minor spelling, grammar,
   punctuation, and wording that don't affect meaning are excluded.
```

Prompt A.2: General Output Format

```
# [Output Format]

Return only a valid JSON object with the key `"errors"`. The following
example shows the required structure of each error item:

{
 "errors": [
  {
    "id": 1,
    "error_text": "These findings suggest that Compound X attenuates
    inflammation in patients with inflammatory bowel disease.",
    "error_type": "Claims & Contributions",
    "error_location": "Page 1, Abstract",
    "error_evidence": [
      {
        "location": "Page 3, Methods: Animal model",
        "quote": "Experiments were performed using male C57BL/6J mice
        subjected to DSS-induced colitis.",
        "supports": "The experiments were conducted in a mouse model
        rather than human patients."
      },
      {
        "location": "Page 6, Figure 2: Experimental design schematic",
        "quote": "Visual evidence: Figure 2 depicts DSS-induced
        colitis in mice, followed by Compound X administration and
        colon-tissue collection.",
        "supports": "The experimental workflow is entirely preclinical
        and contains no human-patient intervention."
      },
      {
        "location": "Page 9, Limitations",
        "quote": "Clinical validation in patients with inflammatory
        bowel disease remains necessary.",
        "supports": "The paper acknowledges that patient-level
        efficacy has not been clinically validated."
      },
      {
        "location": "General domain knowledge: preclinical-to-clinical
        generalization",
        "quote": null,
        "supports": "Animal-model evidence alone does not establish
        therapeutic efficacy in human patients."
      }
    ],
    "error_reasoning": [
        "<claim>The paper claims that Compound X attenuates
        inflammation in inflammatory bowel disease patients.</claim>",
        "<evidence>The experiments evaluated Compound X only in mice,
        and the paper states that clinical validation remains
        necessary.</evidence>",
        "<inference>(1) The reported evidence establishes efficacy
        only in the tested mouse model. (2) Preclinical animal results
        alone do not support a claim of efficacy in human
        patients.</inference>",
        "<judgment>The paper overstates the clinical applicability of
        its findings.</judgment>"
    ],
    "error_summary": "The paper generalizes preclinical mouse-model
    evidence to human patients without clinical validation."
  }
 ]
}

## [Field Requirements]

{{ GENERAL_FIELD_REQUIREMENTS }}
```

Prompt A.3: General Field Requirements

```
## [Field Requirements]

1. `id`: Must start from 1, increase sequentially, and be unique.

2. `error_text`: Must be the erroneous passage in the paper.

3. `error_type`: Must be one, and only one, of the 9 error categories
above. Select the `error_type` according to the primary point in the
research workflow where the error compromises the paper's validity.

4. `error_location`: The page number where the error appears, along
with the nearest section or subsection title.

5. `error_evidence`: Must be a JSON array containing at least one
evidence item needed to verify the error. Evidence should be concise,
non-redundant, sufficient for verification, and each item should
contribute distinct support for the error judgment. Each item must
include:

   - `location`: (1) If the evidence comes from the paper, provide the
   page number plus the nearest section, subsection, figure/table
   caption, formula, or reference entry. (2) If the evidence comes
   from general domain knowledge, specify the relevant principle,
   concept, rule, or fact beginning with `General domain knowledge:`.

   - `quote`: (1) For text evidence, provide a brief quote copied from
   the paper. (2) For visual evidence, provide a concise description
   beginning with `Visual evidence:`. (3) For general domain knowledge,
   set `quote` to `null`.

  - `supports`: A concise summary of the specific evidence, principle,
  rule, or fact established by that location.

6. 'error_reasoning': Must be a JSON array containing exactly four
ordered strings, progressing from claim -> evidence -> inference ->
judgment:

  - `<claim>...</claim>`: State the erroneous claim or issue.

  - `<evidence>...</evidence>`: State what the evidence establishes.

  - `<inference>...</inference>`: Explain how the evidence leads to the
  judgment. Use `(1) (2) ...` for multiple inference points when
  needed.

  - `<judgment>...</judgment>`: State the final error judgment.

7. `error_summary`: Concisely summarize the root cause of the error.
```

A.1 Detection

Prompt A.4: Error Detection

```
# [Task]

You are an expert academic reviewer tasked with identifying
objectively verifiable scientific errors in an academic paper.

Identify errors that can be established from evidence in the paper,
necessary contextual reasoning, or widely accepted domain knowledge. A
valid error must render a claim, method, result, or conclusion
factually incorrect or internally inconsistent.

Do not report concerns based on subjective assumptions, contestable
interpretations, or insufficient evidence. Subjective or
insufficiently supported concerns, study limitations, and minor
spelling, grammar, or wording issues that don't affect meaning are
excluded.

# [Detection Requirements]

1. Identify all objectively sufficiently-supported scientific errors.

2. Do not treat normal variation in academic expression as an error.

3. Support every error with a clear and verifiable chain of evidence.
Do not introduce assumptions not justified by the paper or widely
accepted domain knowledge.

4. Use only evidence from the paper and necessary, widely accepted
domain knowledge.

5. Avoid redundant reporting. Merge repeated or closely related
manifestations of the same underlying error into one item. For
example, multiple nonexistent references or multiple consequences of
the same incorrect definition may be consolidated.

6. If no sufficiently supported scientific errors are found, output
`{"errors":[]}`.

# [Error Types]

{{ GENERAL_ERROR_TYPES }}

# [Output Format]

{{ GENERAL_OUTPUT_FORMAT }}
```

A.2 GOLD

Prompt A.5: Error Mining

```
# [Task]

You will receive an **academic paper** and an associated **Review
PDF** containing public reviews of the paper.

Use the Review PDF as guidance and independently conduct a full-paper
review to identify **all scientific errors**. Consider both
review-raised issues and additional errors identified from the paper.
Report only errors that can be established from the paper, necessary
contextual reasoning, or widely accepted domain knowledge and that
make a claim, method, result, or conclusion factually incorrect or
internally inconsistent.

The Review PDF is **not ground truth and must not serve as the sole
evidence of an error**. Every reported error must be independently
supported by the paper, necessary contextual reasoning, or widely
accepted domain knowledge.

# [Detection Requirements]

1. Use the Review PDF as a **source of candidate errors**, but
independently verify every reported error.

2. Report only **objectively verifiable and sufficiently supported
scientific errors**. Exclude subjective comments, reviewer
preferences, speculative concerns, general limitations, and requests
for additional experiments or analyses.

3. Conduct a **full-paper open-ended review** and report supported
errors even if they are not mentioned in the Review PDF.

4. Support every error with a clear and verifiable chain of evidence
that remains valid without the Review PDF. Do not introduce
assumptions not justified by the paper or widely accepted domain
knowledge.

5. Avoid redundant reporting. Merge repeated or closely related
manifestations of the same underlying error into one item. For
example, multiple nonexistent references or multiple consequences of
the same incorrect definition may be consolidated.

6. Do not treat public criticism, rejection, correction, or
retraction-related discussion itself as proof of an error.

7. If no sufficiently supported errors are found, output
`{"errors":[]}`.

# [Error Types]

{{ GENERAL_ERROR_TYPES }}

# [Output Format]

{{ GENERAL_OUTPUT_FORMAT }}
```

A.3 Silver

A.3.1 Draft: Error Generation

Prompt A.6: Error Generation

```
# [Task]

You are an expert in generating errors for academic papers.
You will receive a high-quality, accepted scientific paper. Your task
is to precisely insert synthetic errors into specific text passages
based on the paper. These errors must be carefully designed so that
human experts can clearly determine them to be errors by only reading
the paper and using logical reasoning and cross-verification among the
text, formulas, figures, figure captions, table notes, citation and
reference information, and statements across sections, without relying
on subjective speculation or conjecture about the authors' intentions.
In your final response, output only a valid JSON object.

# [Error Types]

{{ GENERAL_ERROR_TYPES }}

# [Generation Requirements]

1. Design challenging but natural and unambiguous errors. Prefer
errors whose diagnosis requires cross-context or cross-modal
reasoning, using multiple evidence items across sections, tables,
figures, formulas, or references.
2. Each error must be objectively verifiable from the paper itself, or
necessary domain knowledge. Avoid subjective interpretation or
author-intent speculation.
3. The generated errors must not be immediately obvious or appear
artificially planted. Avoid awkward wording, overstatement,
implausible values, abrupt reversals, obvious self-contradictions,
absolute claims, dramatic numerical shifts, or trap-like phrasing.
4. Diversify error mechanisms within each error category. Each
category may involve multiple distinct ways for an error to arise;
avoid repeatedly relying on only one obvious mechanism under the same
category.
5. Prioritize important sections, but distribute errors across
multiple major sections whenever possible. Avoid concentrating errors
in a single section or subsection.
6. You may edit local content in the main text, figure captions, table
notes, existing formulas, and existing reference entries. Do not edit
the paper title, author list, image body, or table body. Do not add or
delete figures, tables, formulas, or reference entries.

# [Independence Constraints]

1. **Objective Validity**: A valid error must be provably wrong from
the paper itself, or necessary domain knowledge.
2. **Independence**: Each error must stand on its own. The validity of
one error must not depend on another generated error.
3. **Measured Expression**: The error type defines the logical nature
of the issue, not the intensity of the wording. Even errors involving
exaggeration, contradiction, overgeneralization, or inverted
interpretation must remain academically plausible and consistent with
the tone of the original paper.
4. **Figure/Table Constraint**: If an error relies on a conflict with
a figure, table, caption, or table note, only modify the surrounding
descriptive text, figure captions, or table notes. Do not modify the
image body or table body.

# [Output Format]

{{ ERROR_GENERATION_OUTPUT_FORMAT }}
```

Prompt A.7: Error Generation Output Format

```
# [Output Format]

Return only a valid JSON object with the key `"errors"`. Generate up to
15 errors, prioritizing quality over quantity. The following example
shows the required structure of each error item:

{
 "errors": [
  {
    "id": 1,

    "original_text": "These findings suggest that Compound X
    attenuates inflammation in the mouse model under study.",

    "error_text": "These findings suggest that Compound X attenuates
    inflammation in patients with inflammatory bowel disease.",

    "error_type": "Claims & Contributions",

    "error_location": "Page 1, Abstract",

    "error_summary": "The paper generalizes preclinical mouse-model
    evidence to human patients without clinical validation."
  }
 ]
}

## [Field Requirements]

1. `id`: Must start from 1, increase sequentially, and be unique.

2. `original_text`: Must be an exact, contiguous substring copied
verbatim from the original paper. Do not paraphrase, shorten,
summarize, or omit any part of the text. Select a complete and as
short as possible sentence as the location segment, and avoid spanning
multiple sentences unless necessary.

3. `error_text`: Make a natural modification based on `original_text`,
ensuring that the replacement fits naturally into the paper context.
The replacement must introduce an objectively verifiable error, not
correct, improve, normalize, clarify, or reinterpret the original
text. It should avoid exaggerated, implausible, abruptly reversed, or
obviously artificial wording.

4. `error_type`: Must be one, and only one, of the 9 error types above.

5. `error_location`: The page number where the error appears, along
with the nearest section or subsection title.

6. `error_summary`: Summarize the error cause, assuming the
`error_text` is already part of the paper.
```

A.3.2 Draft: Error Annotation

Prompt A.8: Error Annotation

```
# [Task]

You are an expert academic error annotator.

You will receive:
1. An error JSON including the fields `id`, `error_type`, `error_text`,
`error_location` and `error_summary`.
2. An academic paper that contains the errors recorded in the above
JSON.

Your task is to annotate each error listed in the error JSON based on
the flawed paper. For each error, you must annotate it with verifiable
evidence and structured reasoning.

In your final response, output only a valid JSON object.

# [Annotation Principles]

1. Use only evidence from the erroneous paper state, necessary widely
accepted domain knowledge.

2. Do not rely on subjective interpretation, author-intent
speculation, or loose plausibility judgments.

3. Evidence should be concise, non-redundant, directly useful for
verifying the error, and sufficiently comprehensive. Each evidence
item should make a distinct contribution to establishing the error.

4. Reasoning should explicitly connect the erroneous claim, the
evidence, the inference process, and the final judgment.

# [Error Types]

{{ GENERAL_ERROR_TYPES }}

# [Output Format]

{{ ERROR_ANNOTATION_OUTPUT_FORMAT }}
```

Prompt A.9: Error Annotation Output Format

```
# [Output Format]

Return only a valid JSON object with the key `"errors"`. Each item must
follow this structure:

{
 "errors": [
  {
    "id": 1,
    "error_evidence": [
      {
        "location": "Page 3, Methods: Animal model",
        "quote": "Experiments were performed using male C57BL/6J mice
        subjected to DSS-induced colitis.",
        "supports": "The experiments were conducted in a mouse model
        rather than human patients."
      },
      {
        "location": "Page 6, Figure 2: Experimental design schematic",
        "quote": "Visual evidence: Figure 2 depicts DSS-induced
        colitis in mice, followed by Compound X administration and
        colon-tissue collection.",
        "supports": "The experimental workflow is entirely preclinical
        and contains no human-patient intervention."
      },
      {
        "location": "Page 9, Limitations",
        "quote": "Clinical validation in patients with inflammatory
        bowel disease remains necessary.",
        "supports": "The paper acknowledges that patient-level
        efficacy has not been clinically validated."
      },
      {
        "location": "General domain knowledge: preclinical-to-clinical
        generalization",
        "quote": null,
        "supports": "Animal-model evidence alone does not establish
        therapeutic efficacy in human patients."
      }
    ],
    "error_reasoning": [
        "<claim>The paper claims that Compound X attenuates
        inflammation in inflammatory bowel disease patients.</claim>",
        "<evidence>The experiments evaluated Compound X only in mice,
        and the paper states that clinical validation remains
        necessary.</evidence>",
        "<inference>(1) The reported evidence establishes efficacy
        only in the tested mouse model. (2) Preclinical animal results
        alone do not support a claim of efficacy in human
        patients.</inference>",
        "<judgment>The paper overstates the clinical applicability of
        its findings.</judgment>"
    ]
  }
 ]
}

## [Field Requirements]

{{ SELECT_GENERAL_FIELD_REQUIREMENTS }}
```

A.4 CHECK

Prompt A.10: Check

```
# [Task]

You are a strict expert in checking artificially injected errors in
academic papers.

You will receive:
1. A candidate JSON containing artificially injected errors;
2. An academic paper that is expected to contain the errors recorded
in the above JSON.

