Article

Synthesizing scientific literature with retrieval-augmented language models

https://doi.org/10.1038/s41586-025-10072-4

Received: 16 February 2025

Accepted: 17 December 2025

Published online: 04 February 2026

Open access

Check for updates

Akari Asai[1,2], Jacqueline He[1,7], Rulin Shao[1,7], Weijia Shi[1], Amanpreet Singh[2], Joseph Chee Chang[2], Kyle Lo[2], Luca Soldaini[2], Sergey Feldman[2], Mike D'Arcy[2], David Wadden[2], Matt Latzke[2], Jenna Sparks[2], Jena D. Hwang[2], Varsha Kishore[1,2], Minyang Tian[3], Pan Ji[4], Shengyan Liu[3], Hao Tong[3], Bohao Wu[3], Yanyu Xiong[5], Luke Zettlemoyer[1], Graham Neubig[6], Daniel S. Weld[1,2], Doug Downey[2], Wen-tau Yih[1], Pang Wei Koh[1,2] & Hannaneh Hajishirzi[1,2] ✉

Scientific progress depends on the ability of researchers to synthesize the growing body of literature. Can large language models (LLMs) assist scientists in this task? Here we introduce OpenScholar, a specialized retrieval-augmented language model (LM)[1] that answers scientific queries by identifying relevant passages from 45 million open-access papers and synthesizing citation-backed responses. To evaluate OpenScholar, we develop ScholarQABench, the first large-scale multi-domain benchmark for literature search, comprising 2,967 expert-written queries and 208 long-form answers across computer science, physics, neuroscience and biomedicine. Despite being a smaller open model, OpenScholar-8B outperforms GPT-4o by 6.1% and PaperQA2 by 5.5% in correctness on a challenging multi-paper synthesis task from the new ScholarQABench. Although GPT-4o hallucinates citations 78–90% of the time, OpenScholar achieves citation accuracy on par with human experts. OpenScholar's data store, retriever and self-feedback inference loop improve off-the-shelf LMs: for instance, OpenScholar-GPT-4o improves the correctness of GPT-4o by 12%. In human evaluations, experts preferred OpenScholar-8B and OpenScholar-GPT-4o responses over expert-written ones 51% and 70% of the time, respectively, compared with 32% for GPT-4o. We open-source all artefacts, including our code, models, data store, datasets and a public demo.

Synthesizing knowledge from the scientific literature is essential for discovering new directions, refining methodologies and supporting evidence-based decisions, yet the rapid growth of publications makes it increasingly difficult for researchers to stay informed. Effective synthesis requires precise retrieval, accurate attribution and access to up-to-date literature. LLMs can assist but suffer from hallucinations[2,3], outdated pre-training data[4] and limited attribution. In our experiments, GPT-4o fabricated citations in 78–90% of cases when asked to cite recent literature across fields such as computer science and biomedicine.

Retrieval-augmented LMs[5–7] mitigate some of these issues by incorporating external knowledge at inference time and have encouraged systems for literature search and synthesis[8–10]. However, most rely on black-box application programming interfaces (APIs) or general-purpose LMs and lack open, domain-specific retrieval data stores (processed corpora with retrieval indices) tailored to scientific domains. Evaluations for literature synthesis are also limited, typically focusing on narrow, single-discipline studies[8,9] or simplified tasks such as multiple-choice question answering[10].

To address the challenges of accurate, comprehensive and transparent scientific literature synthesis, we introduce OpenScholar (Fig. 1, top), to our knowledge the first fully open, retrieval-augmented LM specifically designed for scientific research tasks. OpenScholar integrates a domain-specialized data store (OpenScholar DataStore, OSDS), adaptive retrieval modules and a new self-feedback-guided generation mechanism that enables iterative refinement of long-form outputs. OSDS is a fully open, up-to-date corpus of 45 million scientific papers and 236 million passage embeddings, offering a reproducible foundation for training and inference. OpenScholar retrieves from OSDS using trained retrievers and rerankers, generates cited responses and iteratively refines them by means of a self-feedback loop to improve factuality, coverage and citation accuracy. This same pipeline is used to generate high-quality synthetic data, enabling the training of a compact 8B model (OpenScholar-8B) and retrievers without relying on proprietary LMs.

To evaluate OpenScholar, we introduce ScholarQABench (Fig. 1, middle), to our knowledge the first multidisciplinary benchmark for open-ended scientific synthesis. Unlike previous benchmarks focused on short-form outputs, multiple-choice formats or domain reasoning tasks[10–12], ScholarQABench requires long-form responses grounded in up-to-date literature from numerous papers. It includes 3,000 research

[1]University of Washington, Seattle, WA, USA. [2]Allen Institute for AI, Seattle, WA, USA. [3]University of Illinois Urbana-Champaign, Urbana, IL, USA. [4]University of North Carolina at Chapel Hill, Chapel Hill, NC, USA. [5]Stanford University, Stanford, CA, USA. [6]Carnegie Mellon University, Pittsburgh, PA, USA. [7]These authors contributed equally: Jacqueline He, Rulin Shao. ✉e-mail: hannaneh@cs.washington.edu

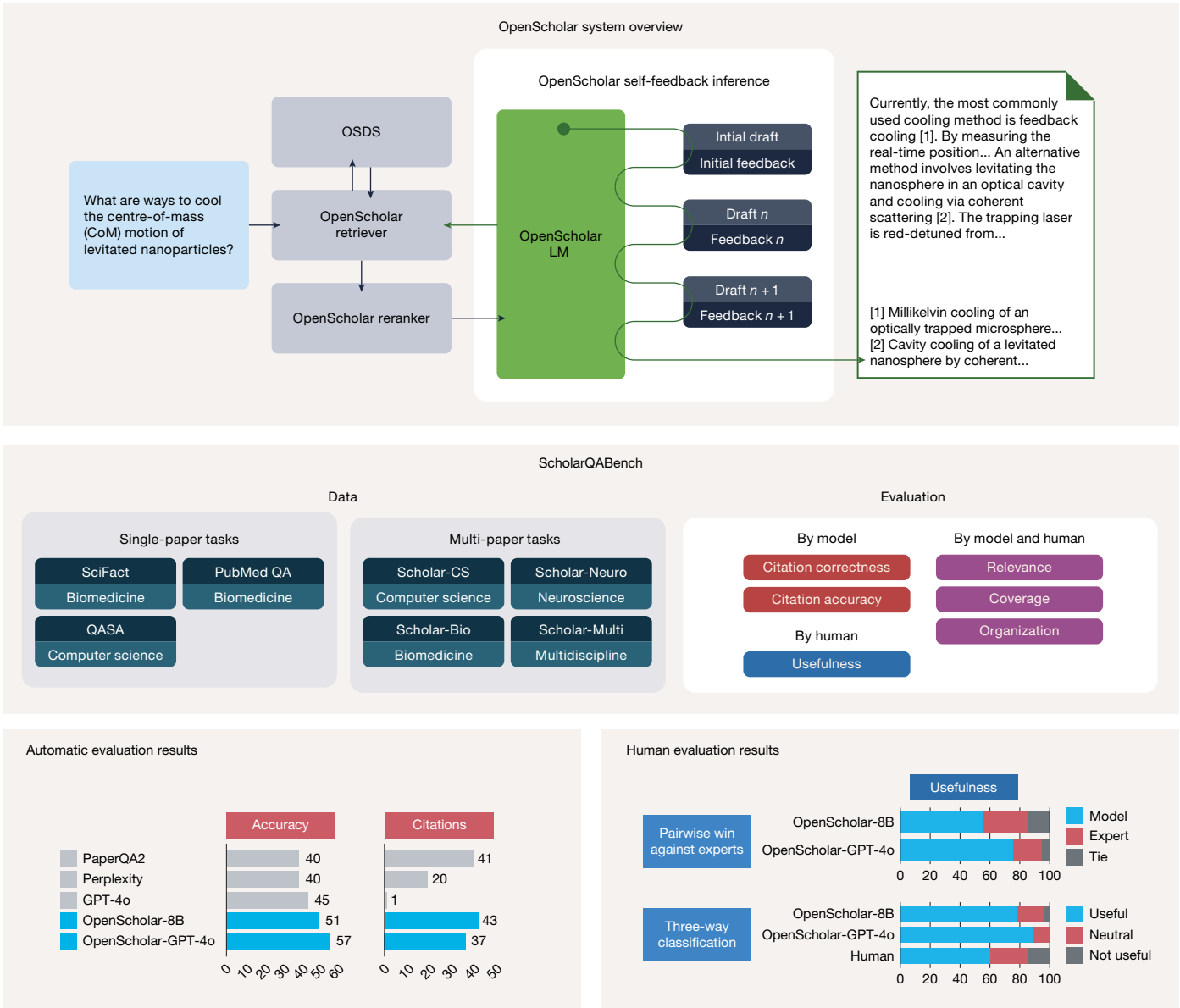


Fig. 1 | Overview of OpenScholar, ScholarQABench and evaluation results. Top, overview of OpenScholar. OpenScholar consists of a specialized data store (OSDS), retrievers and LMs and iteratively improves responses using self-feedback inference with retrieval. Middle, overview of ScholarQABench. ScholarQABench consists of 2,200 expert-written questions across several scientific disciplines and we introduce automatic and human evaluation protocols for ScholarQABench. Bottom, automatic and human evaluation results: experimental results from the ScholarQABench computer science subset (Scholar-CS, 100 questions) show that OpenScholar with our trained 8B or GPT-4o substantially outperforms other systems and is preferred over experts more than 50% of the time in human evaluations. Our human evaluations were conducted by 16 experts with PhDs across 108 questions from Scholar-Multi.

questions and 250 expert-written answers across computer science, physics, biomedicine and neuroscience, authored by experienced PhD students and postdocs to reflect real-world literature review practices. To overcome the difficulties of evaluating long-form, comprehensive responses[13–16], ScholarQABench introduces a rigorous evaluation protocol combining automatic metrics (for example, citation accuracy) with human rubric-based assessments of coverage, coherence, writing quality and factual correctness to enable reliable assessments of the detailed long-form answers of LMs. Our expert analysis shows that the proposed multifaceted evaluation pipeline achieves high agreement with expert judgements, reliably capturing coverage, coherence, writing quality and factual correctness in long-form scientific answers.

