PEERISPECT: Claim Verification in Scientific Peer Reviews

Ali Ghorbanpour
Reviewerly
Toronto, Ontario, Canada

Soroush Sadeghian
Reviewerly
Toronto, Ontario, Canada

Alireza Daghighfarsoodeh
Reviewerly
Toronto, Ontario, Canada

Sajad Ebrahimi
Reviewerly
Toronto, Ontario, Canada

Negar Arabzadeh
Reviewerly, UC Berkeley
Berkeley, California, United States

Seyed Mohammad Hosseini
Reviewerly
Toronto, Ontario, Canada

Ebrahim Bagheri
University of Toronto, Reviewerly
Toronto, Ontario, Canada

Abstract

Peer review is central to scientific publishing, yet reviewers frequently include claims that are subjective, rhetorical, or misaligned with the submitted work. Assessing whether review statements are factual and verifiable is crucial for fairness and accountability. At the scale of modern conferences and journals, manually inspecting the grounding of such claims is infeasible. We present **PEERISPECT**, an interactive system that operationalizes claim-level verification in peer reviews by extracting check-worthy claims from peer reviews, retrieving relevant evidence from the manuscript, and verifying the claims through natural language inference. Results are presented through a visual interface that highlights evidence directly in the paper, enabling rapid inspection and interpretation. PEERISPECT is designed as a modular Information Retrieval (IR) pipeline, supporting alternative retrievers, rerankers, and verifiers, and is intended for use by reviewers, authors, and program committees. We demonstrate PEERISPECT through a live, publicly available demo[1] and API services[2], accompanied by a video tutorial[3].

ACM Reference Format:
Ali Ghorbanpour, Soroush Sadeghian, Alireza Daghighfarsoodeh, Sajad Ebrahimi, Negar Arabzadeh, Seyed Mohammad Hosseini, and Ebrahim Bagheri. 2026. PEERISPECT: Claim Verification in Scientific Peer Reviews. In . ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 Introduction

Peer review remains the cornerstone of scholarly publishing, ensuring that only research meeting standards of novelty, rigor, and clarity enters the scientific record. Despite its importance, peer review is far from flawless [5, 16, 17]. Reviewers frequently include claims that go beyond objective evaluation, sometimes misinterpreting the content of the submission, overstating limitations, or questioning results in ways not fully supported by the manuscript itself [2, 10]. Such issues can compromise the fairness of reviews, mislead editors and program committees, and ultimately slow down the progress of scientific research [7]. While prior work on factuality assessment and claim verification has largely focused on pipelines for general fact-checking or scientific fact verification [15, 18, 20], there has been little effort to address the unique setting of verifying reviewer statements against a manuscript.

A substantial body of work has documented limitations of the peer review process, including bias, inconsistency, and noise [2, 5, 10, 16, 17]. Reviewers may misinterpret experimental settings, overlook details, or overstate shortcomings, leading to claims that are not fully grounded in the paper itself [3, 19]. These issues are particularly consequential because review statements directly influence editorial decisions, acceptance outcomes, and author revisions. Yet, prior works on peer-review analysis have largely focused on systemic properties of the review process and high-level assessments of review quality. Studies such as Lee et al. [10] analyze sources of bias, including reviewer identity and institutional affiliation, while Tennant et al. [17] provide a broad examination of transparency and fairness. Even recent benchmarks like RottenReviews [6] or models for review constructiveness [14] primarily operate at the document or discourse level. They assess overall tone or utility but do not perform the grounded, claim-level verification necessary to identify the specific inconsistencies described above. Addressing this verification gap presents a unique technical challenge. While the information retrieval community has developed mature techniques for factuality assessment, claim verification, and evidence retrieval, most prior work has focused on open domain fact checking or scientific claim verification against large external corpora [15, 18, 20]. In contrast, verifying reviewer claims requires grounding statements against a single submitted document, giving way to a distinct and underexplored retrieval problem.

The need for scalable support has intensified as the volume of scientific submissions continues to grow [1, 11]. Major conferences now receive tens of thousands of papers per cycle. For example, AAAI received approximately 12,000 submissions in 2025, which increased to roughly 23,000 submissions in 2026 [8]. As submission volumes increase, so does the volume of review text, making it infeasible for program committees and editors to manually inspect whether review claims are well grounded. From an information retrieval perspective, this setting naturally raises the question of whether reviewer claims can be treated as information seeking queries over the submitted paper, and whether retrieval and inference techniques can be used to assess their grounding automatically.