The candidate JSON uses `error_text` to denote the erroneous passage in
the paper. Do not assume a candidate is valid simply because it is
provided. Independently verify whether each candidate constitutes an
objectively verifiable scientific error and whether its `error_text`
and annotations are clear, accurate, verifiable, and consistent with
the paper.

In your final response, output only a valid JSON object.

# [Check Order]

Check each candidate error in three progressive stages:

## Stage 1. Error Validity

Decide whether the candidate error itself is valid or valuable from
the following aspects:

1. **Existence and validity**: The `error_text` must appear in the
paper and make the passage objectively wrong.

2. **Verifiability**: The error must be provable from the paper
itself, or necessary domain knowledge, not from speculation,
preference, convention, author intent.

3. **Independence**: The error and its evidence must not depend on any
other injected error. It must not duplicate another candidate unless
it has a distinct error mechanism.

4. **Benchmark value**: The error should have meaningful benchmark
value within its own category and paper context. `Discard` valid but
low-value candidates if they are trivial, overly obvious, artificial,
repetitive, or weakly impactful. Judge value within the same category
to avoid unfairly penalizing inherently lighter but necessary
categories, such as `Language & Expression`.

Mark the candidate as `discard` if the core error is invalid,
unverifiable, dependent on another injected error, or too low-value to
keep.

If the candidate is not marked as `discard`, proceed to Stage 2.

## Stage 2. Annotation Soundness

For candidates that pass Stage 1, decide whether the current
annotation correctly explains and supports the error.

Check all fields of each candidate error. Mark the candidate as `keep`
only if it can be accepted into the final dataset without substantive
revision. If the error is valid and valuable but any annotation field
is inaccurate, incomplete, inconsistent, or misaligned, mark it as
`revise`:

{{ GENERAL_FIELD_REQUIREMENTS }}
```

```
## Stage 3. Refinement Guidance

For candidates checked in Stage 1 and Stage 2, summarize what should
be kept, revised, or discarded, and how to revise. Do not rewrite the
candidate JSON. Only provide feedback and revision advice for the
later refinement process.

Return only a valid JSON object with the key `"check_results"`. The
following example shows the required structure of each check result
item:

{
  "check_results": [
   {
     "candidate_error_id": 1,
     "decision": "keep",
     "reason": "The candidate identifies an objectively verifiable
     error, with current evidence directly supporting this judgment.",
     "revision_guidance": null
   },
   {
     "candidate_error_id": 2,
     "decision": "revise",
     "reason": "The core error is valid, but the current annotation
     cites the wrong evidence. The candidate claims that Method A
     performs best, while the paper's result table shows that Method B
     has the highest score. However, the current evidence does not
     quote or describe that result table.",
     "revision_guidance": "(1) Replace the current evidence with the
     result table showing the scores of Method A and Method B. (2)
     Explain that Method B, not Method A, has the highest score. (3)
     Revise the reasoning so that it clearly connects the reported
     claim, the table evidence, and the final judgment."
   },
   {
     "candidate_error_id": 3,
     "decision": "discard",
     "reason": "The candidate does not identify an objectively
     verifiable error; the issue depends on interpretation rather than
     paper evidence.",
     "revision_guidance": null
   }
  ]
}

**Field meanings**:

- `candidate_error_id`: The `id` of the candidate error being checked.

- `decision`: Use `keep`, `revise`, or `discard`.

- `reason`: Briefly explain the main reason for the decision.

- `revision_guidance`:

  - If `decision` is `revise`, provide concrete advice on what should
  be fixed, added, removed, merged, or verified during the later
  refinement process. Use (1) (2) ... for multiple actions when
  needed.

  - If `decision` is `keep` or `discard`, set `revision_guidance` to
  `null`.

# [Error Types]

{{ GENERAL_ERROR_TYPES }}
```

A.5 REFINE

Prompt A.11: Refine

```
# [Task]

You are an expert in refining artificially injected errors in academic
papers.

You will receive:
1. a candidate JSON containing artificially injected errors;
2. a checker JSON containing `keep`, `revise`, or `discard` decisions
and revision guidance;
3. the injected paper that already contains the listed errors.

Your task is to produce a cleaned final JSON by using the checker
feedback as a reference and improving the candidate JSON. The
candidate JSON uses `error_text` for the erroneous passage now present
in the paper. Checker feedback is useful, but it is not automatically
correct. When checker feedback conflicts with the injected paper or
objective evidence, follow the injected paper and your own
verification.

In your final response, output only a valid JSON object.

# [Core Principles]

- Do not create new error items.

- Do not keep invalid errors merely to preserve quantity.

- Do not blindly follow checker feedback.

- Each kept error must be objectively verifiable from the injected
paper itself or from necessary domain knowledge.

- Preserve each kept candidate's original `id` exactly as provided in
the Candidate JSON. Do not renumber items after discarding. Do not
mention checker feedback in the output.

# [Error Types]

{{ GENERAL_ERROR_TYPES }}

# [Output Format]

{{ GENERAL_OUTPUT_FORMAT }}
```

A.6 Evaluation

A.6.1 Error Match

Prompt A.12: Error Match

```
# [Task]

You are a rigorous evaluator.

You will receive two JSON objects, named **Ground Truth** and **Model
Prediction**. The **Ground Truth** records the errors annotated by
experts in a paper, while the **Model Prediction** records the errors
found by a model in the same paper.

Your task is to compare the **Ground Truth** with the **Model
Prediction**, verify each error in the **Ground Truth** one by one,
determine whether it matches the **Model Prediction**.

# [Matching Rules]

1. Do not match by ID. IDs are only references.

2. A match is valid only if the Ground Truth and Model Prediction
refer to the same erroneous statement and the same underlying error
logic.

3. Same page, same section, same topic, or similar wording is not
enough for a match.

4. Minor differences in wording, error type, page number, or section
number are acceptable if the actual error is the same.

5. Matching must be one-to-one. Each `matched_prediction_id` can be
used at most once.

6. If one prediction could match multiple Ground Truth items, assign
it only to the best match and set the others to `null`.

7. Before returning the final JSON, check that no non-null
`matched_prediction_id` is repeated.


# [Output Format]

Returns only a valid JSON object. The format is as follows:

{
  "matches": [
    {
      "gt_id": 1,
      "matched_prediction_id": 3
    },
    {
      "gt_id": 2,
      "matched_prediction_id": null
    }
  ]
}

# [Field Requirements]

- `gt_id`: The ID of the `Ground Truth` item.

- `matched_prediction_id`: The ID of the matched Model Prediction item;
use `null` if unmatched.
```

A.6.2 WHAT

Prompt A.13: What

```
# [Task]

You are a rigorous evaluator.

You will receive two JSON objects, named **Ground Truth Evidence** and
**Model Prediction Evidence**. They come from one already matched
error item.

Your task is to compare the evidence items in **Ground Truth
Evidence** with the evidence items in **Model Prediction Evidence**,
and determine whether each Ground Truth evidence item is matched by at
least one Model Prediction evidence item.

# [Matching Rules]

1. Do not match by ID. IDs are for reference only.

2. A valid evidence match requires both conditions:

  - The two evidence items refer to the same or substantively
  equivalent evidence location/source.
  - The two evidence items support the same verification point for the
  matched error.

3. Minor differences in wording, quote length, page number, or section
number are acceptable and do not invalidate a match.

4. Same location or similar wording alone is not enough. If two
evidence items support different points, do not match them.

5. Many-to-many matching is allowed. One Ground Truth evidence item
may be covered by multiple Model Prediction evidence items, and one
Model Prediction evidence item may cover multiple Ground Truth
evidence items. For example, the Ground Truth may split several
verification points into separate evidence items, while the prediction
may merge them into one item, or vice versa.

# [Output Format]

Returns only a valid JSON object. The format is as follows:

{
  "matches": [
    {
      "gt_evidence_id": 1,
      "matched_prediction_evidence_ids": [1, 2, 3]
    },
    {
      "gt_evidence_id": 2,
      "matched_prediction_evidence_ids": []
    }
  ]
}

# [Field Requirements]

- `gt_evidence_id`: The ID of the Ground Truth Evidence item.

- `matched_prediction_evidence_ids`: An array of IDs of the matched
Model Prediction Evidence items. Use an empty array `[]` if the Ground
Truth evidence item is unmatched.
```

A.6.3 WHY

Prompt A.14: Why

```
# [Task]

You are a rigorous evaluator.

You will receive two JSON objects, named **Ground Truth** and **Model
Prediction**. They are reasoning explanations for the same
already-matched error item.

Your task is to extract the core reasoning chain from each side and
use shared tags to mark semantically equivalent reasoning steps.

# [Matching Rules]

1. Each reasoning step should express one core inference unit and
preserve the original logical order.

2. Use the same tag for Ground Truth and Model Prediction steps that
express the same core inference.

3. Use different tags for steps with different meanings, even if they
are related.

4. Do not invent unsupported reasoning steps.

# [Output Format]

Returns only a valid JSON object. The format is as follows:

{
  "gt_chain": [
    "<A>The conclusion claims that neither GRPO- nor SFT-trained
    models generalized to a related task.</A>",
    "<B>The paper reports limited Qwen3-VL generalization between
    related top-block tasks.</B>",
    "<C>The conclusion uses an absolute negative formulation despite
    evidence of limited generalization.</C>",
    "<D>Therefore, the conclusion overstates the negative result.</D>"
  ],
  "prediction_chain": [
    "<A>The prediction identifies that the conclusion claims no model
    generalization.</A>",
    "<B>The prediction cites reported Qwen3-VL generalization between
    related tasks.</B>",
    "<D>The prediction concludes that the no-generalization claim is
    overstated.</D>"
  ]
}