We evaluated proprietary and open models (for example, GPT-4o, Llama 3.1 8B and 70B) with and without retrieval capabilities, as well as specialized systems such as PaperQA2 (ref. 10), on ScholarQABench. Although GPT-4o demonstrated strong general performance, it struggled with citation accuracy and coverage, often producing inaccurate or non-existent citations. OpenScholar outperformed both LM-only and retrieval-augmented pipelines, surpassing proprietary and open-source systems. Notably, using fully open-source checkpoints, OpenScholar-8B outperformed PaperQA2, built on proprietary LMs, and production systems such as Perplexity Pro, achieving 6% and 10% improvements, respectively. Furthermore, OpenScholar's use of smaller, efficient retrievers substantially reduced costs. The OpenScholar pipeline can also enhance off-the-shelf LMs. For example, when using GPT-4o as the underlying model, OpenScholar-GPT-4o achieves a 12% improvement in correctness compared with GPT-4o alone. Furthermore, although expert human performance exceeds that of GPT-4o and other competitive baselines, OpenScholar systems match or surpass expert humans in both answer correctness and citation accuracy. Our extensive evaluations demonstrate the importance of the core components of OpenScholar, including reranking, self-feedback and

Table 1 | Results of ScholarQABench

	Single-paper performance						Multi-paper performance						Cost
	Pub		Sci		QASA		CS		Multi		Bio	Neu	CS
Model	Acc.	Cite	Acc.	Cite	Acc.	Cite	Rub.	Cite	LLM	Cite	Cite	Cite	USD per question
Llama 3.1 8B	61.5	0	66.8	0	14.3	0	41.9	0	3.79	0	0	0	0.0001
+RAG^{OSDS}	75.2	63.9	75.5	36.2	18.6	47.2	46.7	26.1	4.22	25.3	38.0	36.8	0.0001
OpenScholar-8B	76.4	68.9	76.0	43.6	23.0	56.3	51.1	47.9	4.12	42.8	50.8	56.8	0.003
Llama 3.1 70B	69.5	0	76.9	0	13.7	0	44.9	0	3.82	0	0	0	0.0004
+RAG^{OSDS}	77.4	71.1	78.2	42.5	22.7	63.6	48.5	24.5	4.24	41.4	53.8	58.1	0.0004
OpenScholar-70B	79.6	74.0	82.1	47.5	23.4	64.2	52.5	45.9	4.03	54.7	55.9	63.1	0.01
GPT-4o	65.8	0	77.8	0	21.2	0	45.0	0.1	4.16	0.7	0.2	0.1	0.006
+RAG^{OSDS}	75.1	73.7	79.3	47.9	18.3	53.6	52.4	31.1	4.03	31.5	36.3	21.9	0.01
OpenScholar-GPT-4o	74.8	77.1	81.3	56.5	18.7	60.4	57.7	39.5	4.51	37.5	51.5	43.5	0.05
PaperQA2	–	–	–	–	–	–	45.6	48.0	3.82	47.2	56.7	56.0	0.3–2.3
Perplexity	–	–	–	–	–	–	40.0	–	4.15	–	–	–	0.002*

Pub, Sci and QASA indicate the three single-paper tasks, PubMedQA[41], SciFact[42] and QASA[43]. CS, Multi, Bio and Neu indicate Scholar-CS (computer science), Scholar-Multi, Scholar-Bio (biomedicine) and Scholar-Neuro (neuroscience), which require multi-paper synthesis and long-form answer generations, respectively. 'Acc.' indicates the correctness metrics in single-paper tasks (accuracy for PubMedQA and SciFact, ROUGE-L for QASA). Rubric accuracy ('Rub.') in Scholar-CS is used as the primary metric for correctness on multi-paper synthesis tasks. 'Cite' indicates citation F1. 'LLM' indicates the average score of organization, relevance and coverage as predicted by Prometheus[44]. PaperQA2 is based on GPT-4o and its pricing is dependent on the number of PDF files used during inference. For the 8B and 70B model costs, although evaluations were conducted on our local machines, we estimated costs based on Together AI pricing. *We used Perplexity Pro (which requires a monthly subscription at US$20) and divided this cost by 9,000, which is the maximum number of queries allowed under the Pro subscription. Because the Perplexity user interface does not provide snippets for each citation, we were unable to evaluate its citation accuracy.

verification, as well as the value of combining diverse retrieval pipelines and training domain-specialized retrieval systems.

As well as automatic evaluations on ScholarQABench, we conducted detailed expert assessments with 16 scientists from fields such as computer science, physics and biomedicine. These experts performed pairwise and fine-grained evaluations of the outputs of OpenScholar against 108 expert-written responses to literature synthesis queries in ScholarQABench. OpenScholar, when paired with GPT-4o and our trained 8B model, consistently outperformed expert-written responses, with win rates of 70% and 51%, respectively. By contrast, vanilla GPT-4o (that is, without retrieval) struggled with information coverage and was rated as less helpful than human experts, achieving only a 31% win rate against human responses. Overall, these findings demonstrate that OpenScholar can produce high-quality outputs that are not only on par with expert-written answers but, in some cases, above par, particularly in terms of coverage and organization. We also released the first public demo for scientific literature synthesis, powered by OpenScholar-8B. Since launch, the demo has been used by more than 30,000 users and has collected nearly 90,000 user queries across diverse scientific fields.

OpenScholar performance on ScholarQABench

We first provide an overview of our key results of OpenScholar on our newly created expert-annotated benchmark, ScholarQABench. Table 1 shows scores for several aspects of the main baselines.

Baseline models

We compare three settings. (1) Parametric LMs (no retrieval): Llama 3.1 8B/70B (ref. 17) and GPT-4o (gpt-4o-2024-05-13 (ref. 18)) generate answers and a list of paper titles. We verify that the titles exist and, when they do, fetch their abstracts as citations. (2) Retrieval-augmented generation (RAG) baselines: using our OSDS (RAG^{OSDS}), we retrieve the top N passages and concatenate them with the input, following standard RAG pipelines[2,18]. (3) Our method (OpenScholar): a custom inference pipeline with a trained 8B model (OpenScholar-8B) and with Llama 3.1 70B and GPT-4o back ends (OpenScholar-70B, OpenScholar-GPT-4o). For multi-paper tasks, we also test Perplexity Pro. We use the paid subscription version; because there is no API, we collect final predictions through selenium, and cannot extract citations, and PaperQA2 (ref. 10). As the data store of PaperQA2 is not public, we use OSDS as its retrieval source.

Main results

On single-paper tasks, OpenScholar consistently outperforms other models. OpenScholar-8B and OpenScholar-70B outperform Llama 3.1 8B and 70B with and without retrieval augmentation in terms of final accuracy and citation accuracy (Table 1). OpenScholar-70B even matches or outperforms GPT-4o on PubMedQA and QASA. We also found that OpenScholar models consistently show substantial improvements in terms of citation accuracy compared with standard RAG baselines (RAG^{OSDS}).

In multi-paper tasks, we report the Scholar-CS rubric score—the number of expert-annotated answer rubrics satisfied by the response of a model (see Methods for scoring details)—as our primary measure of correctness. We also evaluate overall writing quality with a LLM judge ('LLM') on Scholar-Multi and track citation accuracy across all datasets. OpenScholar-8B, OpenScholar-70B and OpenScholar-GPT-4o, which use the OpenScholar pipeline with our fine-tuned Llama 3.1 8B-based LM and off-the-shelf Llama 3.1 70B and GPT-4o as the generator LM, respectively, demonstrate strong performance. Specifically, OpenScholar-GPT-4o provides a 12.7-point improvement over GPT-4o alone in the Scholar-CS rubric score and a 5.3 improvement over standard RAG. When combined with trained OpenScholar-8B, OpenScholar greatly outperforms the pipeline that uses off-the-shelf Llama 3.1 8B, showcasing the benefits of domain-specific training. Furthermore, OpenScholar-8B shows better rubric performance by a substantial margin than proprietary systems such as GPT-4o, Perplexity Pro or PaperQA2, which use GPT-4o models for passage reranking, summarization and answer generation. Although we found that PaperQA2 matches or even outperforms OpenScholar in citation accuracy, its responses often rely on only one or a few papers, summarizing each retrieved snippet individually. This leads to limited coverage and contributes to its lower performance on the Scholar-CS rubric and LLM judge scores. These findings highlight the importance of balancing both precision and recall in effective literature synthesis. Notably, by making use of efficient retrieval pipelines with lightweight

Table 2 | Statistics of hallucinated papers in the computer science and biomedicine domains

	Computer science			Biomedicine		
Model	**Total no.**	**No. of hallucinated (↓)**	**Ratio (↓)**	**Total no.**	**No. of hallucinated (↓)**	**Ratio (↓)**
OpenScholar-8B	9.65	0	0	6.25	0	0
Llama 3.1 8B	5.20	4.79	92.1%	5.58	5.46	97.6%
Llama 3.1 70B	6.14	4.78	78.1%	6.98	6.74	96.6%
GPT-4o	5.74	4.52	78.7%	5.24	4.97	94.8%

Our analysis revealed a substantial number of non-existent cited papers in predictions made by LLMs without retrieval, which is a problem not observed in OpenScholar.

bi-encoders, cross-encoders and in-house models, OpenScholar-8B and OpenScholar-GPT-4o achieve much lower costs—orders of magnitude cheaper than PaperQA2—while maintaining high performance.

Limitations of parametric LMs

On both single-paper and multi-paper tasks, we observe that non-retrieval augmented baselines struggle—retrieval is almost always conducive to achieving better performance—and models without any retrieval often struggle to generate correct citations and show limited coverage on multi-paper tasks. Table 2 presents statistics on the cited papers in the outputs of four models. We report the number of fully fabricated citations ('No. of hallucinated') by verifying whether the cited paper titles exist using the Semantic Scholar API. Across models, the share of cited papers that actually exist is very low: despite plausible-looking reference lists, 78–98% of titles are fabricated, with the worst rates in biomedicine. This mirrors previous findings that LLMs hallucinate on long-tail, underrepresented knowledge[2,19] and we suggest that the effect is amplified in scientific areas undercovered on the open web. Repeating the analysis on GPT-5, which was released in August 2025, lowers title-level hallucination to 39% but fabricated citations remain common. Examples of model responses, as well as a list of paper titles, are available in Supplementary Tables 19 and 20. We also noticed that, even when citations refer to real papers, most of them are not substantiated by the corresponding abstracts, resulting in near-zero citation accuracy.

We also observe that such models generate responses with limited coverage. On Scholar-Multi, non-retrieval models (Llama 3.1 8B, 70B and GPT-4o) consistently exhibit much lower average scores compared with retrieval-augmented models. This discrepancy is primarily driven by substantially lower coverage scores; for instance, Llama 3.1 8B achieves a coverage score of 3.45, whereas Llama 3.1 8B + OSDS (a standard RAG baseline) improves the coverage score to 4.01. These results suggest that relying on the parametric knowledge of models alone is particularly difficult in scientific domains, especially for smaller LMs.

Human performance on ScholarQABench

We also analysed expert performance on this challenging literature synthesis task. Specifically, we evaluated human-written answers on the two subsets of ScholarQABench with long-form human annotations: Scholar-CS and Scholar-Multi. For both, we applied the same evaluation pipeline used for model-generated responses to assess rubric and citation accuracy. For Scholar-Multi, rubric evaluation is not available, but we conducted expert evaluations on both human and model responses and compare the results in the next section. Table 3 compares human performance with OpenScholar-GPT-4o, OpenScholar-8B, PaperQA2 and GPT-4o (no retrieval). Our analysis shows that human-written answers remain strong baselines for quality and relevance. On rubric-based evaluations, human responses outperform GPT-4o without retrieval by 9.6 points and OpenScholar-8B by 2.9 points. PaperQA2 demonstrates high citation accuracy but its scores for rubrics, organization, coverage, and relevance are lower. By contrast, OpenScholar-GPT-4o achieves even higher rubric scores than human experts and OpenScholar-8B matches expert-level citation accuracy. We found that OpenScholar tends to produce more comprehensive responses than humans or other baseline systems, citing a greater number of papers, as reflected in both answer length and citation count. In Supplementary Information Section 6, we present a detailed human analysis of model-written and human-written answers and further examine key factors for improving scientific literature synthesis.