In this work, we introduce PEERISPECT, an interactive system that operationalizes claim level verification in peer reviews by framing reviewer statements as queries over the manuscript. The

[1] https://app.reviewer.ly/app/peerispect
[2] https://github.com/Reviewerly-Inc/Peerispect
[3] https://bit.ly/3LMobm8

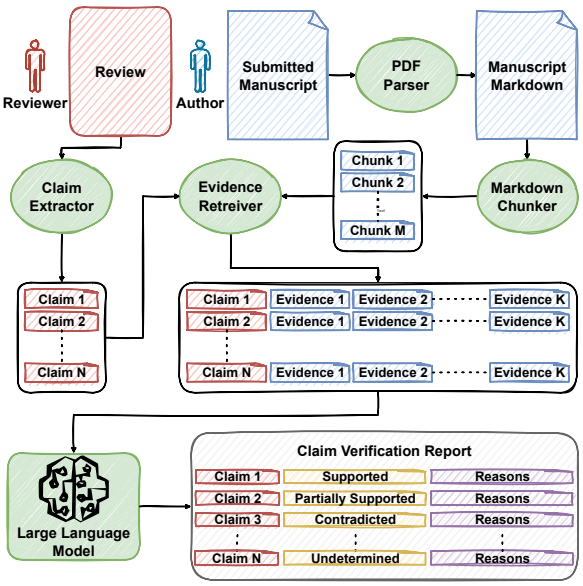

Figure 1: PEERISPECT architecture and processing pipeline. The system consists of four stages: data acquisition, claim extraction, evidence retrieval, and claim verification.

system extracts check worthy claims from reviews, retrieves the most relevant passages from the paper, and verifies alignment via natural language inference. The output indicates whether a claim is supported, partially supported, contradicted, or undetermined, and presents these judgments through a visual interface that highlights evidence directly in the manuscript.

The system supports multiple stakeholders in the peer review process. Reviewers can inspect whether their factual statements are grounded in the manuscript. Authors can identify which review claims are supported or contradicted when preparing rebuttals or revisions. Program committees and editors can use the system as a discretionary aid to maintain review quality at scale. By making claim evidence alignments explicit, PEERISPECT increases transparency while preserving human judgment as the final authority.

Beyond live demonstration, we empirically validate the underlying pipeline using two complementary datasets that reflect both controlled and real world conditions. The `Controlled Manuscript Claims (CMC)` benchmark consists of 500 paper derived claims extracted from 50 ICLR 2024 manuscripts, representing an upper bound scenario in which all claims are supported by construction. The `Real World Review Claims (RRC)` comprises 150 manually annotated reviewer claims from 25 papers. comprising 150 manually annotated reviewer claims from 25 papers. These complementary datasets allow us to rigorously assess evidence retrieval and verification accuracy, ensuring the demo reflects a tested, reliable system rather than a purely illustrative prototype.

Our PEERISPECT tool demonstrates how retrieval, ranking, and inference techniques can be integrated to support accountability and transparency in peer review workflows. By treating review verification as a document grounded information retrieval problem and providing an interactive and extensible system, this work positions peer review analysis as a practical and impactful application area for the IR research community.

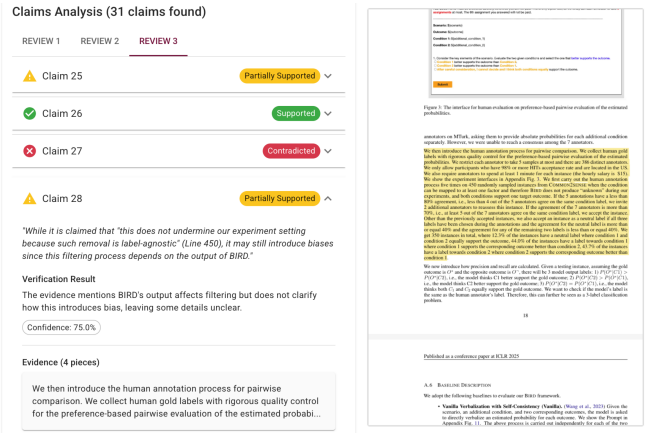

Figure 2: Screenshot of the PEERISPECT interface.