# [Field Requirements]

- `gt_chain`: The ordered core reasoning chain extracted from Ground
Truth.
- `prediction_chain`: The ordered and concise core reasoning chain
extracted from Model Prediction.
- Each reasoning step must be wrapped by a tag, such as `<A>...</A>`,
`<B>...</B>`.
- The same tag indicates semantic equivalence. Different tags indicate
different reasoning steps.
```

B Data Construction

B.1 Review Details

Gold Review Protocol.

GPT-5.4, Gemini-3.1-Pro, and Qwen-3.6-Plus first extract and structure candidate errors from the papers and their associated public reviews. After merging and deduplication, four experts with peer-review experience conduct the formal review. Experts may use AI tools to assist with paper reading, evidence retrieval, and cross-section analysis, but all decisions regarding error validity, annotation revision, and final inclusion are independently made by the human experts based on the paper. The review follows four main criteria:

- **Validity screening.** Experts determine whether each candidate constitutes a substantive and objectively verifiable scientific error. Subjective concerns, general limitations, unsupported speculation, and minor language issues are discarded.
- **Evidence and reasoning review.** Experts verify that the reported evidence is traceable to the paper and that the reasoning logically supports the final error judgment. Inaccurate, incomplete, or overstated annotations are revised.
- **Localization and classification review.** Experts confirm the error location and its assignment to one of the nine predefined categories, correcting the annotation when necessary.
- **Coverage and adjudication.** Each paper is independently reviewed by one expert, who may also add valid errors missed during candidate extraction. When uncertainty arises, the reviewer consults other experts, and the case is discussed until a final decision is reached.

Starting from 3,482 candidates, the experts discard 939, revise 1,154, and add 395 previously missed errors. The resulting Gold subset contains 2,938 human-verified issues across 293 papers.

Gold Quality Evaluation.

To further assess the quality of the final Gold annotations, we randomly sampled approximately 10% of the 293 Gold papers and audited all 277 final errors from these 30 papers. The sampled annotations were independently evaluated by two experts who were not responsible for their original review. The evaluators assessed only the final annotations and were not given the original candidate errors or intermediate review decisions.

Each error was independently scored along three dimensions:

- **Validity**: Is the error objectively real and reliably decidable from the paper, its context, or necessary widely accepted domain knowledge?
- **Value**: Is this a useful benchmark error with a substantive effect on scientific correctness, reproducibility, interpretation, or research credibility?
- **Correctness**: Are the error type, location, affected object, evidence, and explanation consistent with the alleged error?

Each dimension was scored independently on a 0–4 scale: 0 denotes unsupported or incorrect, 1 weak, 2 partially supported or ambiguous, 3 mostly supported, and 4 clear and fully supported.

Table 5: Manual evaluation of 277 Gold errors, with scores ranging from 0 to 4. Results for Mean Score, Score ≥ 3, and Score $= 4$ are reported as Evaluator A / Evaluator B. Inter-evaluator agreement is shown as Exact (identical scores) / ± 1 (score difference within one point).

Dimension	Mean Score	Score ≥ 3	Score $= 4$	Agreement (Exact / ± 1)
Validity	3.78 / 3.83	94.9% / 97.5%	83.4% / 85.2%	80.9% / 96.8%
Value	3.12 / 3.11	91.0% / 89.9%	21.3% / 21.3%	66.8% / 98.9%
Correctness	3.80 / 3.89	97.5% / 98.6%	82.7% / 90.6%	78.7% / 96.4%
Average	3.57 / 3.61	94.5% / 95.3%	62.5% / 65.7%	75.5% / 97.4%

As shown in Table 5, the Gold annotations exhibit consistently high quality. Across the two evaluators, 94.9–97.5% of the errors received a validity score of at least 3, while 97.5–98.6%

received a correctness score of at least 3. Value scores were also high, with 89.9–91.0% of the errors scoring at least 3. Inter-evaluator agreement was strong: 75.5% of scores matched exactly on average, and 97.4% differed by at most one point. These results indicate that the Gold subset consists predominantly of objectively verifiable, scientifically meaningful, and accurately documented errors, with strong consistency between independent evaluators.

Silver Quality Evaluation.

To assess the quality of the Silver set, we randomly sampled 50 of the 527 papers and audited all 541 generated errors from these papers. Each error was independently evaluated by two evaluators using the same three dimensions and 0–4 scoring scheme as in the Gold evaluation.

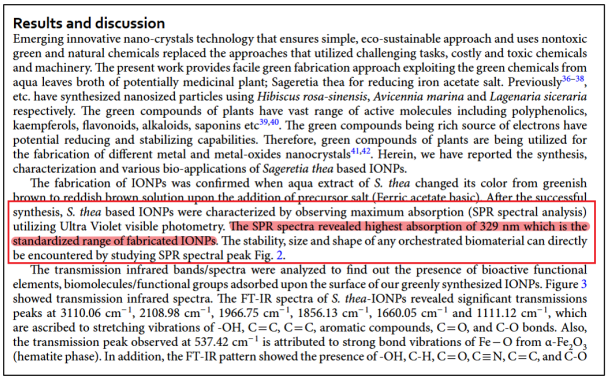
Table 6: Manual evaluation of 541 Silver-generated errors, with scores ranging from 0 to 4. Results for Mean Score, Score ≥ 3, and Score $= 4$ are reported as Evaluator A / Evaluator B. Inter-evaluator agreement is shown as Exact (identical scores) / ± 1 (score difference within one point).

Dimension	Mean Score	Score ≥ 3	Score $= 4$	Agreement (Exact / ± 1)
Validity	3.77 / 3.75	94.1% / 93.3%	84.1% / 84.7%	82.4% / 94.6%
Value	2.92 / 3.03	74.3% / 80.0%	21.6% / 27.0%	60.4% / 96.7%
Correctness	3.69 / 3.68	94.1% / 90.0%	77.3% / 80.2%	74.3% / 90.9%
Average	3.46 / 3.48	87.5% / 87.8%	61.0% / 64.0%	72.4% / 94.1%

As shown in Table 6, the Silver errors are of high overall quality. Across the two evaluators, 93.3–94.1% received a validity score of at least 3, showing that the large majority represent objectively verifiable scientific errors or internal inconsistencies rather than speculative concerns or ordinary study limitations. Annotation correctness was similarly strong, with 90.0–94.1% scoring at least 3, indicating that the assigned error types, locations, evidence, and reasoning were generally consistent with the source papers. In addition, 74.3–80.0% received a value score of at least 3, demonstrating that most errors capture scientifically meaningful issues. Inter-evaluator agreement was also high: 72.4% of scores matched exactly on average, and 94.1% differed by at most one point. Taken together, these results indicate that the Silver set predominantly consists of verifiable, well-documented, and scientifically relevant errors grounded in explicit evidence from the source papers, with strong consistency between independent evaluators.

B.2 Examples of Gold Annotation

B.2.1 Case 1: Discard



Results and discussion

Emerging innovative nano-crystals technology that ensures simple, eco-sustainable approach and uses nontoxic green and natural chemicals replaced the approaches that utilized challenging tasks, costly and toxic chemicals and machinery. The present work provides facile green fabrication approach exploiting the green chemicals from aqua leaves broth of potentially medicinal plant; Sageretia thea for reducing iron acetate salt. Previously[36–38], etc. have synthesized nanosized particles using *Hibiscus rosa-sinensis*, *Avicennia marina* and *Lagenaria siceraria* respectively. The green compounds of plants have vast range of active molecules including polyphenolics, kaempferols, flavonoids, alkaloids, saponins etc[39,40]. The green compounds being rich source of electrons have potential reducing and stabilizing capabilities. Therefore, green compounds of plants are being utilized for the fabrication of different metal and metal-oxides nanocrystals[41,42]. Herein, we have reported the synthesis, characterization and various bio-applications of *Sageretia thea* based IONPs.

The fabrication of IONPs was confirmed when aqua extract of *S. thea* changed its color from greenish brown to reddish brown solution upon the addition of precursor salt (Ferric acetate basic). After the successful synthesis, *S. thea* based IONPs were characterized by observing maximum absorption (SPR spectral analysis) utilizing Ultra Violet visible photometry. The SPR spectra revealed highest absorption of 329 nm which is the standardized range of fabricated IONPs. The stability, size and shape of any orchestrated biomaterial can directly be encountered by studying SPR spectral peak Fig. 2.

The transmission infrared bands/spectra were analyzed to find out the presence of bioactive functional elements, biomolecules/functional groups adsorbed upon the surface of our greenly synthesized IONPs. Figure 3 showed transmission infrared spectra. The FT-IR spectra of *S. thea*-IONPs revealed significant transmissions peaks at 3110.06 cm^{-1}, 2108.98 cm^{-1}, 1966.75 cm^{-1}, 1856.13 cm^{-1}, 1660.05 cm^{-1} and 1111.12 cm^{-1}, which are ascribed to stretching vibrations of -OH, C=C, C=C, aromatic compounds, C=O, and C-O bonds. Also, the transmission peak observed at 537.42 cm^{-1} is attributed to strong bond vibrations of Fe−O from α-Fe_2O_3 (hematite phase). In addition, the FT-IR pattern showed the presence of -OH, C-H, C=O, C≡N, C=C, and C-O

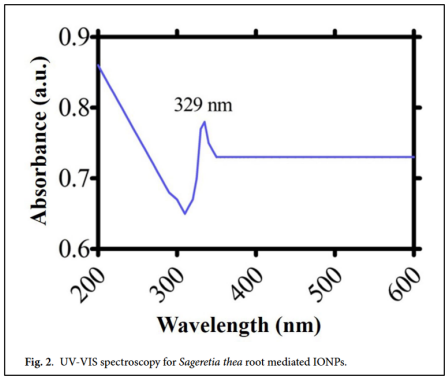


Fig. 2. UV-VIS spectroscopy for *Sageretia thea* root mediated IONPs.

Original annotation: ***Conclusion & Interpretation***

<claim>The results text interprets the UV-Vis spectrum as having its highest absorption at 329 nm.</claim>

<evidence>The scan range is 200–600 nm, and Fig. 2 shows a higher absorbance near 200 nm than at the labeled 329 nm feature.</evidence>

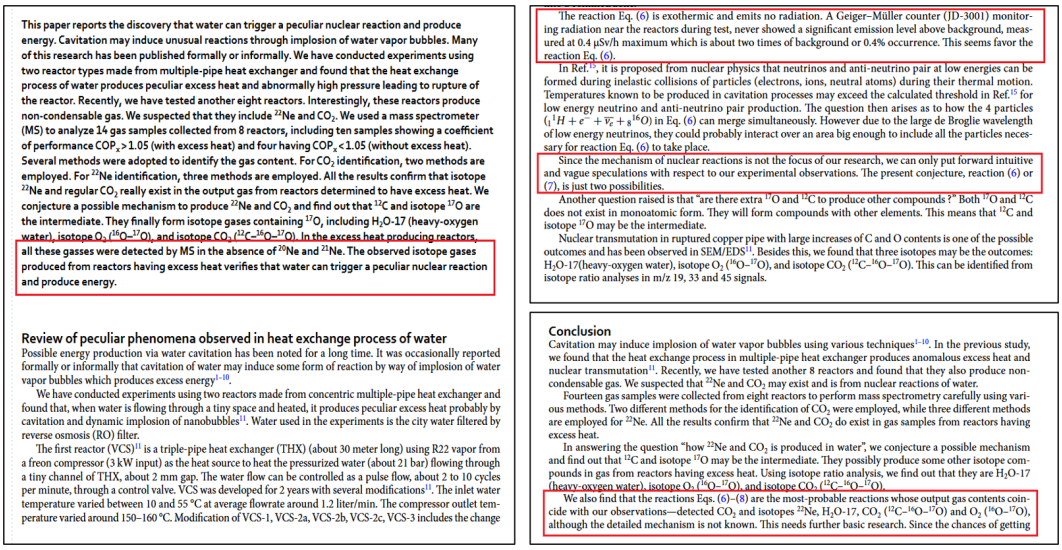
<inference>(1) The highest absorption over the reported spectrum must be the highest point on the plotted trace. (2) Because the low-wavelength end is visibly higher than the 329 nm feature, 329 nm is not the highest absorption in the spectrum.</inference>

<judgment>The paper incorrectly interprets Fig. 2 by calling 329 nm the highest absorption peak.</judgment>

Reason:

This is primarily a wording issue rather than a solid scientific error. Fig. 2 shows a distinct local absorption peak at 329 nm, while the larger absorbance near 200 nm occurs at the boundary of the scan and is not presented as a separate spectral peak. Thus, the phrase "highest absorption" is imprecise if interpreted as the largest absorbance value over the entire 200–600 nm range, but the underlying identification of 329 nm as the characteristic absorption peak remains consistent with the figure. The issue therefore concerns ambiguous wording rather than a demonstrably incorrect scientific interpretation.

B.2.2 Case 2: Revise



This paper reports the discovery that water can trigger a peculiar nuclear reaction and produce energy. Cavitation may induce unusual reactions through implosion of water vapor bubbles. Many of this research has been published formally or informally. We have conducted experiments using two reactor types made from multiple-pipe heat exchanger and found that the heat exchange process of water produces peculiar excess heat and abnormally high pressure leading to rupture of the reactor. Recently, we have tested another eight reactors. Interestingly, these reactors produce non-condensable gas. We suspected that they include ^{22}Ne and CO_2. We used a mass spectrometer (MS) to analyze 14 gas samples collected from 8 reactors, including ten samples showing a coefficient of performance $COP_x > 1.05$ (with excess heat) and four having $COP_x < 1.05$ (without excess heat). Several methods were adopted to identify the gas content. For CO_2 identification, two methods are employed. For ^{22}Ne identification, three methods are employed. All the results confirm that isotope ^{22}Ne and regular CO_2 really exist in the output gas from reactors determined to have excess heat. We conjecture a possible mechanism to produce ^{22}Ne and CO_2 and find out that ^{12}C and isotope ^{17}O are the intermediate. They finally form isotope gases containing ^{17}O, including H_2O-17 (heavy-oxygen water), isotope O_2 (^{16}O–^{17}O), and isotope CO_2 (^{12}C–^{16}O–^{17}O). In the excess heat producing reactors, all these gasses were detected by MS in the absence of ^{20}Ne and ^{21}Ne. The observed isotope gases produced from reactors having excess heat verifies that water can trigger a peculiar nuclear reaction and produce energy.

Review of peculiar phenomena observed in heat exchange process of water

Possible energy production via water cavitation has been noted for a long time. It was occasionally reported formally or informally that cavitation of water may induce some form of reaction by way of implosion of water vapor bubbles which produces excess energy[1–10].

We have conducted experiments using two reactors made from concentric multiple-pipe heat exchanger and found that, when water is flowing through a tiny space and heated, it produces peculiar excess heat probably by cavitation and dynamic implosion of nanobubbles[11]. Water used in the experiments is the city water filtered by reverse osmosis (RO) filter.