Ablations and analysis

Ablations of inference components. We ablate inference components by removing: (1) reranking (use top *N* OSDS results only); (2) feedback (generate once then attribute); and (3) citation verification (omit the final check). For OpenScholar-8B, we also ablate training by swapping in off-the-shelf Llama 3.1 8B with the same inference pipeline (as in OpenScholar-GPT-4o). Extended Data Table 2 shows notable drops in both correctness and citation accuracy for all removals, with the largest losses from removing reranking. Feedback removal hurts GPT-4o more than our trained 8B (probably because the latter learned feedback patterns during training) and skipping post-hoc attribution reduces both citation accuracy and final correctness. The gap between trained versus vanilla OpenScholar-8B underscores the value of domain-specific training.

Ablations of retrieval. We also compare OSDS-only (dense retrieval), S2-only (Semantic Scholar keyword API), web-only (You.com) and their combination. To isolate retrieval, we use our 8B LM without self-feedback or citation verification and rerank to the top 15 with OpenScholar reranker. On Scholar-CS (Extended Data Table 2), web-only performs worst (45.9 correctness, 12.6 citation F1), S2-only improves especially on citations (47.9/39.1) and the combined pipeline is best (49.6/47.6). Tailored, literature-focused retrieval (dense + API + reranking) yields the strongest factuality and attribution.

We analyse how the number of retrieved passages (top *N*) affects performance. We compare standard RAG and OpenScholar with our trained 8B model and Llama 3.1 8B, evaluating generation and citation accuracy on Scholar-CS. Extended Data Figs. 3 and 4 summarize the results. Although Llama 3.1 is trained to accept up to 128,000 tokens, its performance degrades beyond a certain context size: increasing top *N* from 5 to 10 improves correctness but larger *N* harms both correctness and citation accuracy. This suggests that, despite long-context capacity, smaller LMs may struggle to effectively use many passages without specialized training. By contrast, our trained 8B model remains strong up to $N = 20$ and larger models (for example, Llama 3.1 70B) are more robust to longer contexts.

Expert evaluation of OpenScholar's effectiveness

To complement automatic metrics and examine the strengths and limits of OpenScholar, we ran expert evaluations comparing human-written answers with those generated by LLM systems. This study involved more than 100 literature review questions and more than 15 participants, including PhD students, research scientists and university professors with expertise in the relevant fields. In total, we curated more than 400 fine-grained expert evaluations of expert and model answers.

Table 3 | Expert-written answer stats

	Scholar-CS			Scholar-Multi						
	Length	No. of cit.	Rubric	Prec.	Rec.	Prec.	Rec.	Org.	Cov.	Rel.
Human	424.3	7.1	54.0	43.2	40.1	44.4	41.5	–	–	–
OpenScholar-GPT-4o	1,447.3	11.8	57.7	41.1	38.1	38.0	37.1	4.63	4.50	4.23
OpenScholar-8B	706.3	10.1	51.1	40.6	44.1	42.5	43.2	3.92	4.44	4.02
PaperQA2	288.7	2.7	45.6	45.9	48.0	53.1	42.2	3.67	3.46	4.20
GPT-4o	281.5	0.5	45.0	1.0	0.8	0.8	0.6	4.06	3.94	4.21

Models tend to generate longer responses and cite more papers than humans. For reference, we include GPT-4o outputs without retrieval, evaluated on the human evaluation queries. 'Length' refers to the average length of the answers and 'No. of cit.' indicates the average number of cited papers in the final answers. 'Prec.' and 'Rec.' denote citation precision and recall and 'Org.', 'Cov.' and 'Rel.' represent organization, coverage and relevance, respectively, as evaluated by LLM-as-a-judge. For the number of citations of GPT-4o, we only consider valid citations.

Evaluation design

We use 108 question-answer (QA) pairs from Scholar-Multi, written by experts (expert writers). We evaluated three set-ups on these questions: GPT-4o (no external retrieval), OpenScholar with GPT-4o as the generator (OpenScholar-GPT-4o) and OpenScholar with our trained 8B model (OpenScholar-8B), each producing answers with citations. We then recruited a separate group of PhD-level domain experts to rate the model-generated answers against expert-written answers.

In particular, each evaluation involves presenting a question, a model-generated answer and a human-written answer. Expert raters then conduct fine-grained assessments of each answer and provide pairwise preference judgements between the two. For fine-grained evaluations, we use the five-scale evaluation criteria described in Methods (coverage, relevance and organization), with annotators scoring both model and human answers using the same rubrics. Detailed prompts are presented in Supplementary Information Section 6. For usefulness, annotators assign scores on a scale from 1 to 5, which we convert into three classes: not useful (1, 2), neutral (3) and useful (4, 5). We then calculate the percentage of answers that fall into the useful category. For pairwise preference, annotators either choose one of the answers or mark a 'tie' if they judge both answers to be of equal quality. Optionally, experts provide explanations on why one answer is better than the other.

Details of expert writers. Our expert writers for question and answer writing are 12 PhD students and postdoctoral researchers from research institutions across the USA, all of whom have at least three years of research experience and have published several papers in journals or conferences in their fields. The expert areas covered by our writers include the computer science (natural language processing, computer vision, humancomputer interaction), physics (astrophysics, photonics/optics) and biomedical (neuroscience, bioimaging) domains and we assign our expert annotators to questions in their expertise. On average, we paid US$35–40 per person.

Details of expert raters. Sixteen expert raters from the three fields contributed to our evaluations, with 12 of them also participating in answer generation. All expert raters meet the same qualifications as those who composed the answers. To minimize potential biases, we ensured that raters did not evaluate responses to their own questions by assigning evaluation tasks to different groups of experts. Each instance was reviewed by one to three expert raters, depending on availability. The inter-annotator agreement was 0.68 using pairwise comparison with ties and 0.70 using a relaxed approach, in which ties were merged. On average, each expert rater spent five minutes per instance on evaluation and received compensation ranging from US$25 to US$35.

Expert evaluation results

Overall result. Table 4 presents the average scores for each evaluation aspect, alongside the relative win rates against human responses. Extended Data Fig. 5 illustrates the score distributions for human, GPT-4o and OpenScholar with Llama 3.1 8B and GPT-4o. Notably, both OpenScholar-GPT-4o and our OpenScholar-8B versions outperform human answers in more than 50% of cases, with their advantage primarily attributed to their ability to provide a greater breadth and depth of information (coverage). By contrast, GPT-4o, which lacks retrieval capabilities, demonstrates greatly limited coverage and wins in fewer than 35% of cases, with its overall usefulness rated much lower than responses from humans and the other two models. These results highlight that, even for state-of-the-art models, synthesizing and answering scientific literature review questions remains a challenging task, consistent with our findings on ScholarQABench. Overall, OpenScholar-GPT-4o and OpenScholar-8B are rated as useful in 80% and 72% of the queries, respectively.

Although the performance of OpenScholar using an open 8B LM already surpasses that of human experts, the output of the 8B model is judged to be less organized or fluent than the present state-of-the-art private LLM-based OpenScholar. We found that GPT-4o incorporates feedback more effectively and tends to generate longer and more fluent outputs, leading to much higher organization scores compared with both the OpenScholar-8B as well as human responses.

Effects of length control on model responses. Although we found that model outputs are often preferred over expert-written outputs, one potential confounding factor is the large difference in their output length—OpenScholar-GPT-4o and OpenScholar-8B are 2.4 times and 2.0 times longer than expert-written answers, respectively, which affects judgement[20]. To understand the effect of output length, we conducted a controlled experiment. For a random sample of 50 questions, we generate abbreviated responses for OpenScholar-GPT-4o by prompting GPT-4o to create summaries of responses that are less than 300 words. This led to OpenScholar answers that average around 333 words, which is close to the average length of human answers. We then repeat the human evaluation, considering both fine-grained and overall responses. On average, the shortened GPT-4o scores 4.5 for organization, 4.6 for coverage and 4.6 for relevance. The shortened OpenScholar-GPT-4o responses are preferred or tied with expert answers in 75% of the queries. The experimental results show that the superior performance of the model is not only because of the increased length of the OpenScholar answers. Moreover, the explanations of human annotators often mention that both shortened OpenScholar and human answers could be improved by incorporating more details, implying that a 300-word restriction may limit the usefulness of answers.

Analyses on human explanations for pairwise judgements. We randomly sampled 59 instances with free-form explanations of pairwise preferences and conducted a manual analysis to identify factors that influence overall preferences. Specifically, we examined whether the explanations referenced one or more of the following four categories: organization, relevance, coverage and citations. Although the first three categories align with the fine-grained human evaluation criteria, the citation category also considers the quality of the cited

Table 4 | Expert rater evaluations of literature synthesis responses written by expert writers and LMs

	Fine-grained (1–5, average)			Overall usefulness (%)	Relative to human (%)		
	Organization	Coverage	Relevance		Win	Tie	Lose
GPT-4o	4.63 (+0.4)	4.06 (−0.2)	4.50 (−0.1)	69.7 (−13.9)	31.9	13.8	54.2
OpenScholar-8B	3.82 (−0.3)	4.30 (+0.7)	4.00 (−0.4)	72.1 (+8.7)	50.8	12.3	36.9
OpenScholar-GPT-4o	4.47 (+0.8)	4.38 (+0.9)	4.30 (0)	80.0 (+22.5)	70.0	6.8	23.2

Fine-grained aspect evaluations are conducted on a five-point scale across four aspects, following our detailed instructions and rubrics. Values in parentheses denote relative performance differences; positive values indicate that the model achieves higher average performance, whereas negative values indicate that expert writers achieve higher average performance.

papers (for example, whether the system includes seminal papers in the field). Our analysis (Supplementary Information Table 27) revealed that 12%, 23%, 29% and 9% of the explanations cited organization, relevance, coverage and citation accuracy, respectively, as key factors in pairwise decisions. This suggests that coverage plays a crucial role in how humans assess the quality of responses, with annotators largely favouring model-generated answers for their greater coverage and depth of information. However, annotators also noted that the citations provided by models could be improved, pointing out that the suggested papers were occasionally outdated or less relevant compared with more representative work.

Discussion

To further research on LM-based systems that can help scientists navigate the complex, ever-growing task of scientific literature review, we introduce OpenScholar and ScholarQABench. OpenScholar, the first fully open retrieval-augmented system, uses open-weight LLMs and trained retrieval models to iteratively refine scientific output, addressing challenges such as hallucinations and citation accuracy. ScholarQABench, a new large-scale benchmark, provides a standardized way to evaluate literature review automation across several scientific domains. In evaluations using ScholarQABench, OpenScholar demonstrates substantial improvements, outperforming existing systems such as GPT-4o and the concurrent proprietary system PaperQA2. Our expert evaluation across three scientific disciplines reveals that OpenScholar generates answers that are more helpful than those produced by expert annotators, who required an hour per annotation. Specifically, OpenScholar, using our trained 8B and GPT-4o achieves a 51% and 70% win rate against human-generated answers, respectively. We open-source the OpenScholar code, data, model checkpoints, data stores and ScholarQABench, along with a public demo, to support and accelerate future research efforts. Our public demo has engaged more than 30,000 users across diverse scientific disciplines. Future work can further improve OpenScholar by integrating user feedback from this platform to enhance retrieval quality, improve citation accuracy and optimize overall usability.

Limitations

We highlight several limitations of our work in this section. It is important to note that we do not claim that LM-based systems can fully automate scientific literature synthesis. To further advance research in this area, we are releasing both ScholarQABench and OpenScholar to the community.