2 Tool Description

PEERISPECT is an interactive system that operationalizes claim level verification in peer reviews through a modular information retrieval pipeline. The system processes a manuscript together with its associated reviews and produces structured annotations that characterize the grounding of individual review claims with respect to the paper. The design emphasizes modularity and extensibility, allowing alternative retrieval, ranking, and verification components to be integrated without altering the overall workflow. Figure 1 presents an overview of the pipeline, while Figure 2 illustrates the user facing interface. The pipeline consists of four conceptual stages: *data ingestion, claim extraction, evidence retrieval*, and *claim verification*. Each stage corresponds to a well defined IR or NLP subtask and can be independently modified or replaced. This design reflects the goal of presenting PEERISPECT as a reusable research artifact rather than a fixed end to end model.

Data ingestion The *data ingestion* stage prepares the manuscript and reviews for downstream processing. Manuscripts are segmented into semantically coherent textual units that serve as retrievable documents, while reviews are normalized into sentence level inputs. Although PEERISPECT supports multiple ingestion modes, including direct document upload and integration with external review platforms such as OpenReview.net, these mechanisms are treated as interchangeable front ends whose primary purpose is to supply clean textual representations to the retrieval pipeline.

Claim extraction. The *claim extraction* stage identifies check worthy factual claims from review text. Peer reviews typically interleave factual observations with subjective judgments and prescriptive feedback. To isolate verifiable content, PEERISPECT decomposes review sentences into candidate spans and filters them using a classifier trained to distinguish factual, evidence seeking statements from opinionated or rhetorical language. This process follows prior observations that early removal of subjective content improves downstream factuality assessment [15]. The remaining spans are normalized into atomic claims by resolving coreference and removing hedging expressions, yielding a set of minimal claims suitable for retrieval based verification.

Evidence retrieval. In the *evidence retrieval* stage, each extracted claim is treated as a query over the manuscript. The paper is indexed as a collection of fixed length passages, and relevant

Table 1: Dataset statistics for the two evaluation benchmarks CMC and RRC.

Dataset	#Papers	#Claims	Description
CMC	50	500	Claims extracted from manuscripts
RRC	25	150	Manually labeled from author Reviewer Interactions

passages are retrieved using a combination of semantic similarity and reranking. In the deployed demo, dense retrieval is performed using `MiniLM-L6-v2` embeddings [13] with FAISS based nearest neighbor search [4]. Retrieved candidates are then reranked using a cross encoder that jointly models the claim and passage [12]. While this configuration is used in the live system, the retrieval stage is explicitly designed to support alternative sparse, dense, or hybrid retrieval strategies.

Claim Verification. The final stage performs claim verification by assessing the relationship between each claim and its retrieved evidence. Following standard formulations in claim verification and natural language inference [18, 20], the claim is treated as a hypothesis and the evidence passages as premises. An LLM based verifier assigns one of four labels indicating support, partial support, contradiction, or uncertainty. These labels provide a structured characterization of claim grounding that can be inspected by users and consumed by downstream analysis.

PEERISPECT output. The output of PEERISPECT is presented through an interactive interface that aligns claims, evidence, and verification outcomes. Claims are listed alongside their labels, and corresponding evidence passages are highlighted directly within the manuscript view. This design enables users to quickly inspect retrieval and verification behavior, making the system suitable for practical use. PEERISPECT exposes claim verification in peer review as a document grounded information retrieval problem and provides an extensible platform for studying this task in realistic review settings.

3 Implementation Details

PEERISPECT boasts a web interface that sits over its service-oriented architecture. The design emphasizes modularity, scalability to support interactive use in real peer review scenarios. The backend, implemented in FastAPI, orchestrates data flow across modules, exposes a REST API, and manages communication with external services such as OpenReview.net. The frontend, built in React with react-pdf-highlighter, renders PDF documents and highlights evidence passages for each claim returned by the API. We integrate the VLLM inference engine [9], which enables us to easily set up local LLM installations that are both efficient and scalable, while maintaining data privacy by avoiding reliance on external APIs. For the live demo, we use `Qwen-2.5-7B` [21], but the API supports integration with any model available through VLLM. To ensure portability and reproducibility, PEERISPECT is packaged using Docker, with each service (backend, frontend, and model server) running in its own container to simplify deployment and resource management. This containerized setup allows PEERISPECT to be deployed on local servers for development or scaled to cloud environments for broader accessibility.