The first reactor (VCS)[11] is a triple-pipe heat exchanger (THX) (about 30 meter long) using R22 vapor from a freon compressor (3 kW input) as the heat source to heat the pressurized water (about 21 bar) flowing through a tiny channel of THX, about 2 mm gap. The water flow can be controlled as a pulse flow, about 2 to 10 cycles per minute, through a control valve. VCS was developed for 2 years with several modifications[11]. The inlet water temperature varied between 10 and 55 °C at average flowrate around 1.2 liter/min. The compressor outlet temperature varied around 150–160 °C. Modification of VCS-1, VCS-2a, VCS-2b, VCS-2c, VCS-3 includes the change

The reaction Eq. (6) is exothermic and emits no radiation. A Geiger–Müller counter (JD-3001) monitoring radiation near the reactors during test, never showed a significant emission level above background, measured at 0.4 μSv/h maximum which is about two times of background or 0.4% occurrence. This seems favor the reaction Eq. (6).

In Ref.[15], it is proposed from nuclear physics that neutrinos and anti-neutrino pair at low energies can be formed during inelastic collisions of particles (electrons, ions, neutral atoms) during their thermal motion. Temperatures known to be produced in cavitation processes may exceed the calculated threshold in Ref.[15] for low energy neutrino and anti-neutrino pair production. The question then arises as to how the 4 particles ($_1{}^1H + e^- + \overline{\nu_e} + {}_8{}^{16}O$) in Eq. (6) can merge simultaneously. However due to the large de Broglie wavelength of low energy neutrinos, they could probably interact over an area big enough to include all the particles necessary for reaction Eq. (6) to take place.

Since the mechanism of nuclear reactions is not the focus of our research, we can only put forward intuitive and vague speculations with respect to our experimental observations. The present conjecture, reaction (6) or (7), is just two possibilities.

Another question raised is that "are there extra ^{17}O and ^{12}C to produce other compounds ?" Both ^{17}O and ^{12}C does not exist in monoatomic form. They will form compounds with other elements. This means that ^{12}C and isotope ^{17}O may be the intermediate.

Nuclear transmutation in ruptured copper pipe with large increases of C and O contents is one of the possible outcomes and has been observed in SEM/EDS[11]. Besides this, we found that three isotopes may be the outcomes: H_2O-17(heavy-oxygen water), isotope O_2 (^{16}O–^{17}O), and isotope CO_2 (^{12}C–^{16}O–^{17}O). This can be identified from isotope ratio analyses in m/z 19, 33 and 45 signals.

Conclusion

Cavitation may induce implosion of water vapor bubbles using various techniques[1–10]. In the previous study, we found that the heat exchange process in multiple-pipe heat exchanger produces anomalous excess heat and nuclear transmutation[11]. Recently, we have tested another 8 reactors and found that they also produce non-condensable gas. We suspected that ^{22}Ne and CO_2 may exist and is from nuclear reactions of water.

Fourteen gas samples were collected from eight reactors to perform mass spectrometry carefully using various methods. Two different methods for the identification of CO_2 were employed, while three different methods are employed for ^{22}Ne. All the results confirm that ^{22}Ne and CO_2 do exist in gas samples from reactors having excess heat.

In answering the question "how ^{22}Ne and CO_2 is produced in water", we conjecture a possible mechanism and find out that ^{12}C and isotope ^{17}O may be the intermediate. They possibly produce some other isotope compounds in gas from reactors having excess heat. Using isotope ratio analysis, we find out that they are H_2O-17 (heavy-oxygen water), isotope O_2 (^{16}O–^{17}O), and isotope CO_2 (^{12}C–^{16}O–^{17}O).

We also find that the reactions Eqs. (6)–(8) are the most-probable reactions whose output gas contents coincide with our observations—detected CO_2 and isotopes ^{22}Ne, H_2O-17, CO_2 (^{12}C–^{16}O–^{17}O) and O_2 (^{16}O–^{17}O), although the detailed mechanism is not known. This needs further basic research. Since the chances of getting

Original annotation: ***Conclusion & Interpretation***

<claim>The abstract states that the observed isotope gases verify that water triggers a peculiar nuclear reaction and produces energy.</claim>

<evidence>The paper later describes the proposed nuclear mechanism as "intuitive and vague speculations," says reactions (6) or (7) are "just two possibilities," reports no significant Geiger–Müller-detectable radiation above background, and acknowledges that the detailed mechanism is not known.</evidence>

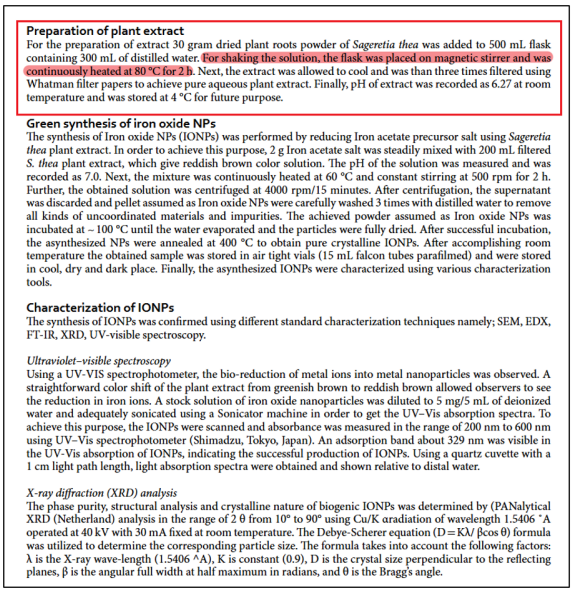
<inference>(1) "Verifies" implies definitive confirmation of the nuclear-reaction claim. (2) The paper's own discussion frames the mechanism as conjectural and unresolved rather than verified. (3) The absence of a significant direct radiation signature further limits, rather than strengthens, the evidential basis for claiming definitive nuclear verification.</inference>

<judgment>The abstract overstates the paper's evidence by claiming verification of a nuclear reaction when the mechanism is explicitly speculative and unresolved.</judgment>

Reason:

"Conclusion & Interpretation" should be changed to "Claims & Contributions." The broader concern about overstatement is justified, but the original annotation is not quite precise. An unresolved mechanism and the absence of radiation detectable by a Geiger–Müller counter do not, by themselves, rule out a nuclear reaction. The stronger basis for the error is that the isotope measurements are non-specific, leaving a gap between the observed nominal-mass signals, the claim of nuclear production, and the proposed energy-generation mechanism.

B.2.3 Case 3: Add



Preparation of plant extract

For the preparation of extract 30 gram dried plant roots powder of *Sageretia thea* was added to 500 mL flask containing 300 mL of distilled water. For shaking the solution, the flask was placed on magnetic stirrer and was continuously heated at 80 °C for 2 h. Next, the extract was allowed to cool and was than three times filtered using Whatman filter papers to achieve pure aqueous plant extract. Finally, pH of extract was recorded as 6.27 at room temperature and was stored at 4 °C for future purpose.

Green synthesis of iron oxide NPs

The synthesis of Iron oxide NPs (IONPs) was performed by reducing Iron acetate precursor salt using *Sageretia thea* plant extract. In order to achieve this purpose, 2 g Iron acetate salt was steadily mixed with 200 mL filtered *S. thea* plant extract, which give reddish brown color solution. The pH of the solution was measured and was recorded as 7.0. Next, the mixture was continuously heated at 60 °C and constant stirring at 500 rpm for 2 h. Further, the obtained solution was centrifuged at 4000 rpm/15 minutes. After centrifugation, the supernatant was discarded and pellet assumed as Iron oxide NPs were carefully washed 3 times with distilled water to remove all kinds of uncoordinated materials and impurities. The achieved powder assumed as Iron oxide NPs was incubated at ~ 100 °C until the water evaporated and the particles were fully dried. After successful incubation, the asynthesized NPs were annealed at 400 °C to obtain pure crystalline IONPs. After accomplishing room temperature the obtained sample was stored in air tight vials (15 mL falcon tubes parafilmed) and were stored in cool, dry and dark place. Finally, the asynthesized IONPs were characterized using various characterization tools.

Characterization of IONPs

The synthesis of IONPs was confirmed using different standard characterization techniques namely; SEM, EDX, FT-IR, XRD, UV-visible spectroscopy.

Ultraviolet–visible spectroscopy

Using a UV-VIS spectrophotometer, the bio-reduction of metal ions into metal nanoparticles was observed. A straightforward color shift of the plant extract from greenish brown to reddish brown allowed observers to see the reduction in iron ions. A stock solution of iron oxide nanoparticles was diluted to 5 mg/5 mL of deionized water and adequately sonicated using a Sonicator machine in order to get the UV–Vis absorption spectra. To achieve this purpose, the IONPs were scanned and absorbance was measured in the range of 200 nm to 600 nm using UV–Vis spectrophotometer (Shimadzu, Tokyo, Japan). An adsorption band about 329 nm was visible in the UV-Vis absorption of IONPs, indicating the successful production of IONPs. Using a quartz cuvette with a 1 cm light path length, light absorption spectra were obtained and shown relative to distal water.

X-ray diffraction (XRD) analysis

The phase purity, structural analysis and crystalline nature of biogenic IONPs was determined by (PANalytical XRD (Netherland) analysis in the range of 2 θ from 10° to 90° using Cu/K aradiation of wavelength 1.5406 ° A operated at 40 kV with 30 mA fixed at room temperature. The Debye-Scherer equation (D = Kλ/ βcos θ) formula was utilized to determine the corresponding particle size. The formula takes into account the following factors: λ is the X-ray wave-length (1.5406 ^A), K is constant (0.9), D is the crystal size perpendicular to the reflecting planes, β is the angular full width at half maximum in radians, and θ is the Bragg's angle.

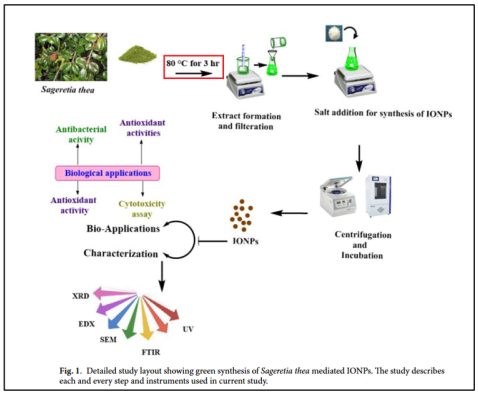


Fig. 1. Detailed study layout showing green synthesis of *Sageretia thea* mediated IONPs. The study describes each and every step and instruments used in current study.

Explanation: ***Method Design & Specification***

<claim>The Methods state that the extract solution was heated at 80 °C for 2 h during preparation of the plant extract.</claim>

<evidence>Fig. 1 labels the corresponding extract-formation and filtration step as '80 °C for 3 hr'.</evidence>

<inference>The Methods text and the study-layout figure refer to the same extract-preparation heating step but give conflicting durations, 2 h versus 3 hr.</inference>

<judgment>The extract-preparation heating duration is internally inconsistent and not reproducible as stated.</judgment>

C Experiment

C.1 Experiment Setup

C.1.1 Model Inference Configuration

Table 7: Inference configurations for the non-thinking and thinking settings. All models use a maximum generation length of 16,384 tokens. Reasoning is enabled for models providing an explicit switch, while `High` is used for models providing discrete reasoning levels.

	Non-thinking or Lowest-Available Setting			Thinking Setting			
Model	**Temp.**	**Top-p**	**Seed**	**Reasoning Configuration**	**Temp.**	**Top-p**	**Seed**
GPT-5.4	0	1.0	42	High	0	1.0	42
Claude-Sonnet-4.6	0	1.0	N/A	High	1.0 (fixed)	1.0	N/A
Gemini-3.1-Pro	0	1.0	42	High	0	1.0	42
Gemma-4-31B-it	0	1.0	42	Enabled	0	1.0	42
Qwen-3.6-Plus	0	1.0	42	Enabled	0	1.0	42
Qwen3.5-27B	0	1.0	42	Enabled	0	1.0	42
Qwen3.5-122B-A10B	0	1.0	42	Enabled	0	1.0	42
GLM-4.6V	0	1.0	42	Enabled	0	1.0	42
Mistral-Small-4	0	1.0	42	High	0	1.0	42
Llama-4-Maverick	0	1.0	42	N/A	N/A	N/A	N/A
Grok-4.3	0	1.0	42	High	0	1.0	42
Intern-S2-Preview	0	1.0	42	Enabled	0	1.0	42
Doubao-Seed-2.0-Pro	0	1.0	42	Enabled	0	1.0	42

N/A indicates that the corresponding capability or parameter is not supported by the provider API. Gemini-3.1-Pro uses its lowest available reasoning level as the non-thinking counterpart because its reasoning process cannot be fully disabled. Claude-Sonnet-4.6 requires temperature $= 1.0$ when thinking is enabled and does not support the seed parameter. Llama-4-Maverick does not expose a dedicated reasoning mode and is therefore evaluated only in the non-thinking setting.

Common generation protocol. All models receive the same benchmark inputs, task instruction, review criteria, and output schema. We request a single completion and set the maximum generation length to 16,384 tokens for every model.

Sampling and reasoning configurations. In the non-thinking setting, we standardize the sampling parameters to temperature $= 0$, top-p $= 1.0$, and seed $= 42$ whenever supported by the provider API. This configuration minimizes sampling variance and avoids model-specific decoding tuning. Claude-Sonnet-4.6 does not support a seed parameter. Since Gemini-3.1-Pro does not provide a thinking-off mode, we use its lowest available reasoning level as the non-thinking mode.

The evaluated APIs expose heterogeneous reasoning controls. Some models provide an explicit on/off switch, whereas others provide only discrete reasoning levels whose underlying computational budgets are not directly comparable. Although the three Qwen APIs additionally expose a numerical reasoning-token budget, comparable numerical controls are unavailable for most other models. Consequently, a unified reasoning-token budget cannot be applied across all models.

We therefore adopt a consistent provider-compatible protocol. Reasoning is enabled whenever an explicit switch is available, while models exposing only discrete reasoning levels are uniformly configured to `High`. All remaining sampling parameters are kept identical to the non-thinking setting whenever permitted by the provider API. Table 7 summarizes the complete configurations. This protocol controls all commonly exposed inference parameters, avoids arbitrary model-specific tuning, and maximizes comparability under current provider constraints.

C.1.2 PDF PARSING