Limitations of ScholarQABench

First, expert annotation is costly and time-consuming, so our human-written evaluation sets are small (for example, 110 for computer science long-form question answering; 108 expert answers), which may introduce variance and annotator-expertise bias. We open-source data and annotation pipelines to facilitate scaling.

Second, our automatic evaluation may not perfectly capture quality. In Scholar-CS, we combine length, excerpts and rubric items with heuristic weights. Annotators often requested ancillary elements (background, elaborations, challenges) that are not strictly required and LLMs tend to supply these, potentially inflating scores or enabling exploitation of rubric style. Despite good correlations with expert judgements, scoring emphases and aggregation merit refinement. Our citation precision/recall is sentence-level and can be overly strict when adjacent sentences carry support. Annotations reflect specific time points (July 2024 for Scholar-CS, September 2024 for Scholar-Multi); for fair comparison, papers published after these dates should be excluded. We recommend using the OSDS or restricting sources to publications up to October 2024 for ScholarQABench v1 and we plan regular updates.

Third, ScholarQABench is a static, public benchmark, raising future contamination risks. Although multi-paper synthesis data were newly written by experts, public availability may expose it during training or search[21,22]. We will continue updating the benchmark and monitoring its use.

Last, ScholarQABench primarily focuses on computer science, biomedicine and physics, with no instances from social sciences or other engineering and scientific disciplines. We recognize that our findings may not fully generalize to other domains, particularly those with more restricted access to paper data.

Limitations of OpenScholar

Although OpenScholar demonstrates strong performance on ScholarQABench and in human evaluations, as discussed in the relevant sections, our expert annotators identified several limitations.

First, as highlighted by our expert annotators, OpenScholar does not consistently retrieve the most representative or relevant papers for certain queries. Enhancing retrieval methodologies by incorporating further information, such as citation networks or metadata such as publication recency, could substantially improve its performance. OpenScholar outputs may contain factual inaccuracies or unsupported information, particularly in versions based on our 8B model, which has limited capacity for instruction-following and scientific knowledge. Future work can explore training that further improves OpenScholar-8B. In parallel, although competitive, OpenScholar-GPT-4o relies on invoking the proprietary GPT-4o through the OpenAI API, which may evolve over time, making exact result replication a challenge. Furthermore, note that OpenScholar does not use license-protected papers at inference time. There are continuing discussions on how to ensure fair data use in retrieval-augmented LMs and we leave the exploration of properly incorporating copyright-protected content to future work.

We encourage future research to address these limitations and continue improving LM-based systems for scientific literature review.

Limitations of our expert evaluation process. In our human evaluations, annotators performed fine-grained assessments on aspects such as coverage, relevance, organization and usefulness, whereas other factors, such as citation precision and recall, were separately evaluated. As a result, when assessing usefulness or pairwise preferences, annotators may have focused more on the overall quality of writing instead of carefully evaluating factual correctness or citation accuracy.

We leave a more detailed human analysis of citation accuracy, validity and factuality for future work.

Our evaluations were conducted by 16 PhD students and postdoctoral professionals and we made an effort to align their expertise with the evaluated topics. However, because research often necessitates deep domain knowledge, the annotators may not have captured more nuanced differences for questions outside their immediate areas of expertise. Furthermore, these evaluations were based on 108 questions that span three scientific disciplines, meaning that findings may not fully generalize to other fields or domains.

Online content

Any methods, additional references, Nature Portfolio reporting summaries, source data, extended data, supplementary information, acknowledgements, peer review information; details of author contributions and competing interests; and statements of data and code availability are available at https://doi.org/10.1038/s41586-025-10072-4.

1. Asai, A., Min, S., Zhong, Z. & Chen, D. Retrieval-based language models and applications. In *Proc. 61st Annual Meeting of the Association for Computational Linguistics (Volume 6: Tutorial Abstracts)* (eds Chen, Y.-N., Mieskes, M. & Reddy, S.) 41–46 (ACL, 2023).
2. Mallen, A. et al. When not to trust language models: investigating effectiveness of parametric and non-parametric memories. In *Proc. 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (eds Rogers, A., Boyd-Graber, J. & Okazaki, N.) 9802–9822 (ACL, 2023).
3. Mishra, A. et al. Fine-grained hallucination detection and editing for language models. In *Proc. 1st Conf. Language Modeling* (Philadelphia, 2024; https://colmweb.org).
4. Kasai, J. et al. RealTime QA: what's the answer right now? In *Proc. 37th Int. Conf. Neural Information Processing Systems (NeurIPS '23)* (eds Oh, A. et al.) 49025–49043 (Curran Associates, 2023).
5. Lewis, P. et al. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Proc. 34th Int. Conf. Neural Information Processing Systems (NIPS '20)* (eds Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M. F. & Lin, H.) 9459–9474 (Curran Associates, 2020).
6. Guu, K., Lee, K., Tung, Z., Pasupat, P. & Chang, M. Retrieval augmented language model pre-training. In *Proc. 37th International Conference on Machine Learning* (eds Daumé, H. & Singh, A.) 3929–3938 (PMLR, 2020).
7. Asai, A., Wu, Z., Wang, Y., Sil, A. & Hajishirzi, H. Self-RAG: learning to retrieve, generate, and critique through self-reflection. In *Proc. 12th Int. Conf. Learning Representations* (eds Kim. B et al.) (2024).
8. Agarwal, S., Laradji, I.H., Charlin, L. & Pal, C. Litllm: a toolkit for scientific literature review. Preprint at https://arxiv.org/abs/2402.01788v1 (2024).
9. Zheng, Y. et al. OpenResearcher: unleashing AI for accelerated scientific research. In *Proc. 2024 Conf. Empirical Methods in Natural Language Processing: System Demonstrations* (eds Hernández Farías, D. I., Hope, T. & Li, M.) 209–218 (Association for Computational Linguistics, 2024).
10. Skarlinski, M. D. et al. Language agents achieve superhuman synthesis of scientific knowledge. Preprint at https://arxiv.org/abs/2409.13740 (2024).
11. Phan, L. et al. Humanity's last exam. Preprint at https://arxiv.org/abs/2501.14249 (2025).
12. Wang, X. et al. Scibench: evaluating college-level scientific problem-solving abilities of large language models. In *Proc. 41st Int. Conf. Machine Learning* (eds Salakhutdinov, R. et al.) 50622–50649 (PMLR, 2024).
13. Xu, F. et al. KIWI: a dataset of knowledge-intensive writing instructions for answering research questions. In *Findings of the Association for Computational Linguistics: ACL 2024* (eds Ku, L.-W., Martins, A. & Srikumar, V.) 12969–12990 (ACL, 2024).
14. Xu, F., Song, Y., Iyyer, M. & Choi, E. A critical evaluation of evaluations for long-form question answering. In *Proc. 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (eds Rogers, A., Boyd-Graber, J. & Okazaki, N.) 3225–3245 (ACL, 2023).
15. Kamoda, G., Asai, A., Brassard, A. & Sakaguchi, K. Quantifying the influence of evaluation aspects on long-form response assessment. In *Proc. 31st Int. Conf. Computational Linguistics* (eds Rambow, O. et al.) 8787–8808 (ACL, 2025).
16. Krishna, K., Roy, A. & Iyyer, M. Hurdles to progress in long-form question answering. In *Proc. 2021 Conf. North American Chapter of the Association for Computational Linguistics: Human Language Technologies* (eds Toutanova, K. et al.) 4940–4957 (ACL, 2021).
17. Dubey, A. et al. The Llama 3 herd of models. Preprint at https://arxiv.org/abs/2407.21783v1 (2024).
18. Hurst, A. et al. Gpt-4o System Card. Preprint at https://arxiv.org/abs/2410.21276 (2024).
19. Kandpal, N., Deng, H., Roberts, A., Wallace, E. & Raffel, C. Large language models struggle to learn long-tail knowledge. In *Proc. 40th International Conference on Machine Learning* (eds Krause, A. et al.) 15696–15707 (PMLR, 2023).
20. Dubois, Y., Galambosi, B., Liang, P. & Hashimoto, T. B. Length-controlled alpacaeval: a simple way to debias automatic evaluators. In *Proc. 1st Conf. Language Modeling* (Philadelphia, 2025; https://colmweb.org).
21. Wu, M. et al. Reasoning or memorization? Unreliable results of reinforcement learning due to data contamination. In *Proc. 40th Annual AAAI Conference on Artificial Intelligence* (2026).
22. Jiang, M. et al. Investigating data contamination for pre-training language models. In *Proc. 2024 Conf. North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)* (eds Duh, K., Gomez, H. & Bethard, S.) 8706–8719 (ACL, 2024).
23. Shao, R. et al. Scaling retrieval-based language models with a trillion-token datastore. In *Proc. 38th Int. Conf. Neural Information Processing Systems (NIPS '24)* (eds Globerson, A. et al.) 91260–91299 (Curran Associates, 2024).
24. Soldaini, L. et al. Dolma: an open corpus of three trillion tokens for language model pretraining research. In *Proc. 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (eds Ku, L.-W., Martins, A. & Srikumar, V.) 15725–15788 (ACL, 2024).
25. Lo, K., Wang, L. L., Neumann, M., Kinney, R. & Weld, D. S2ORC: the semantic scholar open research corpus. In *Proc. 58th Annual Meeting of the Association for Computational Linguistics* (eds Jurafsky, D. Chai, J., Schluter, N. & Tetreault, J.) 4969–4983 (ACL, 2020).
26. Kinney, R. M. et al. The Semantic Scholar open data platform. Preprint at https://arxiv.org/abs/2301.10140v1 (2023).
27. Karpukhin, V. et al. Dense passage retrieval for open-domain question answering. In *Proc. 2020 Conf. Empirical Methods in Natural Language Processing (EMNLP 2020)* 6769–6781 (ACL, 2020).
28. Thakur, N., Reimers, N., Rücklé, A., Srivastava, A. & Gurevych, I. BEIR: a heterogeneous benchmark for zero-shot evaluation of information retrieval models. In *Proc. Neural Information Processing Systems Track on Datasets and Benchmarks 1 (NeurIPS Datasets and Benchmarks 2021)* (eds Vanschoren, J. & Yeung, S.) (Curran Associates, 2021).
29. Izacard, G. et al. Unsupervised dense information retrieval with contrastive learning. *Trans. Machine Learning Res.* https://api.semanticscholar.org/CorpusID:249097975 (2022).
30. Asai, A. et al. Task-aware retrieval with instructions. In *Findings of the Association for Computational Linguistics: ACL 2023* (eds Rogers, A., Boyd-Graber, J. & Okazaki, N.) 3650–3675 (ACL, 2023).
31. Liu, N. F. et al. Lost in the middle: how language models use long contexts. *Trans. Assoc. Comput. Linguist.* **12**, 157–173 (2024).
32. Xu, F., Shi, W. & Choi, E. RECOMP: improving retrieval-augmented LMs with compression and selective augmentation. In *Proc. 12th Int. Conf. Learning Representations* (Vienna, 2024; https://iclr.cc/Conferences/2024).
33. Nogueira, R. & Cho, K. Passage re-ranking with BERT. Preprint at https://arxiv.org/abs/1901.04085v1 (2019).
34. Xiao, S. et al. C-Pack: packed resources for general Chinese embeddings. In *Proc. 47th Int. ACM SIGIR Conf. Research and Development in Information Retrieval (SIGIR'24)* 641–649 (ACM, 2024).
35. Ram, O. et al. In-context retrieval-augmented language models. *Trans. Assoc. Comput. Linguist.* **11**, 1316–1331 (2023).
36. Liu, N. F., Zhang, T. & Liang, P. Evaluating verifiability in generative search engines. In *Findings of the Association for Computational Linguistics: EMNLP 2023* (eds Bouamor, H., Pino, J. & Bali, K.) 7001–7025 (ACL, 2023).
37. Jiang, Z. et al. Active retrieval augmented generation. In *Proc. 2023 Conf. Empirical Methods in Natural Language Processing* (eds Bouamor, H., Pino, J. & Bali, K.) 7969–7992 (ACL, 2023).
38. Wadden, D. et al. SciRIFF: a resource to enhance language model instruction-following over scientific literature. In *Proc. 2025 Conf. Empirical Methods in Natural Language Processing* (eds Christodoulopoulos, C. Chakraborty, T. Rose, C. & Peng, V.) 6083–6120 (ACL, 2024).
39. Li, M. et al. Superfiltering: weak-to-strong data filtering for fast instruction-tuning. In *Proc. 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (eds Ku, L.-W., Martins, A. & Srikumar, V.) 14255–14273 (ACL, 2024).
40. Ivison, H. et al. Camels in a changing climate: enhancing LM adaptation with Tulu 2. Preprint at https://arxiv.org/abs/2311.10702 (2023).
41. Jin, Q., Dhingra, B., Liu, Z., Cohen, W. & Lu, X. PubMedQA: a dataset for biomedical research question answering. In *Proc. 2019 Conf. Empirical Methods in Natural Language Processing and the 9th Int. Joint Conf. Natural Language Processing (EMNLP-IJCNLP)* (eds Inui, K., Jiang, J., Ng, V. & Wan, X.) 2567–2577 (ACL, 2019).
42. Wadden, D. et al. Fact or fiction: verifying scientific claims. In *Proc. 2020 Conf. Empirical Methods in Natural Language Processing (EMNLP)* (eds Webber, B., Cohn, T., He, Y. & Liu, Y.) 7534–7550 (ACL, 2020).
43. Lee, Y. et al. QASA: advanced question answering on scientific articles. In *Proc. 40th Int. Conf. Machine Learning* (eds Krause, A. et al.) 19036–19052 (PMLR, 2023).
44. Kim, S. et al. PROMETHEUS: inducing fine-grained evaluation capability in language models. In *Proc. 12th International Conference on Learning Representations* (eds Kim. B et al.) (2024).