4 Empirical Validation

For PEERISPECT to be useful in real peer-review workflows, its behavior must be predictable and trustworthy. Rather than aiming for a comprehensive research evaluation, our goal in this section is to provide practical evidence that the system behaves reliably under conditions that resemble actual conference and journal reviewing. To this end, we built two complementary benchmarks from real ICLR 2024 submissions and their OpenReview discussions and used them to assess PEERISPECT.

4.1 Benchmarks Derived from Real Peer Review

All data is drawn from publicly available OpenReview entries for ICLR 2024. We sampled 50 submissions, stratified across accepted (oral), accepted (poster), and rejected papers, and collected their PDFs, metadata, and full review-rebuttal threads. Table 1 summarizes the resulting corpus.

4.1.1 Controlled Manuscript Claims Benchmark (`CMC`). To understand how PEERISPECT behaves in an idealized setting where every claim is guaranteed to be grounded in the manuscript, we constructed the Controlled Manuscript Claims benchmark (`CMC`). Using `OpenAI-o4-mini`, we decomposed each manuscript into atomic factual units and randomly sampled 10 claims per paper, yielding 500 instances in total. Because each claim is taken directly from the manuscript, every instance receives a gold label of *Supported*.

This controlled setup serves two purposes. First, it allows us to check whether the retrieval component reliably surfaces the paragraphs that originally expressed each claim. Second, it lets us observe how the verifier behaves when it receives either oracle passages or system-retrieved passages as evidence. In practice, `CMC` acts as an upper-bound, low-ambiguity scenario where any errors can be attributed to retrieval or verification behavior rather than to unclear ground truth.

4.1.2 Real-World Review Claims Benchmark (`RRC`). Real peer reviews are less tidy. Reviewer comments may mix interpretations, paraphrases, and partially grounded critiques, and they often get clarified only through back-and-forth discussion with the authors. To capture this setting, we built the Real-World Review Claims benchmark (`RRC`) by randomly selecting 25 papers from the same 50 OpenReview submissions. We applied `OpenAI-o4-mini` to convert reviewer comments into atomic claims, and appended the author response thread as context. From which, we sampled 150 claim-dialog pairs and manually assigned one of the following four labels:

- **Supported:** when the author explicitly agrees with the reviewer or cites the manuscript in a way that confirms the claim.
- **Contradicted:** when the author disputes the claim and refers to manuscript-grounded evidence.
- **Partially Supported:** when the author qualifies the claim (e.g., "only under condition X" or "only for subset Y").
- **Undetermined:** when the dialog does not resolve the issue or centers on subjective or policy-level statements.

Here, both the formulation of the claim and the relevance of evidence are less clear-cut. Thus we evaluate the full pipeline and examine whether PEERISPECT assigns labels that align with the human annotations. `RRC` is meant to reflect the conditions under which a reviewer, author, or editor would actually use the tool.

Table 2: Performance comparison across CMC and RRC benchmarks with different retrievers and LLMs.

Model	Retriever	CMC		RRC	
		ACC	Recall	ACC	Recall
Qwen-2.5-3b	BM25	0.728	**0.486**	0.220	–
	Dense Retriever	0.618	0.418	0.193	–
	Dense + Reranker	0.740	0.476	**0.247**	–
	Sparse + Reranker	**0.744**	0.478	**0.247**	–
Qwen-2.5-7b	BM25	**0.905**	**0.486**	0.247	–
	Dense Retriever	0.804	0.418	0.247	–
	Dense + Reranker	0.880	0.469	**0.287**	–
	Sparse + Reranker	0.896	0.478	0.260	–

4.2 Evaluating Different Configurations

We use the two benchmarks in complementary ways. CMC is used to probe individual components, while RRC is used to understand end-to-end behavior under noisy, discourse-driven conditions. On CMC, we vary retrieval strategies and evidence selection to see how design choices affect the tool's ability to recover the original manuscript passages and assign the expected *Supported* label. On RRC, we focus on the configurations that remain robust when claims are paraphrased, partially grounded, or indirectly linked to the text.