Figure 6: Representative PDF parsing results across scientific papers. Colored overlays indicate detected layout regions, including text, figures, tables, formulas, captions, headers, and footers.

OCR Processing and Image Extraction. We render each PDF page at 300 DPI and use dots.mocr (Zheng et al., 2026), a state-of-the-art multimodal document parser, to convert it into a structured JSON representation containing text, formulas, tables, layout regions, and content-bearing figures, together with their locations and semantic categories. The 300 DPI setting preserves fine-grained visual details and improves the readability of small text, formulas, and complex layouts during parsing. In its official evaluation, dots.mocr achieves an overall score of 83.9 on olmOCR-Bench and leading TextEdit and ReadOrderEdit scores of 0.031 and 0.029 on OmniDocBench v1.5, demonstrating highly accurate text recovery, layout understanding, and reading order reconstruction. These results support its suitability for reliable and scalable scientific document parsing in SciAudit.

Image Filtering and Integration. We manually inspect the extracted visual regions and remove page headers, page footers, decorative icons, duplicated regions, and other non-content elements. Subfigures belonging to the same composite figure are merged to preserve their original structure, caption relationships, and surrounding context. The validated textual and visual blocks are then interleaved according to their original reading order, producing a unified multimodal representation that preserves the scientific content and document structure of the original paper.

C.1.3 Machine Consistency Experiment

Setup and Overall Finding. To assess robustness to machine evaluator choice, we sample 50 papers (25 Gold and 25 Silver) and predictions from four detection models, yielding 200 paper–detector cases. GPT-5.4, Qwen3.5-35B-A3B, and DeepSeek-V4-Pro independently evaluate all cases. Across error matching, continuous scoring, and detector ranking, the evaluators show strong consistency. Aggregate Q-score ICCs reach 0.966–0.970, and all evaluators produce the same detector ranking.

Error-Matching Agreement. Table 8 shows high agreement in error matching. Overall assignment agreement is 0.979, pair Jaccard is 0.930, Cohen's κ is 0.939, and Fleiss' κ is 0.932. Moreover, the complete error mapping is identical in 86.3% of evaluator-pair comparisons. These results show that the evaluators largely agree on which predicted errors correspond to the same ground-truth errors.

Table 8: Agreement of error-matching decisions across three evaluators. Pairwise metrics are macro-averaged over paper–detector cases and evaluator pairs.

Subset	Assignment Agr. ↑	Cohen's κ ↑	Pair Jaccard ↑	Exact Mapping ↑	Fleiss' κ ↑
Overall	**0.979**	**0.939**	**0.930**	**0.863**	**0.932**
Gold	0.973	0.928	0.918	0.853	0.918
Silver	0.985	0.951	0.941	0.873	0.945

Continuous-Score Agreement. Continuous scores are similarly stable across evaluators, as shown in Table 9. ICC(2,1) reaches 0.966, 0.968, and 0.970 for Q-Precision, Q-Recall, and Q-F1, respectively. Dimension-level ICCs are 0.916 for *Which*, 0.957 for *Where*, 0.832 for *What*, and 0.800 for *Why*. Pairwise Q-F1 differences remain small, with MAE ranging from 0.020 to 0.028, while Spearman correlations range from 0.966 to 0.977.

Table 9: Absolute and pairwise agreement of continuous evaluation scores.

Score Group	**Aggregate**			**Dimensions**			
Metric	Q-Precision	Q-Recall	Q-F1	Which	Where	What	Why
ICC(2,1) ↑	0.966	0.968	**0.970**	0.916	0.957	0.832	0.800

Evaluator Pair	**Q-F1 MAE** ↓	**Spearman** ρ ↑
GPT-5.4 vs. Qwen3.5-35B-A3B	0.028	0.971
GPT-5.4 vs. DeepSeek-V4-Pro	**0.020**	**0.977**
Qwen3.5-35B-A3B vs. DeepSeek-V4-Pro	0.026	0.966

Detector-Ranking Stability. Most importantly, evaluator choice does not affect the comparative conclusion. As shown in Table 10, all three evaluators produce exactly the same detector ordering: GPT-5.4, Doubao-Seed-2.0-Pro, Qwen3.5-27B, and Qwen-3.6-Plus. The largest Q-F1 range for any detector is only 3.2 percentage points, and no rank reversal occurs.

Table 10: Mean paper-level Q-F1 (%) and detector rankings across evaluators.

Detection Model	**Q-F1 by Evaluator (%)**			**Mean** ↑	**Rank** ↓
	GPT-5.4	**Qwen3.5-35B-A3B**	**DeepSeek-V4-Pro**		
GPT-5.4	**32.9**	**36.1**	**34.0**	**34.3**	**1**
Doubao-Seed-2.0-Pro	31.3	33.4	31.2	32.0	2
Qwen3.5-27B	19.8	21.1	19.7	20.2	3
Qwen-3.6-Plus	17.5	19.6	17.2	18.1	4

Conclusion. Overall, the evaluation is robust to the choice among the three tested machine evaluators. Error matching is highly consistent, aggregate Q-scores show strong agreement, and detector rankings remain unchanged. Although the fine-grained *What* and *Why* dimensions show relatively greater evaluator variation, this variation has little effect on aggregate scores or model comparisons.

Table 11: Paper-level performance on the Silver set grouped by the model used to generate each draft. Each cell reports Q-F1 / Q-Recall / Q-Precision. All detection models are evaluated in thinking mode.

Detect \ Draft	GPT-5.4	Claude-Sonnet-4.6	Gemini-3.1-Pro
Grok-4.3	19.6 / 11.8 / 69.1	19.1 / 11.7 / 62.1	23.4 / 14.3 / 77.0
Claude-Sonnet-4.6	21.5 / 13.4 / 71.3	19.5 / 12.0 / 64.6	24.4 / 14.9 / 79.0
Gemini-3.1-Pro	29.9 / 21.0 / 56.2	29.4 / 21.8 / 49.6	35.6 / 25.0 / 66.9
Qwen-3.6-Plus	29.6 / 20.3 / 57.5	29.7 / 20.9 / 54.7	38.3 / 26.6 / 73.3
Doubao-Seed-2.0-Pro	32.7 / 21.7 / 70.6	30.4 / 20.6 / 64.0	40.8 / 28.0 / 79.3
GPT-5.4	48.4 / 37.5 / 70.2	42.3 / 34.1 / 58.2	52.9 / 41.4 / 75.6

Detect \ Draft	Grok-4.3	Qwen-3.6-Plus	Doubao-Seed-2.0-Pro
Grok-4.3	29.1 / 19.5 / 69.7	22.9 / 14.3 / 66.8	26.8 / 17.0 / 72.9
Claude-Sonnet-4.6	28.9 / 19.8 / 67.5	22.9 / 14.8 / 67.6	30.1 / 19.7 / 79.4
Gemini-3.1-Pro	30.8 / 24.3 / 48.6	31.4 / 23.8 / 50.5	35.0 / 26.0 / 57.5
Qwen-3.6-Plus	33.2 / 26.3 / 51.3	34.4 / 25.6 / 59.0	38.3 / 27.9 / 65.5
Doubao-Seed-2.0-Pro	41.1 / 31.4 / 67.6	38.2 / 27.0 / 69.1	42.4 / 30.1 / 76.6
GPT-5.4	42.5 / 37.4 / 53.0	45.5 / 38.4 / 58.9	54.2 / 45.0 / 70.2

C.2 Supplementary Results

Generator–Detector Cross-Evaluation. Table 11 reports detection performance separately for Silver papers produced by each Draft model. This analysis examines whether the benchmark results are sensitive to a particular generator and whether detectors benefit from recognizing errors generated by themselves.

Despite variation across Draft sources, the detector ranking remains highly consistent. GPT-5.4 achieves the highest Q-F1 and Q-Recall on all six generator-specific subsets, while stronger detectors remain competitive across every Draft source. Moreover, we observe no systematic self-generator advantage: several detectors perform better on drafts generated by other models than on their own. These results indicate that Silver does not primarily reward generator-specific shortcuts and provides a stable basis for cross-model comparison.

Table 12: Error-level average matched-error scores across the four evaluation dimensions on the Gold and Silver sets. Each entry reports thinking / non-thinking. Best and second-best scores are marked in bold and underline, respectively, within proprietary and open-source MLLMs.

Models	Gold				Silver			
	Which	Where	What	Why	Which	Where	What	Why
Proprietary MLLMs								
Grok-4.3	80.6 / 78.1	67.5 / 59.3	74.3 / 52.7	79.8 / 57.0	85.3 / 78.4	75.6 / 69.2	72.8 / 54.6	85.4 / 44.8
Claude-Sonnet-4.6	78.6 / 80.9	71.5 / 64.5	72.6 / 67.4	68.7 / 75.6	83.7 / 83.9	78.0 / 76.2	**77.2** / 66.9	**92.7** / 66.6
Gemini-3.1-Pro	85.9 / **86.2**	67.6 / 67.7	69.6 / 70.6	78.0 / 77.4	86.7 / **88.1**	83.8 / 81.9	69.2 / 66.7	85.2 / 71.5
Qwen-3.6-Plus	83.1 / 80.4	70.9 / 69.0	74.7 / 68.7	80.4 / 74.0	87.8 / 86.7	80.2 / 82.5	68.5 / 71.6	87.5 / 86.6
Doubao-Seed-2.0-Pro	83.9 / 79.1	73.8 / 71.2	73.7 / 69.8	80.1 / 77.5	87.6 / 84.3	**85.6** / **85.6**	64.6 / 64.6	87.7 / **88.5**
GPT-5.4	**87.9** / 81.2	**75.8** / **71.3**	**75.3** / **70.6**	**85.6** / **79.1**	**89.3** / 85.9	83.8 / 82.5	71.3 / **72.5**	87.9 / 87.4
Open-source MLLMs								
Llama-4-Maverick	- / 74.6	- / 54.5	- / 48.6	- / 53.7	- / 84.4	- / 76.6	- / 41.0	- / 51.3
GLM-4.6V	75.7 / 74.6	59.0 / 60.2	60.4 / 52.2	73.1 / 57.1	84.5 / 85.2	77.4 / 78.1	56.5 / 51.2	62.3 / 55.5
Mistral-Small-4	76.3 / 72.2	53.7 / 57.5	50.9 / 42.1	66.0 / 45.3	82.3 / 77.6	66.2 / 73.2	54.3 / 49.0	49.9 / 49.6
Intern-S2-Preview	80.7 / 71.5	44.9 / 46.7	69.7 / 58.8	77.4 / 74.9	84.9 / 78.6	52.0 / 56.0	**77.1** / 68.9	72.3 / 65.8
Gemma-4-31B-it	**82.9** / 79.8	65.6 / 68.1	65.7 / 61.9	77.6 / **77.2**	87.3 / **87.3**	**81.7** / **84.4**	63.7 / 61.8	83.8 / 74.7
Qwen3.5-122B-A10B	80.7 / 79.9	66.6 / 65.4	71.2 / 65.3	76.2 / 69.1	86.5 / 85.2	77.5 / 80.1	70.8 / 65.7	81.7 / 71.2
Qwen3.5-27B	81.8 / **80.1**	**67.6** / **71.7**	**72.0** / **69.2**	**81.5** / 76.4	**87.7** / 87.2	78.1 / 83.7	72.6 / **69.9**	**84.9** / **87.7**

Fine-Grained Quality of Matched Errors. Table 12 evaluates the quality of successfully matched predictions along the *Which*, *Where*, *What*, and *Why* dimensions. Unlike aggregate detection scores, this analysis examines whether a detected error is correctly categorized, localized, supported by evidence, and explained.

Across both Gold and Silver, matched predictions generally obtain strong scores in error classification and reasoning, while leading models also show reliable localization and evidence grounding. The broadly similar patterns on the two subsets further support the consistency of the structured annotations and the quality-aware evaluation framework. Thinking generally improves or preserves matched-error quality, particularly for evidence coverage and reasoning, showing that additional inference contributes not only to discovering more errors but also to producing more complete verification.

Table 13: Original paper-level performance on the Gold and Silver sets. Each entry reports thinking / non-thinking. Best and second-best scores are marked in bold and underline, respectively, within proprietary and open-source MLLMs.

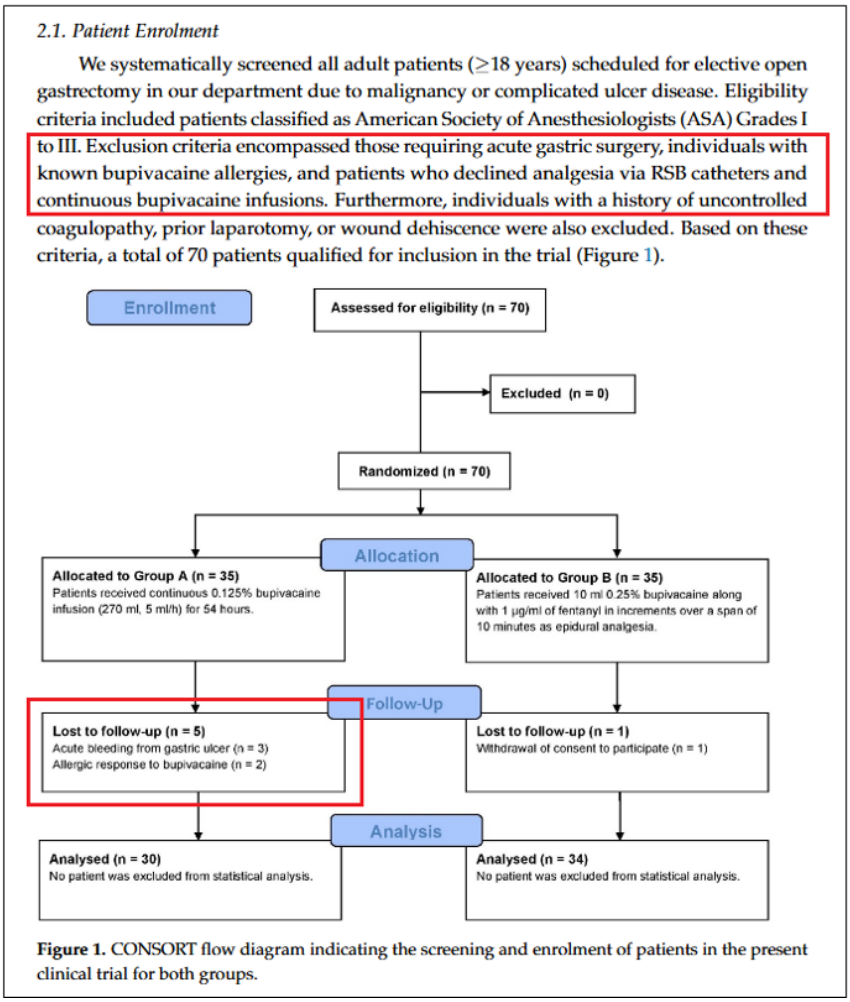
Models	Gold			Silver		
	F1	Recall	Precision	F1	Recall	Precision
Proprietary MLLMs						
Grok-4.3	20.7 / 10.1	12.9 / 8.0	64.2 / 19.5	29.2 / 7.5	18.4 / 5.4	87.5 / 16.4
Claude-Sonnet-4.6	22.8 / 16.9	14.4 / 14.2	**65.8** / 25.9	29.4 / 25.6	18.8 / 17.6	86.0 / 62.2
Gemini-3.1-Pro	29.5 / 26.0	21.6 / 18.5	54.2 / **51.2**	39.3 / 36.7	29.1 / 26.2	67.8 / 70.3
Qwen-3.6-Plus	31.5 / 20.0	23.1 / 13.7	56.1 / 43.9	41.6 / 24.1	30.1 / 15.1	74.2 / 73.3
Doubao-Seed-2.0-Pro	31.2 / 28.5	22.2 / 21.8	61.6 / 46.5	45.9 / 46.9	32.2 / 35.2	**87.6** / **77.1**
GPT-5.4	**47.7** / **33.7**	**40.0** / **30.0**	65.0 / 42.1	**57.5** / **50.2**	**46.9** / **43.3**	78.0 / 63.4
Open-source MLLMs						
Llama-4-Maverick	- / 3.3	- / 2.3	- / 9.6	- / 8.6	- / 5.3	- / 30.9
GLM-4.6V	7.9 / 7.5	4.8 / 4.7	30.3 / 28.4	12.0 / 9.6	7.2 / 5.5	50.4 / 46.7
Mistral-Small-4	17.1 / 12.9	18.4 / **15.2**	18.2 / 12.9	19.8 / 13.9	17.8 / 13.6	26.0 / 16.1
Intern-S2-Preview	24.2 / 11.8	17.1 / 7.9	47.0 / 28.0	37.5 / 15.0	26.2 / 9.5	**72.6** / 42.9
Gemma-4-31B-it	27.3 / 15.5	19.7 / 9.8	50.4 / **42.6**	35.6 / 22.2	24.6 / 13.5	72.3 / **76.8**
Qwen3.5-122B-A10B	29.2 / 19.3	21.2 / 14.1	**53.6** / 34.9	**39.2** / 24.8	28.1 / 16.6	70.5 / 60.5
Qwen3.5-27B	**30.8** / **19.4**	**23.7** / 13.6	49.7 / 37.8	38.9 / **29.9**	**28.8** / **19.7**	65.1 / 69.5

Robustness under Conventional Metrics. Table 13 reports conventional F1, recall, and precision, treating every successfully matched error as a binary true positive. This provides a robustness check for whether the main findings depend on the proposed quality-aware scoring framework.

The conventional metrics reproduce the main trends observed under quality-aware evaluation. GPT-5.4 remains the strongest proprietary model in overall F1 and recall on both Gold and Silver, while the leading Qwen3.5 models remain the strongest open-source systems. Model rankings are also broadly consistent across the two subsets, and thinking improves F1 and recall in most settings. These results confirm that the main conclusions are robust to the choice of evaluation metric.

C.3 Failure Cases of Models

C.3.1 Case 1: Hallucination



2.1. Patient Enrolment

We systematically screened all adult patients (≥18 years) scheduled for elective open gastrectomy in our department due to malignancy or complicated ulcer disease. Eligibility criteria included patients classified as American Society of Anesthesiologists (ASA) Grades I to III. Exclusion criteria encompassed those requiring acute gastric surgery, individuals with known bupivacaine allergies, and patients who declined analgesia via RSB catheters and continuous bupivacaine infusions. Furthermore, individuals with a history of uncontrolled coagulopathy, prior laparotomy, or wound dehiscence were also excluded. Based on these criteria, a total of 70 patients qualified for inclusion in the trial (Figure 1).

Figure 1. CONSORT flow diagram indicating the screening and enrolment of patients in the present clinical trial for both groups.

Model Explanation: ***Data & Materials Handling***

<claim>The study enrolled and randomized 2 patients with bupivacaine allergy to the RSB intervention group, despite this being an explicit exclusion criterion.</claim>