Methods

OpenScholar

OpenScholar (detailed in Extended Data Fig. 1) is a new retrieval-augmented LM designed to ensure reliable, high-quality responses to a range of information-seeking queries about scientific literature.

Task formulation and challenges. Given a scientific query x, the task is to identify relevant papers, synthesize their findings and generate a response y that effectively addresses the query. This response should be accompanied by a set of citations, $\mathbf{C} = c_1, c_2, \ldots, c_K$, in which each citation c_i corresponds to an existing scientific paper. Each c_i in $\mathbf{C}$ corresponds to specific passages from scientific literature and should be provided as an in-line citation, linked to the relevant spans of text in y, following standard practice in scientific writing. These citations allow researchers to trace the output back to the original literature, ensuring transparency and verifiability.

However, this task presents several challenges: (1) retrieving high-recall, high-precision scientific content from a vast, domain-specific corpus; (2) synthesizing accurate, non-hallucinated responses grounded in the retrieved evidence; and (3) producing citation-aware outputs that align generated text with appropriate references at a fine-grained level. A further challenge lies in the scarcity of resources: to our knowledge, there is limited availability of large-scale, up-to-date scientific corpora, especially those suitable for dense retrieval, as well as a lack of supervised training data for both retrieval and generation in scientific domains.

Overview of OpenScholar. To address these challenges, OpenScholar introduces several key innovations that extend the standard RAG (refs. 1,5) model for scientific literature synthesis. Specifically, OpenScholar combines domain-specialized retrieval, citation-aware generation and a new self-feedback inference mechanism, all built on top of a fully open and large-scale scientific data store.

Formally, OpenScholar consists of three key components: a data store $\mathbf{D}$, a retriever R and a generator LM G. In standard retrieval-augmented inference pipelines, the process begins with R, which retrieves a set of passages $\mathbf{P} = \{p_1, p_2, \ldots, p_N\}$ from $\mathbf{D}$—a large-scale corpus of previously published scientific papers—based on semantic relevance to the input query x. These passages serve as context for the next step. The generator LM G then takes both the retrieved passages $\mathbf{P}$ and the input query x to produce the output y along with corresponding citations $\mathbf{C}$. Formally, this process can be represented as:

$$y, \mathbf{C} = G(x, R(x, \mathbf{D})),$$

in which each c_i in $\mathbf{C}$ corresponds to a specific passage from $\mathbf{P}$.

OpenScholar introduces new technical contributions to address the aforementioned challenges. (1) To address the lack of large-scale, up-to-date scientific corpora, we construct OSDS, a database of 45 million scientific papers with precomputed dense embeddings, representing, to our knowledge, the largest and most up to date scientific paper data store available. (2) To enable high-recall, high-precision retrieval and support LM training in scientific domains, we design a retrieval pipeline that integrates both our trained OpenScholar retriever and OpenScholar reranker, optimized on scientific data to select the top N passages for the generator G—and complementary retrieval APIs—ensuring broader coverage and improved relevance. (3) To improve factuality and evidence grounding, we introduce iterative self-feedback inference with retrieval and citation verification, in which the LM first produces an initial draft y_0 with G and then iteratively refines it using retrieval-augmented self-feedback. (4) To enhance citation accuracy and overall output quality, we use this inference pipeline to generate high-quality training data, enabling the training of specialized LMs that produce more accurate and citation-aware long-form answers.

OpenScholar retrieval pipeline. Extended Data Fig. 1 (top left) shows our retrieval pipeline, consisting of a data store $\mathbf{D}$, a bi-encoder retriever θ_{bi} and a cross-encoder reranker θ_{cross}. We first select initial candidate paragraphs using $\mathbf{D}$ and θ_{bi}, as well as external APIs, and then refine and identify the top N relevant paragraphs using θ_{cross}.

Scientific paper collection and data store construction. Although previous work often used a small subset of scientific papers, such as arXiv papers from 2023 to 2024 (ref. 9), it is important to have a diverse set of papers to improve the quality and coverage of model generation[23]. For this, we use peS2o (ref. 24) as our retrieval source, which consists of open-access academic papers from S2ORC (ref. 25). We built our data store using peS2o v3, which includes 45 million papers up to October 2024. For evaluations, we use peS2o v2, which consists of papers up to January 2023, as our main benchmarks and models were constructed before the curation of peS2o v3. Our data store, which we call OSDS, consists of 236 million passages. To our knowledge, this is the largest open-sourced data store for scientific literature.

Initial paragraph retrieval. We retrieve passages from three sources: (1) the OSDS using our trained retriever; (2) publicly available abstracts from papers returned through the Semantic Scholar API (ref. 26) based on search keywords; and (3) publicly available texts from papers retrieved through a web search engine using the original query x.

For (1), we first generate embeddings of each passage in the OSDS $\mathbf{D}$ using the passage bi-encoder θ_{bi}, which processes text chunks (for example, queries or passages) into dense vectors[27] offline. Off-the-shelf retrieval models often struggle in out-of-domain scenarios[28]. To overcome this limitation, we develop θ_{bi} by continually pre-training Contriever[29] on the peS2o data store in an unsupervised fashion to improve domain-specific retrieval performance. During inference, we encode the query using θ_{bi} and retrieve the top 70 passages through a nearest-neighbour search[27]. Following previous work[23], we split the main text of each paper into discrete, 256-word text blocks (as determined by white space) and concatenate the paper title to each block to formulate passages in $\mathbf{D}$. Although semantic segmentation can be used to split scientific articles into meaningful sections, we found that not all papers in our data store consistently retain such semantic or discourse structures. Furthermore, applying segmentation models post hoc would be computationally expensive at this scale. Therefore, following common practice in this area[27,29], we divide articles into fixed-length chunks to ensure scalability and simplicity. Therefore, several text chunks from the same paper can be retrieved at inference time.

For (2), we first generate keywords from the query x using a generator LM. These keywords are then used to retrieve the top 10 papers for each, as ranked by citation count, through the Semantic Scholar search API. This approach addresses a limitation of the Semantic Scholar API, which cannot effectively handle long, question-like search queries. If the full text is available in HTML format (for example, ar5iv), we retrieve the entire text and include all passages from the paper as candidate documents. Otherwise, we only consider the abstract.

For (3), we obtain the top 10 search results using the You.com retrieval API, restricting the search to academic platforms such as arXiv and PubMed. Similarly to (2), if the papers are open access, we extract and add their full texts to the candidate pool; otherwise, we include only their abstracts.

Top N paragraph reranking and finalization. After the initial stage, we have gathered more than a hundred or even a thousand relevant passages per query. However, passages retrieved by the bi-encoder may include unhelpful context owing to deep interactions between a query and passages, as they are encoded separately[30]. Feeding a large number of documents that might include irrelevant content to LLMs can cause efficiency and performance issues, even with state-of-the-art models[31,32]. To overcome these challenges, we use a cross-encoder reranker[33,34], denoted as θ_{cross}. For each candidate paragraph, the cross-encoder reranker jointly encodes and computes the relevance score between the input query and each of the passages. We then use

the relevance score to rank the passages accordingly. To train θ_{cross} for scientific domains, we fine-tune a BGE reranker[34] using synthetic data generated by Llama-3-70B-Instruct. Specifically, we randomly generate queries based on abstracts from peS2o and retrieve the top 10 passages. For each passage, Llama-3-70B-Instruct assigns a relevance score from 1 to 5, for which we consider scores of 4 or 5 as positive and scores of 1 or 2 as negative. Passages with a score of 3 are discarded. More details of θ_{cross} training are in Supplementary Information Section 3.3. During reranking and finalization of the top N passages, we also implement extra meta-filtering, which includes: (1) limiting the number of passages per paper to three passages and (2) incorporating normalized citation counts into relevance scores predicted by the cross-encoder.

Inference: self-reflective iterative RAG. In standard RAG (refs. 5,35), a generator LM takes in the original input x and top N retrieved passages **P** and generates the output y_0. Although effective for tasks such as question answering[2], this one-step generation can lead to unsupported claims[36] or incomplete output owing to missing information[7,37]. To address these challenges, in OpenScholar, we introduce an iterative generation approach with self-feedback, which involves three steps: (1) initial response and feedback generation to output the initial draft y_0 and a set of feedback on y_0; (2) iterative refinement with further retrieval to improve y_0 using the feedback; and (3) citation verification. Our inference is detailed in Extended Data Fig. 1, top right.