Retrievers. We compare three practical retrieval configurations that we considered for deployment in the demo: a sparse retriever (BM25), a dense retriever, and a sparse retriever followed by a cross-encoder reranker. We also examine the behavior of the cross-encoder when applied directly to the top-20 candidates from the sparse + reranker setup.

LLMs. For verification, we experimented with `Qwen-2.5-3B` and `Qwen-2.5-7B` within an otherwise identical pipeline to understand how model size affects stability. For all configurations, the verifier receives the top-3 passages per claim as evidence. We report accuracy and recall on both oracle and system-retrieved evidence (Table 2), not as a leaderboard, but to give a concrete sense of how dependable the system is under different design choices. The configuration used in the live demo is chosen based on this analysis.

5 Findings and Observations

The two benchmarks introduced in Section 4.1 allow us to observe how PEERISPECT behaves in settings that tool users are likely to encounter in practice. CMC represents a controlled, low-ambiguity regime where all claims are supported by construction. RRC captures the more challenging reality of noisy reviewer language and partially grounded comments. Below we summarize the patterns that matter most for users of the demo.

Observations on CMC. Because every claim in CMC originates from the manuscript itself, a well-behaved system should retrieve the corresponding passage and mark the claim as supported. Consistent with this expectation, accuracy on CMC is substantially higher than on RRC across all configurations (Table 2). We see two clear trends:

- Larger verification models (`Qwen-2.5-7B`) yield noticeably higher accuracy than smaller ones (`Qwen-2.5-3B`), suggesting that additional capacity helps with nuanced NLI judgments once relevant evidence is available.
- Sparse retrieval with BM25 performs very strongly in this setting. Since claims are extracted from the manuscript, their wording tends to be lexically close to the source passages, and BM25 achieves high recall and accuracy.

Because CMC also includes an oracle-evidence condition, it acts as a sanity check. Any remaining errors point to either retrieval misses or verifier misclassifications rather than to ambiguous labels. This gives users confidence that, when claims are directly tied to the text, PEERISPECT behaves predictably.

Observations on RRC. RRC is deliberately closer to how reviewers actually write. Claims can be vague, compressed, or only partially grounded in the manuscript, and there is no single "correct" evidence span to retrieve. Here, accuracy naturally drops relative to CMC, since the system must both interpret the claim and find useful evidence without oracle guidance. Furthermore, unlike CMC, there is *no defined oracle evidence span*, and thus retrieval recall cannot be reported. Two observations are particularly relevant:

- Overall accuracy is lower than on CMC, which is expected given the ambiguity in real reviewer text. This reflects the inherent difficulty of the task rather than a failure of the tool.
- The dense retriever followed by a reranker provides the most reliable configuration in this setting (ACC 0.287 for `Qwen-2.5-7B`). Dense embeddings and reranking help bridge paraphrases and conceptual shifts that are common in reviewer phrasing, whereas purely sparse retrieval struggles when wording diverges from the manuscript.

The CMC and RRC results provide practical reassurance that the color-coded outputs shown in the interface of PEERISPECT correspond to a tested and empirically grounded pipeline. Users can therefore treat PEERISPECT not as a black box, but as a system whose behavior has been observed under both *idealized* and *realistic* review conditions.

6 Concluding Remarks

As conferences and journals continue to grow, it becomes increasingly difficult to ensure that peer reviews remain grounded and fair. Reviewers may misstate what a paper contains or overlook details, and these factual claims can influence decisions, rebuttals, and revisions. This creates a timely need for transparent methods that connect review statements to evidence in the submitted manuscript. PEERISPECT addresses this need by treating reviewer claims as document grounded retrieval and verification queries. The system extracts check worthy claims from reviews, retrieves the most relevant passages from the paper, and verifies them via an NLI style formulation. Results are presented in an interactive interface that highlights evidence directly in the manuscript, enabling rapid inspection of claim evidence alignment.

At the conference, we will demonstrate its end to end use on realistic review scenarios, including interactive exploration of evidence highlights and the behavior of the retrieval and verification pipeline. We will also showcase the modular design that allows alternative retrievers, rerankers, and verifiers to be substituted within the same framework. By providing an open, tested, and extensible system for claim level review verification, PEERISPECT offers a practical playground for the IR community to explore future research on retrieval and inference applied to scholarly communication.

References

[1] Ariful Azad and Afeefa Banu. 2024. Publication trends in artificial intelligence conferences: The rise of super prolific authors. *arXiv preprint arXiv:2412.07793* (2024).