<evidence>The evidence confirms bupivacaine allergy is an exclusion criterion, yet 2 allocated patients experienced allergic reactions to bupivacaine post-allocation, indicating they were not properly screened or excluded prior to enrolment.</evidence>

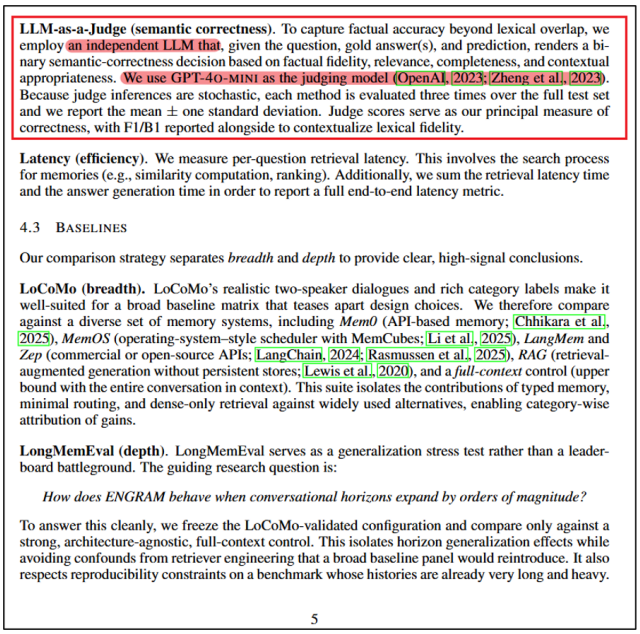
<inference>(1) The enrolment process failed to apply the stated exclusion criterion, resulting in invalid enrolment of ineligible participants. (2) This flaw in participant selection introduces bias and compromises the validity of the study sample.</inference>

<judgment>The study contains an invalid case selection error due to failure to apply pre-specified exclusion criteria during enrolment.</judgment>

Reason:

The prediction incorrectly equates an allergic response occurring after allocation with a known pre-existing bupivacaine allergy. The exclusion criterion concerns known allergy, whereas Figure 1 only reports two post-allocation allergic responses; nothing in the paper establishes that these patients were known to be allergic during screening.

C.3.2 Case 2: Reasoning Error



LLM-as-a-Judge (semantic correctness). To capture factual accuracy beyond lexical overlap, we employ an independent LLM that, given the question, gold answer(s), and prediction, renders a binary semantic-correctness decision based on factual fidelity, relevance, completeness, and contextual appropriateness. We use GPT-4o-mini as the judging model (OpenAI, 2023; Zheng et al., 2023). Because judge inferences are stochastic, each method is evaluated three times over the full test set and we report the mean ± one standard deviation. Judge scores serve as our principal measure of correctness, with F1/B1 reported alongside to contextualize lexical fidelity.

Latency (efficiency). We measure per-question retrieval latency. This involves the search process for memories (e.g., similarity computation, ranking). Additionally, we sum the retrieval latency time and the answer generation time in order to report a full end-to-end latency metric.

4.3 Baselines

Our comparison strategy separates *breadth* and *depth* to provide clear, high-signal conclusions.

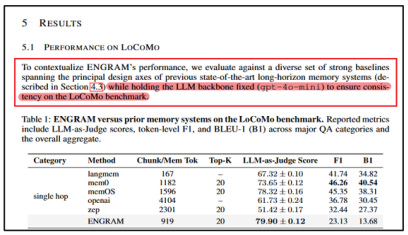
LoCoMo (breadth). LoCoMo's realistic two-speaker dialogues and rich category labels make it well-suited for a broad baseline matrix that teases apart design choices. We therefore compare against a diverse set of memory systems, including *Mem0* (API-based memory; Chhikara et al., 2025), *MemOS* (operating-system–style scheduler with MemCubes; Li et al., 2025), *LangMem* and *Zep* (commercial or open-source APIs; LangChain, 2024; Rasmussen et al., 2025), *RAG* (retrieval-augmented generation without persistent stores; Lewis et al., 2020), and a *full-context* control (upper bound with the entire conversation in context). This suite isolates the contributions of typed memory, minimal routing, and dense-only retrieval against widely used alternatives, enabling category-wise attribution of gains.

LongMemEval (depth). LongMemEval serves as a generalization stress test rather than a leaderboard battleground. The guiding research question is:

How does ENGRAM behave when conversational horizons expand by orders of magnitude?

To answer this cleanly, we freeze the LoCoMo-validated configuration and compare only against a strong, architecture-agnostic, full-context control. This isolates horizon generalization effects while avoiding confounds from retriever engineering that a broad baseline panel would reintroduce. It also respects reproducibility constraints on a benchmark whose histories are already very long and heavy.

5



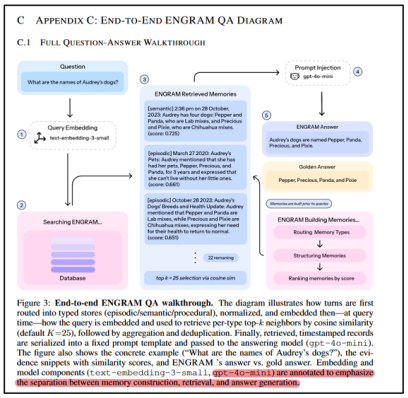
5 Results

5.1 Performance on LoCoMo

To contextualize ENGRAM's performance, we evaluate against a diverse set of strong baselines spanning the principal design axes of previous state-of-the-art long-horizon memory systems (described in Section 4.3) while holding the LLM backbone fixed (`gpt-4o-mini`) to ensure consistency on the LoCoMo benchmark.

Table 1: **ENGRAM versus prior memory systems on the LoCoMo benchmark.** Reported metrics include LLM-as-Judge scores, token-level F1, and BLEU-1 (B1) across major QA categories and the overall aggregate.

Category	Method	Chunk/Mem Tok	Top-K	LLM-as-Judge Score	F1	B1
single hop	langmem	167	–	67.32 ± 0.10	41.74	34.82
	mem0	1182	20	73.65 ± 0.12	**46.26**	**40.54**
	memOS	1596	20	78.32 ± 0.16	45.35	38.31
	openai	4104	–	61.73 ± 0.24	36.78	30.45
	zep	2301	20	51.42 ± 0.17	32.44	27.37
	ENGRAM	919	20	**79.90 ± 0.12**	23.13	13.68



C Appendix C: End-to-End ENGRAM QA Diagram

C.1 Full Question-Answer Walkthrough

Figure 3: **End-to-end ENGRAM QA walkthrough.** The diagram illustrates how turns are first routed into typed stores (episodic/semantic/procedural), normalized, and embedded then—at query time—how the query is embedded and used to retrieve per-type top-k neighbors by cosine similarity (default $K=25$), followed by aggregation and deduplication. Finally, retrieved, timestamped records are serialized into a fixed prompt template and passed to the answering model (`gpt-4o-mini`). The figure also shows the concrete example ("What are the names of Audrey's dogs?"), the evidence snippets with similarity scores, and ENGRAM 's answer vs. gold answer. Embedding and model components (`text-embedding-3-small`, `gpt-4o-mini`) are annotated to emphasize the separation between memory construction, retrieval, and answer generation.

Model Explanation: ***Evaluation & Experimental Validity***

<claim>The paper claims to use an 'independent LLM' judge for semantic correctness.</claim>,

<evidence>The methods state that the judge is GPT-4O-MINI, while the experimental setup and Figure 3 show that the evaluated answering backbone is also gpt-4o-mini.</evidence>,

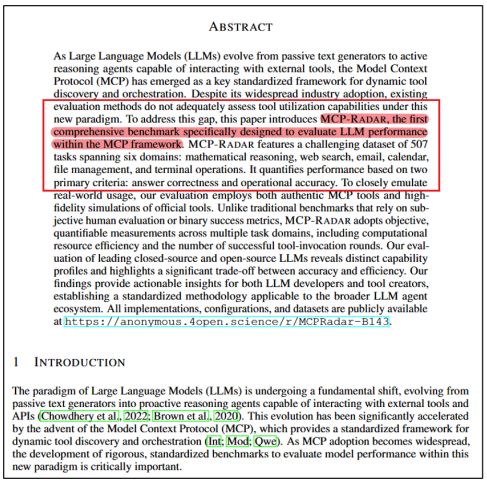
<inference>(1) A judge cannot be described as independent from the evaluated system when it is the same model family used to produce the evaluated outputs. (2) This matters because LLM-as-a-judge validity depends on evaluator independence; otherwise shared model biases can distort comparisons.</inference>,

<judgment>The evaluation mischaracterizes the judge as independent even though the same GPT-4o-mini model family is used for both answer generation and judging, weakening the validity of the main correctness metric.</judgment>

Reason:

The paper does use GPT-4o-mini for both answer generation and judging, but this does not logically contradict calling the judge independent: independence can refer to a separate evaluation invocation that receives only the question, reference, and prediction. Moreover, all compared methods use the same answering backbone, so the evidence does not establish a differential validity failure.

C.3.3 Case 3: Speculation

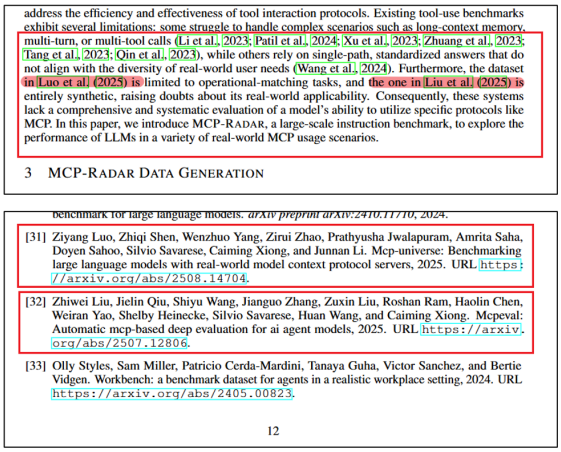


Abstract

As Large Language Models (LLMs) evolve from passive text generators to active reasoning agents capable of interacting with external tools, the Model Context Protocol (MCP) has emerged as a key standardized framework for dynamic tool discovery and orchestration. Despite its widespread industry adoption, existing evaluation methods do not adequately assess tool utilization capabilities under this new paradigm. To address this gap, this paper introduces MCP-RADAR, the first comprehensive benchmark specifically designed to evaluate LLM performance within the MCP framework. MCP-RADAR features a challenging dataset of 507 tasks spanning six domains: mathematical reasoning, web search, email, calendar, file management, and terminal operations. It quantifies performance based on two primary criteria: answer correctness and operational accuracy. To closely emulate real-world usage, our evaluation employs both authentic MCP tools and high-fidelity simulations of official tools. Unlike traditional benchmarks that rely on subjective human evaluation or binary success metrics, MCP-RADAR adopts objective, quantifiable measurements across multiple task domains, including computational resource efficiency and the number of successful tool-invocation rounds. Our evaluation of leading closed-source and open-source LLMs reveals distinct capability profiles and highlights a significant trade-off between accuracy and efficiency. Our findings provide actionable insights for both LLM developers and tool creators, establishing a standardized methodology applicable to the broader LLM agent ecosystem. All implementations, configurations, and datasets are publicly available at https://anonymous.4open.science/r/MCPRadar-B143.

1 Introduction

The paradigm of Large Language Models (LLMs) is undergoing a fundamental shift, evolving from passive text generators into proactive reasoning agents capable of interacting with external tools and APIs (Chowdhery et al., 2022; Brown et al., 2020). This evolution has been significantly accelerated by the advent of the Model Context Protocol (MCP), which provides a standardized framework for dynamic tool discovery and orchestration (Int; Mod; Qwe). As MCP adoption becomes widespread, the development of rigorous, standardized benchmarks to evaluate model performance within this new paradigm is critically important.



address the efficiency and effectiveness of tool interaction protocols. Existing tool-use benchmarks exhibit several limitations: some struggle to handle complex scenarios such as long-context memory, multi-turn, or multi-tool calls (Li et al., 2023; Patil et al., 2024; Xu et al., 2023; Zhuang et al., 2023; Tang et al., 2023; Qin et al., 2023), while others rely on single-path, standardized answers that do not align with the diversity of real-world user needs (Wang et al., 2024). Furthermore, the dataset in Luo et al. (2025) is limited to operational-matching tasks, and the one in Liu et al. (2025) is entirely synthetic, raising doubts about its real-world applicability. Consequently, these systems lack a comprehensive and systematic evaluation of a model's ability to utilize specific protocols like MCP. In this paper, we introduce MCP-RADAR, a large-scale instruction benchmark, to explore the performance of LLMs in a variety of real-world MCP usage scenarios.

3 MCP-RADAR Data Generation

benchmark for large language models. *arXiv preprint arXiv:2410.11710*, 2024.

[31] Ziyang Luo, Zhiqi Shen, Wenzhuo Yang, Zirui Zhao, Prathyusha Jwalapuram, Amrita Saha, Doyen Sahoo, Silvio Savarese, Caiming Xiong, and Junnan Li. Mcp-universe: Benchmarking large language models with real-world model context protocol servers, 2025. URL https://arxiv.org/abs/2508.14704.

[32] Zhiwei Liu, Jielin Qiu, Shiyu Wang, Jianguo Zhang, Zuxin Liu, Roshan Ram, Haolin Chen, Weiran Yao, Shelby Heinecke, Silvio Savarese, Huan Wang, and Caiming Xiong. Mcpeval: Automatic mcp-based deep evaluation for ai agent models, 2025. URL https://arxiv.org/abs/2507.12806.

[33] Olly Styles, Sam Miller, Patricio Cerda-Mardini, Tanaya Guha, Victor Sanchez, and Bertie Vidgen. Workbench: a benchmark dataset for agents in a realistic workplace setting, 2024. URL https://arxiv.org/abs/2405.00823.

12

Model Explanation: ***Claims & Contributions***

<claim>The paper claims MCP-RADAR is the first benchmark specifically designed for evaluating LLMs in the MCP setting.</claim>

<evidence>The paper's own related-work section cites prior MCP-specific evaluation efforts, and references [31] and [32] are explicitly titled as MCP benchmarking/evaluation works.</evidence>

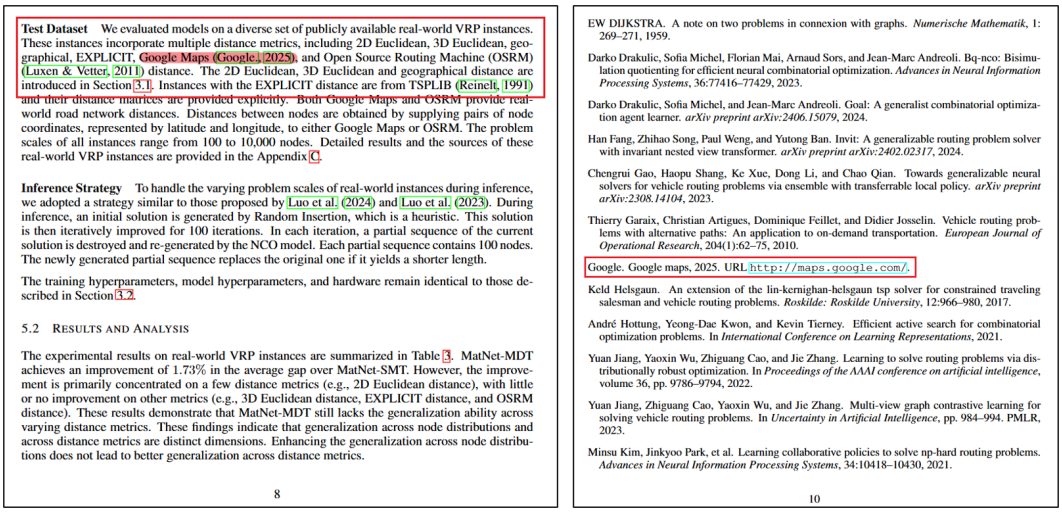
<inference>(1) Once the paper acknowledges existing MCP-specific benchmarks/evaluation frameworks, a blanket 'first benchmark' claim is no longer supported. (2) At most, the paper could argue that MCP-RADAR differs in scope or coverage, but the categorical priority claim is broader than the evidence allows.</inference>

<judgment>The contribution claim is overstated: the paper's own citations contradict the assertion that MCP-RADAR is the first MCP benchmark.</judgment>

Reason:

The cited works establish that prior MCP-related evaluation efforts existed, but their titles and the paper's characterization do not establish that they were equally comprehensive or had the same scope as MCP-RADAR. Thus, the qualified priority claim cannot be objectively disproved without a broader literature and scope comparison.

C.3.4 Case 4: Limitation



Test Dataset We evaluated models on a diverse set of publicly available real-world VRP instances. These instances incorporate multiple distance metrics, including 2D Euclidean, 3D Euclidean, geographical, EXPLICIT, Google Maps (Google, 2025), and Open Source Routing Machine (OSRM) (Luxen & Vetter, 2011) distance. The 2D Euclidean, 3D Euclidean and geographical distance are introduced in Section 3.1. Instances with the EXPLICIT distance are from TSPLIB (Reinelt, 1991) and their distance matrices are provided explicitly. Both Google Maps and OSRM provide real-world road network distances. Distances between nodes are obtained by supplying pairs of node coordinates, represented by latitude and longitude, to either Google Maps or OSRM. The problem scales of all instances range from 100 to 10,000 nodes. Detailed results and the sources of these real-world VRP instances are provided in the Appendix C.