Initial response and feedback generation. Given the input x and retrieved passages **P**, the generator LM first produces an initial response y_0 with citation markers tied to the corresponding passages in **P**. After generating y_0, the LM generates a set of feedback on y_0, $\mathbf{F} = f_1, f_2, \ldots, f_T$, that is aimed at improving the initial response, in which each feedback f_t is a natural language sentence that describes potential improvements. Although the model can generate an arbitrary number of feedback (T), we set a maximum limit of three feedback sentences for efficient inference. Unlike previous work that relies on a predefined set of feedback signals[7], our approach allows the LM to generate flexible natural language feedback on various aspects of the response, such as organization, completeness or further required information. If the feedback sequence identifies missing content (for example, "The answer only includes empirical results on QA tasks. Add results from other task types."), the LM also generates a retrieval query for further retrieval using the pipeline.

Iterative refinement. We then iterate over the feedback **F** to incrementally refine the output. If f_k indicates that further retrieval is needed, the query q_k is used to retrieve extra passages, which are appended to **P** before producing y_k. Although we could iteratively regenerate the output each time feedback is provided, doing so introduces more latency. Empirically, we found that feedback is often diverse, addressing different aspects of generation. As a result, sequentially incorporating feedback from the initial output remains effective. The LM uses the previous output y_{k-1}, the retrieved passages **P** and newly retrieved passages, if any, to generate a revised output y_k. This process is repeated until all feedback has been addressed, resulting in a final output y_T by time step T.

Citation verification. Finally, we instruct the generator LM to verify the citations in y_t. Specifically, the generator ensures that all citation-worthy statements—scientific claims requiring justification—are adequately supported by references from the retrieved passages. If any claims lack proper citations, the LM performs a post-hoc insertion to ensure that citation-worthy statements are supported by passages. In our pipeline, we do not remove sentences that lack citation-worthy statements.

Synthetic training data generation with inference pipeline. Building powerful LMs that can effectively synthesize scientific literature is challenging because of the lack of training data for this problem. Although there are some resources to train scientific LMs[38], most tasks do not require open-retrieval settings and are single-paper tasks. As a result, most previous work in this area[10] relies on proprietary LMs, which poses challenges for reproducibility and inference costs.

We use our inference-time pipeline to synthetically generate high-quality training data through self-feedback, so that the resulting model can get better at generating higher-quality output without going through the self-feedback process (Extended Data Fig. 1, bottom).

Question and response generations. Our data generation process involves three steps: first, selecting the top-cited papers from **D**; second, generating information-seeking queries based on their abstracts; and third, using the OpenScholar inference-time pipeline to produce high-quality responses. We generate data using Llama 3.1 70B (ref. 17). Specifically, we begin by sampling 1 million paper abstracts from the peS2o dataset and gathering their corresponding metadata, such as publication year or citation count. We then randomly select 10,000 papers that were published after 2017 and prompt a LM to generate literature review questions or information-seeking queries based on each abstract that require several papers to answer. Next, we use our OpenScholar pipeline to produce the final output y_T, along with intermediate generations such as feedback **F** and initial outputs.

Data filtering. Despite its effectiveness and scalability, synthetic data may also contain issues such as hallucinations, repetitive writing or limited instruction-following[39]. To address this, we introduce a two-step data filtering process: pairwise filtering and rubric filtering, using the same LM as for data generation. In pairwise filtering, we compare the quality of model outputs y_T (output at the final step) and y_0 (initial output) and retain the output that is judged to be higher quality. We find that y_0 is preferred over y_T around 20% of the time, owing to over-editing or increased redundancy after several iteration steps. We then evaluate the quality of the chosen response on a five-point scale across two aspects: organization and factual precision and citation accuracy. A valid model output must achieve a score of 4.5 or higher in both categories and we discard instances whose outputs do not meet this requirement.

Data mixing and training. From this synthetic pipeline, we generate three types of training data: answer generation ($x \rightarrow y$), feedback generation ($y_0 \rightarrow \mathbf{F}$) and feedback incorporation ($y_{t-1}, f_t \rightarrow y_t$). We found that incorporating both final and intermediate outputs during training helps smaller LMs learn to generate more effective feedback. We further blend this synthetic training data with existing general-domain instruction-tuning data[40] and scientific instruction-tuning data[38], ensuring that 50% of the training data come from scientific domains, whereas the remaining 50% is sourced from general-domain data. We also generate synthetic fact verification and Boolean QA data based on sampled abstract data from peS2o. For this, we sort the papers based on citation count and select the top 100,000 papers. After data mixing, we train generator LMs on our large-scale synthetic training data. We train Llama-3.1-8B-Instruct on the generated training data.

OpenScholar experimental details. We use peS2o v2 as **D**, our default data store. For θ_{bi} and θ_{cross} in OpenScholar, we use our trained bi-encoder and cross-encoder models, which consist of 110 million and 340 million parameters, respectively. We analysed various cross-encoder and bi-encoder models on a customized synthetic benchmark and found that OpenScholar retriever (bi-encoder) and OpenScholar reranker (cross-encoder) achieved the highest normalized discounted cumulative gain among models of comparable size (Supplementary Information Section 5.2. We set the maximum number of papers from web search and Semantic Scholar to 10. For the generator LMs, we set the temperature to 0.7 and limit the maximum token count to 3,000 for response generation and 1,000 for feedback generation and use the vLLM package for faster inference. We trained Llama 3.1 8B for two epochs on 130,000 training instances for two epochs. For all models, we set the number of passages input into the generator LM to five for single-paper tasks and ten for multi-paper tasks. No few-shot demonstrations are provided, except for SciFact and PubMed, for which we include one-shot demonstrations. OpenScholar responses are marked

with special decorators Response_Start and Response_End and citations are indicated as reference numbers (for example, [1]), which correspond to the reference documents provided in the context. We do not add any new special tokens to the model vocabulary; instead, we use these decorators as regular strings. After training, we observe that the model can generate the correct tokens as intended.

ScholarQABench

Challenges and overview. Previous studies on building LMs to synthesize scientific literature use either small-scale, single-domain human evaluation[8,9] or oversimplified multiple-choice QA set-ups[10]. Building high-quality benchmarks for literature review has two main challenges. First, creating such datasets is resource-intensive, as it requires PhD-level domain expertise and research experience, particularly when annotating realistic questions and high-quality answers. Second, even when high-quality data are available, reliably evaluating long-form natural language responses presents a notable challenge, especially in expert domains[13,14]. This contrasts with benchmarks for other scientific processes, such as automated experimental code generation, for which clearer evaluation criteria, such as pass@1, are more readily available[45].

To address these gaps, we introduce ScholarQABench, a benchmark that supports diverse formats of scientific literature synthesis tasks, including closed-form classification, multiple-choice and long-form generation, as shown in Extended Data Table 1. We use three existing single-paper datasets and then construct a suite of high-quality, expert-annotated datasets for computer science, biomedicine, physics and neuroscience. We also built a reliable automatic evaluation pipeline. Extended Data Fig. 2 shows an example and an overview of the evaluation pipeline.

Data curation

ScholarQABench is designed to evaluate model capabilities in automating scientific literature review. The curation process is guided by three key factors. Diversity of tasks: ScholarQABench includes tasks with a range of input-output formats. Diversity of disciplines: unlike previous analyses that often focus on a single discipline such as computer science, ScholarQABench spans four scientific disciplines. Inclusion of multi-paper tasks: unlike previous work that focuses on understanding single, preselected papers, all tasks require retrieving from the entire open-access collection of full texts of papers and four datasets specifically require reasoning over several retrieved papers. As a result, ScholarQABench is the first multidisciplinary literature synthesis benchmark that requires long-form generation grounded in several recent papers, with all examples annotated by PhD-level experts. This sets it apart from previous datasets that focus on short-form or multiple-choice answers or rely on static scientific knowledge reasoning[10–12,46], as well as those that lack expert-annotated refs. 13,47.

Note that our benchmark is designed for single-turn set-ups and does not include multi-turn follow-up questions and answers in dynamic evaluations[48]. Evaluating multi-turn LM–human interactions remains challenging[49], so we begin with a single-turn, static evaluation set-up as a first step towards more realistic assessments of such systems.

Single-paper tasks

SciFact. SciFact[42] is a dataset of 1,400 expert-written scientific claims in the biomedical domain, paired with gold evidence from existing PubMed paper abstracts annotated with labels and rationales. We include validation set queries labelled as either 'supports' (true) or 'contradicts' (false), discarding the original gold evidence, and reformulate the task as binary open retrieval, in which a system needs to identify relevant papers from a large collection of papers.

PubMedQA. PubMedQA[41] has expert-annotated (yes/no/maybe) QA data on PubMed paper abstracts. Similarly to SciFact, we only keep instances with yes or no labels and discard the original abstract passage to formulate the task as an open-retrieval set-up.

QASA. QASA[43] is a single-paper QA dataset that consists of question answering pairs, requiring reasoning over scientific articles in artificial intelligence and machine learning. We evaluate the ability of the model to sufficiently answer a detailed question about the target paper. Although the original dataset provides three subtasks (answer selection, rationale generation and answer compositions) as well as end-to-end QA, we evaluate the performance of the models based on an end-to-end QA set-up.

Multi-paper tasks. Single-paper, closed-set tasks may provide reliable evaluations. However, they may not be reflective of realistic scenarios, in which complex, open-ended questions are asked independently from existing papers and require multi-paper retrieval and reasoning. Few datasets[13,47] explore multi-paper set-ups with realistic queries and most lack a reliable evaluation pipeline or human-written references. We address this gap by recruiting expert-level annotators across several scientific disciplines and curating three new long-form QA datasets for this challenging setting. All answers are written by PhD-level experts, with each taking approximately one hour to compose, reflecting the demanding nature of the task. Details of our annotation process, including compensation (US$30–45 per hour on average), are provided in Supplementary Information Section 2.3. The process was approved by the ethics board (institutional review board) as exempt research. Data collection took place between April and October 2024 and all reference answers (where applicable) are grounded in scientific literature published up to October 2024. Below, we discuss each subset of the four multi-paper tasks, which span four broad scientific disciplines.

Scholar-CS. We collected 100 questions along with detailed answer rubrics for each question across various computer science disciplines by recruiting expert annotators holding PhDs in the field (professors, postdoctoral researchers and research scientists). Annotators were tasked with writing literature review questions that require several research papers to answer. The question topics span areas such as networks, algorithms, the Internet of things, artificial intelligence and human–computer interaction. Then, for each question, two other annotators searched the web to produce a rubric listing the key ingredients for a correct answer, categorized by importance ('must have' and 'nice to have'), along with supporting quotes from sources for each ingredient. The annotators were instructed not to use any LLM services for this initial part of the task. After the initial web search, the annotators were shown corresponding responses from four LLM services (Claude 3.5 Sonnet, GPT-4o, Perplexity Pro and an unpublished RAG prototype based on Claude 3.5) in a randomized order in case they wanted to revise their rubrics. On average, each question is annotated with 4.4 key ingredients, each supported by 4.4 quotes. Furthermore, we collected 31 expert-written long-form answers, authored by a separate pool of PhD-level annotators, to serve as a measure of expert human performance.

To measure agreement, we had both annotators produce rubrics for a subset of ten randomly sampled questions. We then compute the scores for responses from the four LLM services to which the annotators were exposed using our automated approach, once for each set of annotator rubrics. Finally, we calculate Pearson's correlation coefficient among the scores for each question and compute the average. Given the subjectivity of rubric annotation, we assess agreement both with and without the general criterion included in the scores, resulting in values of 79.3 and 59.5, respectively. Extended Data Fig. 1 shows an example.