[2] Kirsten Bell, Patricia Kingori, and David Mills. 2024. Scholarly publishing, boundary processes, and the problem of fake peer reviews. *Science, Technology, & Human Values* 49, 1 (2024), 78–104.

[3] Pradeep Dasigi, Kyle Lo, Iz Beltagy, Arman Cohan, Noah A. Smith, and Matt Gardner. 2021. A Dataset of Information-Seeking Questions and Answers Anchored in Research Papers. (2021). arXiv:2105.03011 [cs.CL] https://arxiv.org/abs/2105.03011

[4] Matthijs Douze, Alexandr Guzhva, Chengqi Deng, Jeff Johnson, Gergely Szilvasy, Pierre-Emmanuel Mazaré, Maria Lomeli, Lucas Hosseini, and Hervé Jégou. 2024. The Faiss library. (2024). arXiv:2401.08281 [cs.LG]

[5] John A Drozdz and Michael R Ladomery. 2024. The peer review process: past, present, and future. *British Journal of Biomedical Science* 81 (2024), 12054.

[6] Sajad Ebrahimi, Soroush Sadeghian, Ali Ghorbanpour, Negar Arabzadeh, Sara Salamat, Muhan Li, Hai Son Le, Mahdi Bashari, and Ebrahim Bagheri. 2025. RottenReviews: Benchmarking Review Quality with Human and LLM-Based Judgments. In *Proceedings of the 34th ACM International Conference on Information and Knowledge Management (CIKM '25)*. 5642–5649. doi:10.1145/3746252.3761506

[7] Prashant Garg. 2020. Problems in peer review. *Journal of Clinical and Diagnostic Research* (2020).

[8] Odest Chadwicke Jenkins and Matthew E. Taylor. 2025. AAAI-26 Review Process Update: Scale, Integrity Measures, and Experimental Use of AI-Assisted Reviewing.

[9] Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient Memory Management for Large Language Model Serving with PagedAttention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*.

[10] Carole J Lee, Cassidy R Sugimoto, Guo Zhang, and Blaise Cronin. 2013. Bias in peer review. *Journal of the American Society for information Science and Technology* 64, 1 (2013), 2–17.

[11] Seth S Leopold. 2015. Increased manuscript submissions prompt journals to make hard choices. *Clinical Orthopaedics and Related Research®* (2015).

[12] Rodrigo Nogueira and Kyunghyun Cho. 2020. Passage Re-ranking with BERT. arXiv:1901.04085 [cs.IR] https://arxiv.org/abs/1901.04085

[13] Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics. https://arxiv.org/abs/1908.10084

[14] Abdelrahman Sadallah, Tim Baumgärtner, Iryna Gurevych, and Ted Briscoe. 2025. The good, the bad and the constructive: Automatically measuring peer review's utility for authors. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*. 28979–29009.

[15] Alessandro Scirè, Karim Ghonim, and Roberto Navigli. 2024. FENICE: Factuality evaluation of summarization based on natural language inference and claim extraction. *arXiv preprint arXiv:2403.02270* (2024).

[16] Richard Smith. 2006. Peer review: a flawed process at the heart of science and journals. *Journal of the royal society of medicine* 99, 4 (2006), 178–182.

[17] Jonathan P Tennant, Jonathan M Dugan, Daniel Graziotin, Damien C Jacques, François Waldner, Daniel Mietchen, Yehia Elkhatib, Lauren B Collister, Christina K Pikas, Tom Crick, et al. 2017. A multi-disciplinary perspective on emergent and future innovations in peer review. *F1000Research* (2017).

[18] James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. 2018. FEVER: a large-scale dataset for fact extraction and VERification. (2018).

[19] David Wadden, Shanchuan Lin, Kyle Lo, Lucy Lu Wang, Madeleine van Zuylen, Arman Cohan, and Hannaneh Hajishirzi. 2020. SciFact: A Benchmark for Fact Checking in Scientific Writing. In *Proceedings of EMNLP*.

[20] Theodora Worledge, Tatsunori Hashimoto, and Carlos Guestrin. 2024. The extractive-abstractive spectrum: Uncovering verifiability trade-offs in llm generations. *arXiv preprint arXiv:2411.17375* (2024).

[21] An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388* (2025).