Inference Strategy To handle the varying problem scales of real-world instances during inference, we adopted a strategy similar to those proposed by Luo et al. (2024) and Luo et al. (2023). During inference, an initial solution is generated by Random Insertion, which is a heuristic. This solution is then iteratively improved for 100 iterations. In each iteration, a partial sequence of the current solution is destroyed and re-generated by the NCO model. Each partial sequence contains 100 nodes. The newly generated partial sequence replaces the original one if it yields a shorter length.

The training hyperparameters, model hyperparameters, and hardware remain identical to those described in Section 3.2.

5.2 Results and Analysis

The experimental results on real-world VRP instances are summarized in Table 3. MatNet-MDT achieves an improvement of 1.73% in the average gap over MatNet-SMT. However, the improvement is primarily concentrated on a few distance metrics (e.g., 2D Euclidean distance), with little or no improvement on other metrics (e.g., 3D Euclidean distance, EXPLICIT distance, and OSRM distance). These results demonstrate that MatNet-MDT still lacks the generalization ability across varying distance metrics. These findings indicate that generalization across node distributions and across distance metrics are distinct dimensions. Enhancing the generalization across node distributions does not lead to better generalization across distance metrics.

8

EW DIJKSTRA. A note on two problems in connexion with graphs. *Numerische Mathematik*, 1: 269–271, 1959.

Darko Drakulic, Sofia Michel, Florian Mai, Arnaud Sors, and Jean-Marc Andreoli. Bq-nco: Bisimulation quotienting for efficient neural combinatorial optimization. *Advances in Neural Information Processing Systems*, 36:77416–77429, 2023.

Darko Drakulic, Sofia Michel, and Jean-Marc Andreoli. Goal: A generalist combinatorial optimization agent learner. *arXiv preprint arXiv:2406.15079*, 2024.

Han Fang, Zhihao Song, Paul Weng, and Yutong Ban. Invit: A generalizable routing problem solver with invariant nested view transformer. *arXiv preprint arXiv:2402.02317*, 2024.

Chengrui Gao, Haopu Shang, Ke Xue, Dong Li, and Chao Qian. Towards generalizable neural solvers for vehicle routing problems via ensemble with transferrable local policy. *arXiv preprint arXiv:2308.14104*, 2023.

Thierry Garaix, Christian Artigues, Dominique Feillet, and Didier Josselin. Vehicle routing problems with alternative paths: An application to on-demand transportation. *European Journal of Operational Research*, 204(1):62–75, 2010.

Google. Google maps, 2025. URL http://maps.google.com/.

Keld Helsgaun. An extension of the lin-kernighan-helsgaun tsp solver for constrained traveling salesman and vehicle routing problems. *Roskilde: Roskilde University*, 12:966–980, 2017.

André Hottung, Yeong-Dae Kwon, and Kevin Tierney. Efficient active search for combinatorial optimization problems. In *International Conference on Learning Representations*, 2021.

Yuan Jiang, Yaoxin Wu, Zhiguang Cao, and Jie Zhang. Learning to solve routing problems via distributionally robust optimization. In *Proceedings of the AAAI conference on artificial intelligence*, volume 36, pp. 9786–9794, 2022.

Yuan Jiang, Zhiguang Cao, Yaoxin Wu, and Jie Zhang. Multi-view graph contrastive learning for solving vehicle routing problems. In *Uncertainty in Artificial Intelligence*, pp. 984–994. PMLR, 2023.

Minsu Kim, Jinkyoo Park, et al. Learning collaborative policies to solve np-hard routing problems. *Advances in Neural Information Processing Systems*, 34:10418–10430, 2021.

10

Model Explanation: ***Citation Alignment & Reference***

<claim>The paper cites Google (2025) as the source for Google Maps road network distance metric data.</claim>

<evidence>The provided reference is a generic, non-specific link to the Google Maps homepage, with no verifiable details about the implementation, version, or access details for the distance metric used.</evidence>

<inference>The reference does not meet academic citation standards and cannot be used to verify or support the claim about the Google Maps distance metric used in the study.</inference>

<judgment>The Google Maps citation is an invalid, non-verifiable reference.</judgment>

Reason:

The Google Maps reference is an identifiable, real source. Although a specific API, version, query settings, and access date would improve reproducibility of the reported road-network distances, their absence is a documentation/reproducibility limitation rather than an objectively invalid citation or substantive scientific error.

C.3.5 Case 5: Overstatement

What is SIP? SIP operates by analyzing learned representations early in the training process. It ensures that task-relevant features are distinct and interpretable before issues arise, leveraging spectral analysis to separate discriminative features. SIP uses eigenvalue separation to ensure stability and interpretability, while clipping prevents outliers from distorting the feature space, ensuring stable learning.

Contributions. Our work makes three key contributions:

(i) **Methodology:** We introduce SIP, a preemptive tool for evaluating probe reliability, ensuring interpretability and stable feature separation from the start of training.

(ii) **Theory:** We establish a novel connection between feature estimation errors and misclassification risk, demonstrating how SIP's spectral guarantees provide a more robust and proactive approach to model stability.

(iii) **Empirics:** Through controlled synthetic experiments, we show that SIP outperforms traditional methods by detecting instability up to 20

In summary, SIP guarantees interpretability and stability from the start, forming a foundation for extending this framework to more complex models. SIP ensures that interpretability and stability are addressed from the beginning, improving model reliability and efficiency. The following sections will present the theoretical foundations and implementation details of SIP, followed by a comparison with traditional methods.

2 Related Work

Opening. Probing methods have become a central tool in analyzing neural representations. However, despite their widespread use, the reliability of these probes remains a subject of debate. Existing studies have examined probes from various perspectives—diagnostic, spectral, and robustness—but these approaches often remain disconnected. This fragmentation leads to the central question: *under what conditions can probe outputs be trusted?* This work addresses this gap by integrating these perspectives into a unified framework, offering a verifiable criterion for probe stability.

Probing Methods. Linear probes were initially introduced as a method for interpreting intermediate neural representations by training lightweight classifiers on frozen model features (Alain & Bengio, 2017). However, their reliability has been questioned. Studies have shown that probes can capture spurious correlations that do not reflect the true underlying information (Hewitt & Liang, 2019). Alternative methods have been proposed, such as minimum description length (MDL) for quantifying the sufficiency of representations (Voita & Titov, 2020; Pimentel et al., 2020; Pimentel & Cotterell, 2021), and counterfactual removal techniques for testing causal relevance (Elazar et al., 2021; Ravfogel et al., 2020). Surveys have consolidated these developments and highlighted their limitations, particularly the heuristic or ex-post nature of these diagnostics (Belinkov, 2022). While valuable, these methods evaluate probe behavior after training, without providing a *verifiable, pre-deployment condition* that guarantees the stability of probe predictions.

Model Explanation: ***Language & Expression***

<claim>The sentence is intended to communicate the core empirical contribution of the work regarding SIP's performance relative to traditional methods.</claim>

<evidence>The sentence is incomplete, cut off mid-phrase, with no clarification of the meaning of the value 20 or the full stated performance advantage.</evidence>

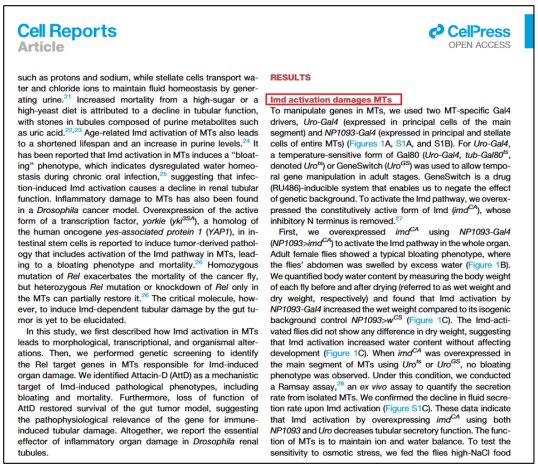
<inference>(1) An incomplete sentence fails to convey the intended empirical contribution clearly to readers. (2) The missing content obscures the magnitude and nature of the stated performance improvement over baseline methods.</inference>

<judgment>The empirical contribution statement is incomplete and fails to communicate its intended meaning clearly.</judgment>

Reason:

The sentence simply ends after "20" which appears to be a minor punctuation or formatting issue. It does not by itself indicate any substantive scientific error.

C.3.6 Case 6: OCR Noise



Cell Reports
Article

CellPress
OPEN ACCESS

such as protons and sodium, while stellate cells transport water and chloride ions to maintain fluid homeostasis by generating urine.[21] Increased mortality from a high-sugar or a high-yeast diet is attributed to a decline in tubular function, with stones in tubules composed of purine metabolites such as uric acid.[22,23] Age-related Imd activation of MTs also leads to a shortened lifespan and an increase in purine levels.[24] It has been reported that Imd activation in MTs induces a "bloating" phenotype, which indicates dysregulated water homeostasis during chronic oral infection,[25] suggesting that infection-induced Imd activation causes a decline in renal tubular function. Inflammatory damage to MTs has also been found in a *Drosophila* cancer model. Overexpression of the active form of a transcription factor, *yorkie* (*yki*^{3SA}), a homolog of the human oncogene *yes-associated protein 1* (*YAP1*), in intestinal stem cells is reported to induce tumor-derived pathology that includes activation of the Imd pathway in MTs, leading to a bloating phenotype and mortality.[26] Homozygous mutation of *Rel* exacerbates the mortality of the cancer fly, but heterozygous *Rel* mutation or knockdown of *Rel* only in the MTs can partially restore it.[26] The critical molecule, however, to induce Imd-dependent tubular damage by the gut tumor is yet to be elucidated.

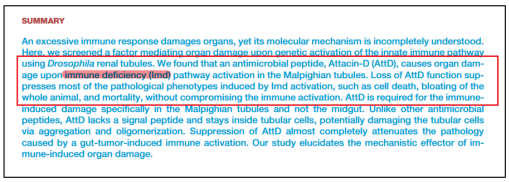
In this study, we first described how Imd activation in MTs leads to morphological, transcriptional, and organismal alterations. Then, we performed genetic screening to identify the Rel target genes in MTs responsible for Imd-induced organ damage. We identified Attacin-D (AttD) as a mechanistic target of Imd-induced pathological phenotypes, including bloating and mortality. Furthermore, loss of function of AttD restored survival of the gut tumor model, suggesting the pathophysiological relevance of the gene for immune-induced tubular damage. Altogether, we report the essential effector of inflammatory organ damage in *Drosophila* renal tubules.

RESULTS

Imd activation damages MTs

To manipulate genes in MTs, we used two MT-specific Gal4 drivers, *Uro-Gal4* (expressed in principal cells of the main segment) and *NP1093-Gal4* (expressed in principal and stellate cells of entire MTs) (Figures 1A, S1A, and S1B). For *Uro-Gal4*, a temperature-sensitive form of Gal80 (*Uro-Gal4, tub-Gal80*^{ts}, denoted *Uro*^{ts}) or GeneSwitch (*Uro*^{GS}) was used to allow temporal gene manipulation in adult stages. GeneSwitch is a drug (RU486)-inducible system that enables us to negate the effect of genetic background. To activate the Imd pathway, we overexpressed the constitutively active form of Imd (*imd*^{CA}), whose inhibitory N terminus is removed.[27]

First, we overexpressed *imd*^{CA} using *NP1093-Gal4* (*NP1093->imd*^{CA}) to activate the Imd pathway in the whole organ. Adult female flies showed a typical bloating phenotype, where the flies' abdomen was swelled by excess water (Figure 1B). We quantified body water content by measuring the body weight of each fly before and after drying (referred to as wet weight and dry weight, respectively) and found that Imd activation by *NP1093-Gal4* increased the wet weight compared to its isogenic background control *NP1093->w*^{CS} (Figure 1C). The Imd-activated flies did not show any difference in dry weight, suggesting that Imd activation increased water content without affecting development (Figure 1C). When *imd*^{CA} was overexpressed in the main segment of MTs using *Uro*^{ts} or *Uro*^{GS}, no bloating phenotype was observed. Under this condition, we conducted a Ramsay assay,[28] an *ex vivo* assay to quantify the secretion rate from isolated MTs. We confirmed the decline in fluid secretion rate upon Imd activation (Figure S1C). These data indicate that Imd activation by overexpressing *imd*^{CA} using both *NP1093* and *Uro* decreases tubular secretory function. The function of MTs is to maintain ion and water balance. To test the sensitivity to osmotic stress, we fed the flies high-NaCl food



SUMMARY