Scholar-Bio and Scholar-Neuro. We further collected 2,759 expert-written literature review questions in biomedicine and neuroscience, recruiting six experts who have a PhD in relevant areas and are at present research scientists and engineers. The annotators were asked

to choose papers from their area of expertise and generate complex scientific questions that biomedical scientists might reasonably ask about the scientific literature based on their parsing of those papers. We collected questions from different areas, such as bioimaging, genetics, microbiology and neuromodulation, for each. Owing to the cost of annotation, we focused only on curating the questions.

Scholar-Multi. Last, we collected 108 literature review questions and expert-written answers with citations in three domains: computer science (artificial intelligence/machine learning, human–computer interaction), biomedicine (bioimaging, genetics) and physics (astrophysics, photonics, biophysics). All annotations are conducted by PhD students or postdoctoral scientists who have more than three years of research experience in the corresponding areas and have several first-author publications. We asked them to come up with questions that are related to the most recent literature and to compose answers to the questions using relevant papers that they found by means of a search. Our annotators were instructed not to use any LLM-based systems such as ChatGPT and told to only use general search (for example, Google Search) or paper search (for example, Semantic Scholar) systems. Statistics of collected questions are available in Table 3, The distribution of subjects is shown in Supplementary Information Fig. 1, along with the average annotation time per subject. We show several examples in Supplementary Information Figs. 12–15. On average, each annotator spent 56 minutes per instance.

Metrics and evaluation protocols

We developed a multifaceted automatic evaluation pipeline to facilitate reproducible and efficient evaluations, complementing expert assessments. An overview of our evaluations is in Extended Data Fig. 2.

Correctness. Correctness evaluates the degree of overlap or agreement between model-generated answers and human-annotated reference answers. This metric is applied only to tasks for which reference answers are available. For single-paper tasks, we directly compare the model outputs to gold reference texts, following the evaluation methodologies proposed in previous work[41–43]. We refer to this metric as accuracy for simplicity. For SciFact and PubMedQA, which have fixed answer classes, we use exact match as the correctness metric. For QASA, we use ROUGE-L as an evaluation metric, following ref. 43.

However, such approaches that rely on a single reference answer often fail to capture all valid outputs, especially in tasks requiring long-form answers synthesized from several papers, such as our multi-paper tasks. To address this, we introduce a new correctness evaluation framework based on Scholar-CS's expert-annotated rubrics, which we refer to as rubric score (rubric-based evaluation). Specifically, we combine two components: annotation-driven criteria (60%), which assess the presence of key content elements ('ingredients') identified by annotators as necessary for a good answer, and general criteria (40% of the score), which evaluate aspects such as length, domain expertise, citation quality and use of supporting excerpts. GPT-4o Turbo scores each criterion and we compute a weighted sum to obtain the final correctness score. We conducted expert evaluations to measure the agreement between human and LLM judges on whether a rubric item was satisfied by a LLM-generated answer, using outputs from two LM systems and two expert annotators. The average agreement between the two human annotators was 0.80, whereas the average agreement between a human annotator and the LLM judge was 0.79. We conducted an analysis on the agreement between an evaluator, LM and a human, and the average correlation between humans was 0.62 and the average correlation between humans and the LLM judge was 0.81. More details are in Supplementary Information Section 2.3.1.

Citation accuracy. Evaluating long-form responses to literature review questions requires citation accuracy: LMs should correctly attribute relevant evidence for all citation-worthy statements. In ScholarQABench, all systems generate outputs with reference numbers (for example, [1], [2]) linked to passages provided during inference. Following previous work[36,50], we check whether each citation-worthy statement has appropriate citations and whether the citations support the statement (citation recall). For each citation, we then verify its relevance and necessity—specifically, whether the citation supports the statement and whether its removal affects the integrity of remaining citations (citation precision). Finally, we compute citation F1 and use it as a primary metric for citation accuracy. Citation accuracy does not require gold reference answers or rubrics, so we apply this evaluation across all tasks. More details are in Supplementary Information Section 2.3.3.

Content quality and organization on Scholar-Multi. We extend our evaluation beyond correctness and citation accuracy by defining further key aspects: relevance to the question, coverage in terms of topic breadth (for example, diversity of discussed papers) and depth (for example, sufficiency of details) and organization and writing flow. These aspects are difficult to capture using standard automatic metrics. We developed detailed instructions and five-point rubrics for each aspect and applied the same rubrics to both LLM and expert human evaluations. For the LLM judge, we use Prometheus v2 (ref. 44), a state-of-the-art open-source model for fine-grained evaluation, chosen to ensure reproducibility and avoid the instability and cost issues associated with proprietary models[51]. For human evaluations, when conducted by expert annotators on those three aspects, we also assess overall usefulness (usefulness). As previous studies show that LLM judges are less reliable when gold reference answers are not available[52], this evaluation is only applied to a task with human-annotated reference answers, namely Scholar-Multi. We analysed the agreement between human and model assessments on fine-grained aspects. We found that, although the model and humans sometimes disagreed on adjacent categories—particularly between scores of 4 and 5—the evaluations of the model aligned well with human rankings and its accuracy on a collapsed three-point rating exceeded 80% across different aspects and subject LMs. More details are in Supplementary Information Section 2.3.2.

Related work

Scientific LMs. Scientific LMs have spanned various domains, including biomedical[53–55], medical[56–59], biomedical[60–62], geoscience[63] and astronomy[64], with some models such as SciGLM[65] and Uni-SMART[66] that aim to cover diverse scientific domains in a single model. Recently, several works show that powerful general-purpose LLMs can also show strong capabilities in scientific tasks, such as medical question answering[56,67], chemistry experimentation[68] and applied mechanics[69]. However, the reliance of a LM on information memorized within its parameters leads to frequent hallucinations in its output[70].

LMs to assist scientists. Recent studies have also examined the capabilities of LLMs to assist scientists in performing a range of scientific procedures, including generating new research ideas[71,72] and automating experimental code generation[73,74]. Our work, however, focuses specifically on benchmarking and developing methods for automating literature reviews and addressing questions related to up-to-date research—tasks that are crucial to, and particularly challenging for, scientific inquiry. Several concurrent studies have attempted to build retrieval-augmented pipelines using proprietary LLMs and external APIs (for example, the Semantic Scholar API) for scientific literature review agents[8,10,75]. Although these studies and our research all explore the potential of retrieval-augmented LMs in automating literature synthesis, previous works often relied on proprietary, black-box systems and limited evaluations, which commonly entail small-scale human evaluation or simplified set-ups such as multiple-choice QA. By contrast, our

work introduces a comprehensive benchmark with automated metrics, involves user studies with experts across three scientific disciplines and develops new methodologies to train specialized open models. OpenScholar greatly outperforms previously introduced systems and shows superiority over human experts in five domains.

Benchmarks for scientific literature understanding. Several works have developed benchmarks to evaluate the abilities of models to understand scientific literature. Previous datasets, such as SciFact[42], QASPER[76] and QASA[43], largely focus on single-paper settings, in which the necessary information to answer queries is contained within a single preselected paper. However, in real-world scenarios, experts often need to synthesize information from several papers to answer questions. To address this gap, ScholarQABench introduces newly annotated tasks that require reasoning across several papers. There are also scientific summarization tasks, such as Multi-XScience[77], in which models are provided with several papers and asked to generate summaries, typically based on the related work sections of those papers. However, in this work, we focus on scenarios in which the relevant papers are not specified in advance, making the task more challenging. Recently, Xu et al.[13] introduced KIWI, a dataset containing 200 questions and human-verified or edited answers generated by state-of-the-art LLMs, with a focus on the natural language processing domain. KIWI also provides a set of relevant papers that models must consider. Although both KIWI and ScholarQABench feature multi-paper, information-seeking tasks, ScholarQABench includes both human-written answers and automatic evaluation pipelines. By contrast, KIWI focuses more on human evaluations and its reference answers are primarily model-generated.

Data availability

Our training data, evaluation benchmark and queries from the OpenScholar demo are all publicly available in the following repositories: https://huggingface.co/datasets/OpenSciLM/OS_Train_Data, https://github.com/AkariAsai/ScholarQABench/tree/main/data and https://huggingface.co/datasets/allenai/openscilm_queries. We provided comprehensive details of our benchmark and training data creation in Methods.

Code availability

All code related to this project is publicly available on GitHub. The main OpenScholar codebase is available at https://github.com/AkariAsai/OpenScholar. The code to run the ScholarQABench evaluation is available at https://github.com/AkariAsai/ScholarQABench. Our public demo as well as expert evaluation interfaces are available at https://github.com/allenai/openscilm and https://github.com/AkariAsai/OpenScholar_ExpertEval, respectively. Our public demo is available at https://openscholar.allen.ai.