An excessive immune response damages organs, yet its molecular mechanism is incompletely understood. Here, we screened a factor mediating organ damage upon genetic activation of the innate immune pathway using *Drosophila* renal tubules. We found that an antimicrobial peptide, Attacin-D (AttD), causes organ damage upon immune deficiency (Imd) pathway activation in the Malpighian tubules. Loss of AttD function suppresses most of the pathological phenotypes induced by Imd activation, such as cell death, bloating of the whole animal, and mortality, without compromising the immune activation. AttD is required for the immune-induced damage specifically in the Malpighian tubules and not the midgut. Unlike other antimicrobial peptides, AttD lacks a signal peptide and stays inside tubular cells, potentially damaging the tubular cells via aggregation and oligomerization. Suppression of AttD almost completely attenuates the pathology caused by a gut-tumor-induced immune activation. Our study elucidates the mechanistic effector of immune-induced organ damage.

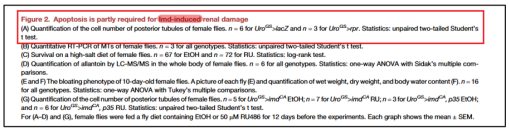


Figure 2. Apoptosis is partly required for Imd-induced renal damage
(A) Quantification of the cell number of posterior tubules of female flies. $n = 6$ for *Uro*^{GS}->lacZ and $n = 3$ for *Uro*^{GS}->*ppv*. Statistics: unpaired two-tailed Student's t test.
(B) Quantitative RT-PCR of MTs of female flies. $n = 3$ for all genotypes. Statistics: unpaired two-tailed Student's t test.
(C) Survival on a high-salt diet of female flies. $n = 67$ for EtOH and $n = 72$ for RU. Statistics: log-rank test.
(D) Quantification of allantoin by LC-MS/MS in the whole body of female flies. $n = 6$ for all genotypes. Statistics: one-way ANOVA with Sidak's multiple comparisons.
(E and F) The bloating phenotype of 10-day-old female flies. A picture of each fly (E) and quantification of wet weight, dry weight, and body water content (F). $n = 16$ for all genotypes. Statistics: one-way ANOVA with Tukey's multiple comparisons.
(G) Quantification of the cell number of posterior tubules of female flies. $n = 5$ for *Uro*^{GS}->*imd*^{CA} EtOH; $n = 7$ for *Uro*^{GS}->*imd*^{CA} RU; $n = 3$ for *Uro*^{GS}->*imd*^{CA}, p35 EtOH; and $n = 6$ for *Uro*^{GS}->*imd*^{CA}, p35 RU. Statistics: unpaired two-tailed Student's t test.
For (A–D) and (G), female flies were fed a fly diet containing EtOH or 50 μM RU486 for 12 days before the experiments. Each graph shows the mean ± SEM.

Model Explanation: ***Language & Expression***

<claim>The paper inconsistently refers to the central pathway/gene as “lmd” even though it is defined elsewhere as “Imd”.</claim>

<evidence>The summary explicitly defines the pathway as Imd, while the Results heading, body text, and Figure 2 caption repeatedly use “lmd,” including the contradictory phrase “lmd ... (imdCA)”.</evidence>

<inference>(1) Because the study’s main manipulated pathway is the immune deficiency pathway, its name must be used consistently. (2) Replacing “Imd” with “lmd” changes the technical symbol and creates ambiguity about the actual molecular target being discussed.</inference>

<judgment>The paper contains a substantive terminology error: the core pathway/gene is inconsistently named as “lmd” versus “Imd,” which obscures the study object.</judgment>

Reason:

The apparent "lmd" wording is an OCR artifact rather than an error in the paper. The original paper consistently uses "Imd" in the Results heading, experimental description, and Figure 2 caption. The OCR likely misrecognized the capital letter "I" in "Imd" as a lowercase "l," creating the spurious form "lmd." Therefore, the alleged terminology inconsistency is caused by OCR noise and is not present in the source document.

C.3.7 Case 7: Missed Annotation



analyzed by a CLSM (Zeiss LSM880, Zeiss, Jena, Germany) and flow cytometry (BD Accuri C6 Plus, BD Biosciences, USA), respectively.

2.9 | Induction of Experimental Autoimmune Uveitis (EAU) and Assessment of Therapeutic Efficacy

All animal experiments comply with the Guide for the Care and Use of Laboratory Animals, Institute of Laboratory Animal Resources. They were approved by the Institutional Animal Care and Use Committee of Wenzhou Medical University (YSG23103019). As reported previously [50], EAU was induced in Lewis rats by the subcutaneous injection of 0.1 mL of emulsion with complete Freund's adjuvant containing IRBP R16 and *Mycobacterium tuberculosis* H37Ra at the left hind footpads. After immunization, the rats were medicated with PBS, blank micelles, 2 mg/kg free CEL, and 1 or 2 mg/kg nanoCEL via an oral administration route. At specific time intervals, the clinical signs of EAU were observed and graded using a slit-lamp biomicroscope (SLM-8E, Chongqing Kang Hua Rui Ming Technology Co., LTD, China) by an experienced ophthalmologist based on the established criteria [51].

At day 11 post-immunization, optical coherence tomography (OCT) was employed to evaluate the impact of various formulations on the retinal architecture and vitreous humor using a Micron IV (Phoenix Research Laboratories, USA). Additionally, the visual function of rats from each group was analyzed by a specialized Ganzfeld dome system (Roland Q400, Wiesbaden, Germany), and the scotopic ERG responses at a light intensity of 3 cd.s/m^2 were recorded.

At day 14 post-immunization, rats from each group were sacrificed, and the eyeballs were harvested. After fixation in FAS eyeball fixative solution (G1109, Servicebio, China) for 24 h at room temperature, the specimens were dehydrated with graded ethanol solution and embedded in paraffin. Thereafter, the paraffins were sectioned (~5 μm of thickness) and stained with hematoxylin and eosin (H&E, Beyotime Biotechnology, China) for histopathological observation. According to the previously established standard [52], the histopathological severity of EAU from each group was scored. Additionally, the intestinal tissue from each group was excised. The harvested eyeball and intestine were embedded in optimal cutting temperature compound and cryo-sectioned (~10 μm of thickness). The sections were blocked with blocking buffer (P0102, Beyotime Biotechnology, China) for 2 h, followed by incubating primary antibodies at 4°C overnight.

At day 14 post-immunization, the rats from each group were sacrificed. The spleen, cervical lymph nodes (CLN), mesenteric lymph nodes (MLN), and retina were excised. The harvested spleen was grinded and lysed with 1 mL of RBC lysis buffer (C3702, Beyotime Biotechnology, China). The excised CLN, MLN, and retina were grinded and homogenized to afford a suspension. The lysate or homogenate was mixed with 3 mL of cell staining buffer (420201, Biolegend, USA) and then centrifuged at 5000 rpm for 5 min. The resulting residue was resuspended in 1 mL of cell staining buffer and filtered through a 300-mesh cell sieve to generate a single-cell suspension. The obtained cell suspension was incubated with Alexa Fluor 700-CD45 (202218, Biolegend, 1:200), PerCP-Cy5.5-CD11b/c (201819, Biolegend, 1:200), Elab Fluor Red 780-CD3 (E-AB-F1228US, Elabscience, 1:200), FITC-CD4 (E-AB-F1105UC, Elabscience, 1:200), PE/Cyanine7-CD8 (E-AB-F1098H, Elabscience, 1:200), BV421-CD25 (565608, BD Bioscience, 1:200), eFluor660-IFN-γ (50-7310-80, Invitrogen, 1:200), BV786-IL-17A (417-7177-82, Invitrogen, 1:80), and eFluor506-Foxp3 (69-5773-82, Invitrogen, 1:40) antibodies at 4°C for 30 min, and then analyzed by a flow cytometer (Attune NxT, Invitrogen, USA).

2.11 | The Assessment of Blood-Retina Barrier (BRB) Integrity

The BRB integrity of EAU rats from each group at day 14 post-immunization was examined by Evans blue perfusion assay and zonula occludens-1 (ZO-1) immunofluorescence assay, respectively. For the Evans blue perfusion assay, EAU rats from each group were intraperitoneally injected with 1 mL of Evans blue (2%) aqueous solution and sacrificed at 2 h post-injection. The excised retinas from each group were flat-mounted and observed by a CLSM (Zeiss LSM880, Zeiss, Jena, Germany). With regard to ZO-1 immunofluorescence assay, the excised retinal pigment epithelium-choroid-sclera complex was flat-mounted and then incubated with a blocking buffer for 2 h at room temperature. Thereafter, the mounted tissue was immersed in primary antibodies of ZO-1 (61-7300, Invitrogen, 1:400) and CD45 (550566, BD Bioscience, 1:200) at 4°C overnight, and then incubated with the secondary antibodies, followed by the visualization using a CLSM (Zeiss LSM880, Zeiss, Jena, Germany).

2.12 | Pharmacokinetics and Biodistribution of nanoCEL After Oral Administration

To measure the plasma pharmacokinetics, the rats were orally administered with either 2 mg/kg of nanoCEL or free CEL. At

To determine whether nanoCEL targeted APCs to ameliorate EAU, we medicated nanoCEL at day 8 post-immunization (Figure S2A). Interestingly, unlike the early phase of medication at day 5, 2 mg/kg of free CEL treatment failed to attenuate clinical signs of EAU, and the severity of EAU was not significantly different from that in the PBS group (Figure S2B). Additionally, 2 mg/kg of nanoCEL treatment partially suppressed the severity of EAU, corresponding to the lower histopathological scores (Figure S2C). Based on these data, it is reasonable to believe that the suppression of EAU by nanoCEL treatment is in a time-dependent manner. Since the primary function of APCs is to initiate and amplify EAU, APCs targeting by nanoCEL during the early phase of disease development (day 5) significantly suppressed the severity of EAU, but exhibited less therapeutic efficacy when medicated at the later phase of disease development (day 8). This result suggested that the function of peripheral macrophages and T cells might contribute to the sustainable intraocular inflammation once APCs were activated at the onset of EAU. We thereafter assessed the populations of APCs and the expression of MHC-II in intestinal tissue. The absolute numbers and populations of CD11b/c$^+$ APCs were greatly elevated in PBS-treated EAU rats compared with normal rats, as evidenced by immunofluorescence and flow cytometry assay. Notably, 2 mg/kg of nanoCEL treatment severely reduced the numbers and populations of CD11b/c$^+$ APCs with a parallel of the reduced level of MHC-II expression (Figure 3F). Thus, oral administration of nanoCEL rather than free CEL at the dosage of 2 mg/kg exhibited the robust inhibitory effect of antigen-presenting capacity.

Th17), consequently attenuating the intraocular inflammation.

3.7 | NanoCEL Protects the Integrity of the Blood-Retinal Barrier (BRB) and RPE Tight Junction in EAU Rats

To visualize the leakage of BRB in rats, the Evans blue dye was injected intravenously via the tail vein (Figure 4A).As shown in Figure 4B, there was almost no dye leakage from retina in normal rats, while apparent dye leakage was observed in the PBS-treated EAU rats, indicating that the disruption of BRB acted a vital role in such leakage. Blank micelles treatment failed to reduce the dye leakage from retina, and 2 mg/kg of free CEL and 2 mg/kg of nanoCEL treatment both resulted in a significant reduction of dye leakage from retina. Of note, nanoCEL treatment had the strongest protective effect of BRB with minimal dye leakage. The disruption of BRB, especially outer BRB, might be ascribed to the disorganization and destroy of RPE tight junction proteins. Comparing with normal rat, the expression of ZO-1 was disrupted and decreased obviously in the RPE layer, accompanied with numerous CD45$^+$ leukocyte infiltrations in PBS-treated EAU rat (Figure 4C,D). Similarly, blank micelles-treated EAU rat displayed a severe destroy of outer BRB and CD45$^+$ leukocyte infiltration in the RPE layer. The treatment of free CEL slightly rescued the BRB with few CD45$^+$ leukocyte infiltrations in the RPE layer. In contrast, nanoCEL treatment almost maintained the integrity of BRB with no CD45$^+$ leukocyte breakthrough in the RPE layer. Based on these data, it is reasonable to believe that oral administration of nanoCEL effectively protects the integrity of BRB and RPE tight junction in EAU rats.

Model Explanation: ***Method Design & Specification***

<claim>The paper gives conflicting administration routes for the Evans blue BRB assay.</claim>

<evidence>The Methods state intraperitoneal injection, while the Results state intravenous tail-vein injection for the same experiment.</evidence>

<inference>(1) Administration route is a fundamental experimental parameter in a vascular leakage assay. (2) An unresolved route contradiction prevents accurate interpretation and reproducibility of the BRB integrity measurement.</inference>

<judgment>The BRB assay is internally inconsistent at the method-specification level.</judgment>

Reason:

The paper contains a genuine internal inconsistency in the Evans blue administration route. The Methods state that Evans blue was administered intraperitoneally, whereas the Results describe the same BRB leakage assay as intravenous injection via the tail vein. These are distinct administration routes and cannot both describe the same procedure as performed. Therefore, regardless of which route was actually used, the manuscript reports the experimental method inconsistently, constituting a clear and objectively verifiable reporting error.