45. Si, C., Yang, D. & Hashimoto, T. Can LLMs generate novel research ideas? A large-scale human study with 100+ NLP researchers. In *Proc. 13th Int. Conf. Learning Representations* (Singapore, 2025; https://iclr.cc/Conferences/2025).
46. Rein, D. et al. GPQA: a graduate-level Google-proof Q&A benchmark. In *Proc. 1st Conf. Language Modeling* (Philadelphia, 2024; https://colmweb.org).
47. Malaviya, C. et al. ExpertQA: expert-curated questions and attributed answers. In *Proc. 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)* (eds Duh, K., Gomez, H. & Bethard, S.) 3025–3045 (ACL, 2024).
48. Chiang, W. L. et al. Chatbot arena: an open platform for evaluating llms by human preference. In *Proc. 41st International Conference on Machine Learning* (eds Salakhutdinov, R. et al.) 8359–8388 (PMLR, 2024).
49. Singh, S. et al. The leaderboard illusion. In *Proc. 39th Annual Conference on Neural Information Processing Systems Datasets and Benchmarks Track* (San Diego, Mexico City, 2025; https://neurips.cc).
50. Gao, T., Yen, H., Yu, J. & Chen, D. Enabling large language models to generate text with citations. In *Proc. 2023 Conference on Empirical Methods in Natural Language Processing* (eds Bouamor, H., Pino, J. & Bali, K.) 6465–6488 (ACL, 2023).
51. Kamalloo, E., Upadhyay, S. & Lin, J. Towards robust QA evaluation via open LLMs. In *Proc. 47th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '24)* 2811–2816 (ACM, 2024).
52. Kim, S. et al. The BiGGen Bench: a principled benchmark for fine-grained evaluation of language models with language models. In *Proc. 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)* (eds Chiruzzo, L., Ritter, A. & Wang, L.) 5877–5919 (ACL, 2025).
53. Phan, L. N. et al. SciFive: a text-to-text transformer model for biomedical literature. Preprint at https://arxiv.org/abs/2106.03598 (2021).
54. Yuan, H. et al. BioBART: pretraining and evaluation of a biomedical generative language model. In *Proc. 21st Workshop on Biomedical Language Processing* (eds Demner-Fushman, D., Cohen, K. B., Ananiadou, S. & Tsujii, J.) 97–109 (ACL, 2022).
55. Luo, R. et al. BioGPT: generative pre-trained transformer for biomedical text generation and mining. *Brief. Bioinform.* **23**, bbac409 (2022).
56. Singhal, K. et al. Large language models encode clinical knowledge. *Nature* **620**, 172–180 (2023).
57. Xie, Q. et al. Medical foundation large language models for comprehensive text analysis and beyond. *npj Digital Med.* **8**, 141 (2025).
58. Wu, C., Zhang, X., Zhang, Y., Wang, Y. & Xie, W. PMC-LLaMA: further finetuning LLaMA on medical papers. Preprint at https://arxiv.org/abs/2304.14454v1 (2023).
59. Chen, Z. et al. MEDITRON-70B: scaling medical pretraining for large language models. Preprint at https://arxiv.org/abs/2311.16079 (2023).
60. Luo, Y. et al. BioMedGPT: open multimodal generative pre-trained transformer for biomedicine. *IEEE J. Biomed. Health Inform.* https://doi.org/10.1109/JBHI.2024.3505955 (2024).
61. Zhang, K. et al. BiomedGPT: a generalist vision-language foundation model for diverse biomedical tasks. *Nat. Med.* **30**, 3129–3141 (2024).
62. Labrak, Y. et al. BioMistral: a collection of open-source pretrained large language models for medical domains. In *Findings of the Association for Computational Linguistics: ACL 2024* (eds Ku, L.-W., Martins, A. & Srikumar, V.) 5848–5864 (ACL, 2024).
63. Deng, C. et al. K2: a foundation language model for geoscience knowledge understanding and utilization. In *Proc. 17th ACM Int. Conf. Web Search and Data Mining (WSDM '24)* 161–170 (ACM, 2024).
64. Nguyen, T. D. et al. AstroLLaMA: towards specialized foundation models in astronomy. In *Proc. Second Workshop on Information Extraction from Scientific Publications* (eds Ghosal, T. et al.) 49–55 (ACL, 2023).
65. Zhang, D. et al. SciInstruct: a self-reflective instruction annotated dataset for training scientific language models. In *Proc. 38th Int. Conf. Neural Information Processing Systems (NIPS '24)* (eds Globerson, A. et al.) 1443–1473 (Curran Associates, 2024).
66. Cai, H. et al. Uni-SMART: Universal Science Multimodal Analysis and Research Transformer. Preprint at https://arxiv.org/abs/2403.10301 (2024).
67. Microsoft Research AI4Science, Microsoft Azure Quantum. The impact of large language models on scientific discovery: a preliminary study using GPT-4. Preprint at https://arxiv.org/abs/2311.07361 (2023).
68. Zheng, Z. et al. A GPT-4 reticular chemist for guiding MOF discovery. *Angew. Chem. Int. Ed.* **62**, e202311983 (2023).
69. Brodnik, N. R. et al. Perspective: Large language models in applied mechanics. *J. Appl. Mech.* **90**, 101008 (2023).
70. Li, J. et al. The dawn after the dark: an empirical study on factuality hallucination in large language models. In *Proc. 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (eds Ku, L.-W., Martins, A. & Srikumar, V.) 10879–10899 (ACL, 2024).
71. Baek, J., Jauhar, S. K., Cucerzan, S. & Hwang, S. J. ResearchAgent: iterative research idea generation over scientific literature with large language models. In Proc. *2025 Conf. Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)* (eds Chiruzzo, L., Ritter, A. & Wang, L.) 6709–6738 (ACL, 2025).
72. Yang, Z. et al. Large language models for automated open-domain scientific hypotheses discovery. In *Findings of the Association for Computational Linguistics: ACL 2024* (eds Ku, L.-W., Martins, A. & Srikumar, V.) 13545–13565 (ACL, 2024).
73. Huang, Q., Vora, J., Liang, P. & Leskovec, J. MLAgentBench: evaluating language agents on machine learning experimentation. In *Proc. 41st Int. Conf. Machine Learning* (eds Salakhutdinov, R. et al.) 20271–20309 (PMLR, 2024).
74. Tian, M. et al. SciCode: a research coding benchmark curated by scientists. In *Proc. 38th Int. Conf. Neural Information Processing Systems (NIPS '24)* (eds Globerson, A. et al.) 30624–30650 (Curran Associates, 2024).
75. Wang, Y. et al. AutoSurvey: large language models can automatically write surveys. In *Proc. 38th Int. Conf. Neural Information Processing Systems (NIPS '24)* (eds Globerson, A. et al.) 115119–115145 (Curran Associates, 2024).
76. Dasigi, P. et al. A dataset of information-seeking questions and answers anchored in research papers. In *Proc. 2021 Conf. North American Chapter of the Association for Computational Linguistics: Human Language Technologies* (eds Toutanova, K. et al.) 4599–4610 (ACL, 2021).
77. Lu, Y., Dong, Y. & Charlin, L., Multi-XScience: a large-scale dataset for extreme multi-document summarization of scientific articles. In *Proc. 2020 Conf. Empirical Methods in Natural Language Processing (EMNLP)* (eds Webber, B., Cohn, T., He, Y. & Liu, Y.) 8068–8074 (ACL, 2020).

Acknowledgements We thank our expert annotators for their help curating high-quality data. We thank Y. Wang for his help in developing the human evaluation interface, H. Ivison for providing an earlier version of the Tulu v3 instruction tuning data we used for OpenScholar-8B training and S. Kim for his help on Prometheus evaluations. For assistance with the public demo, we thank C. Anastasiades, T. Anderson, D. Haddad, R. Kinney, S. Lebrecht, C. Nam, W. Smith and A. Zammaron. We thank F. Xu, E. Choi, A. Komatsuzaki, S. Welleck, X. Yue, T. Chen, V. Viswanathan, S. Shen and the members of H2Lab and NeuLab students for fruitful

discussions on this project and feedback on our human evaluation experiments. P.W.K. is supported by the Singapore National Research Foundation and the National AI Group at the Singapore Ministry of Digital Development and Information under the AI Visiting Professorship Programme (award number AIVP-2024-001) and by the AI2050 program at Schmidt Sciences. This work was partially completed when A.A. was part of the UW–Meta AI Mentorship programme.

Author contributions A.A. led the project. Conception and design: A.A., W.-t.Y., P.W.K. and H.H. OpenScholar development was performed by A.A., W.S., R.S. and J.H.; the public demo was developed by A.S., J.C.C., A.A., R.S., D.D., M.L. and J.D.H. The peS2o corpus was constructed by L.S. and K.L. and the data stores and indices by R.S., J.H. and A.A. Licensing discussions involved K.L., L.S., D.D., P.W.K., A.S. and A.A. OpenScholar LM training was conducted by A.A. and W.S.; OpenScholar retrievers were trained and benchmarked by A.A., J.H. and R.S. ScholarQABench was designed by A.A., P.W.K., D. Wadden, D.D., K.L., W.S., A.S., S.F. and D. Weld; single-paper tasks were collected by A.A. and the evaluation pipeline was designed and implemented by A.A. Scholar-CS was collected and evaluated by D.D., A.S., S.F., D. Weld and M.D.; Scholar-Multi by A.A., M.T., R.S., J.H., W.S., P.J., S.L., H.T., B.W. and Y.X.; and Scholar-Neuro/Scholar-Bio by D.D. and J.S. Results and codebases were produced by A.A., J.H., R.S., W.S. and A.S. Human evaluation was designed by A.A., P.W.K. and G.N., with the interface built and supervised by A.A. and M.T. Public demo analysis was performed by J.D.H., A.A., V.K., A.S. and D.D. Advisory: P.W.K., H.H., D.D., W.-t.Y., G.N., D. Weld and L.Z. All authors tested the public demo, contributed to manuscript writing (led by A.A., J.H., D.D., A.S., K. Lo and P.W.K.) and editing and approved the final version.

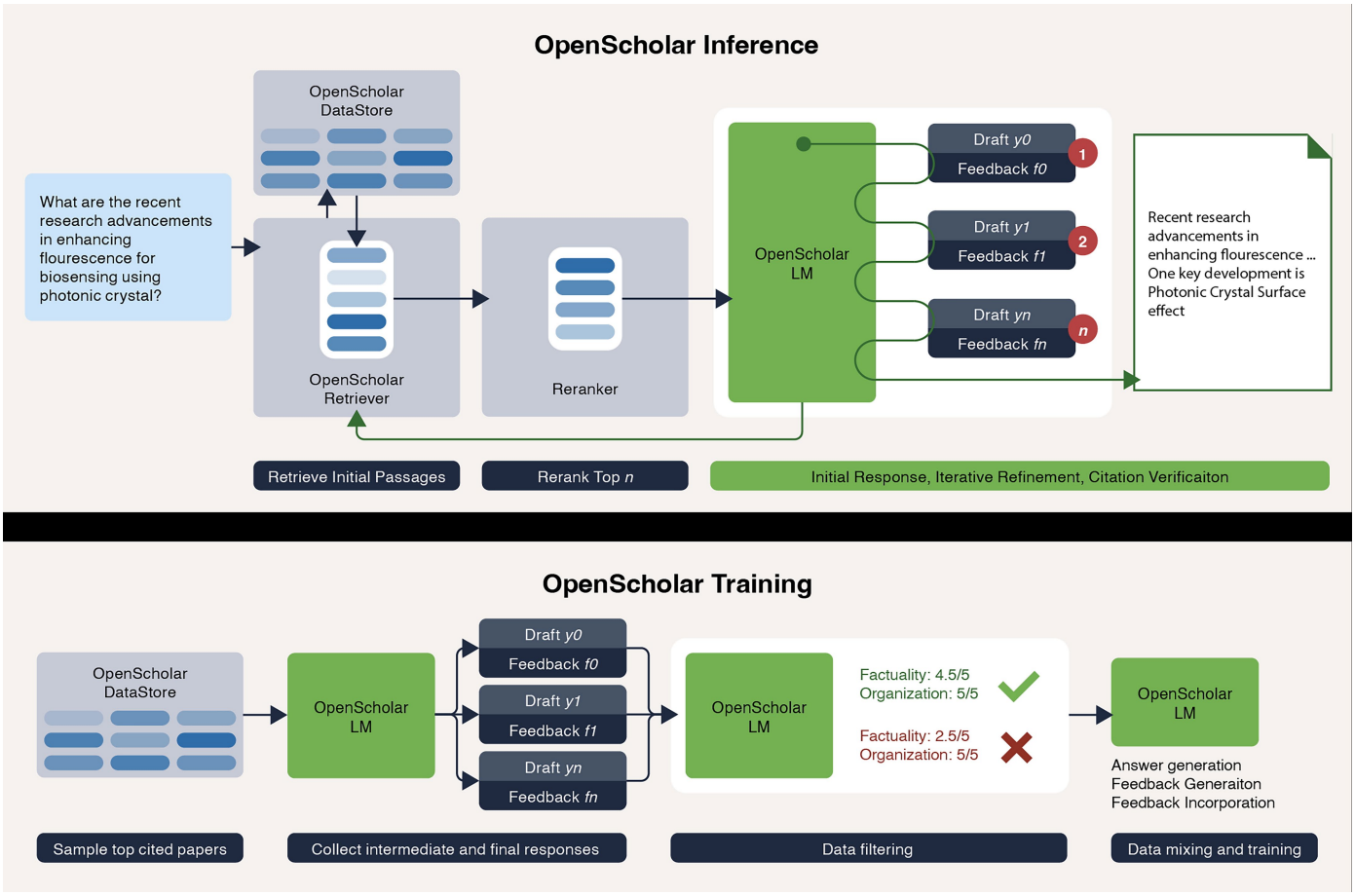
Competing interests The authors declare no competing interests.

Additional information

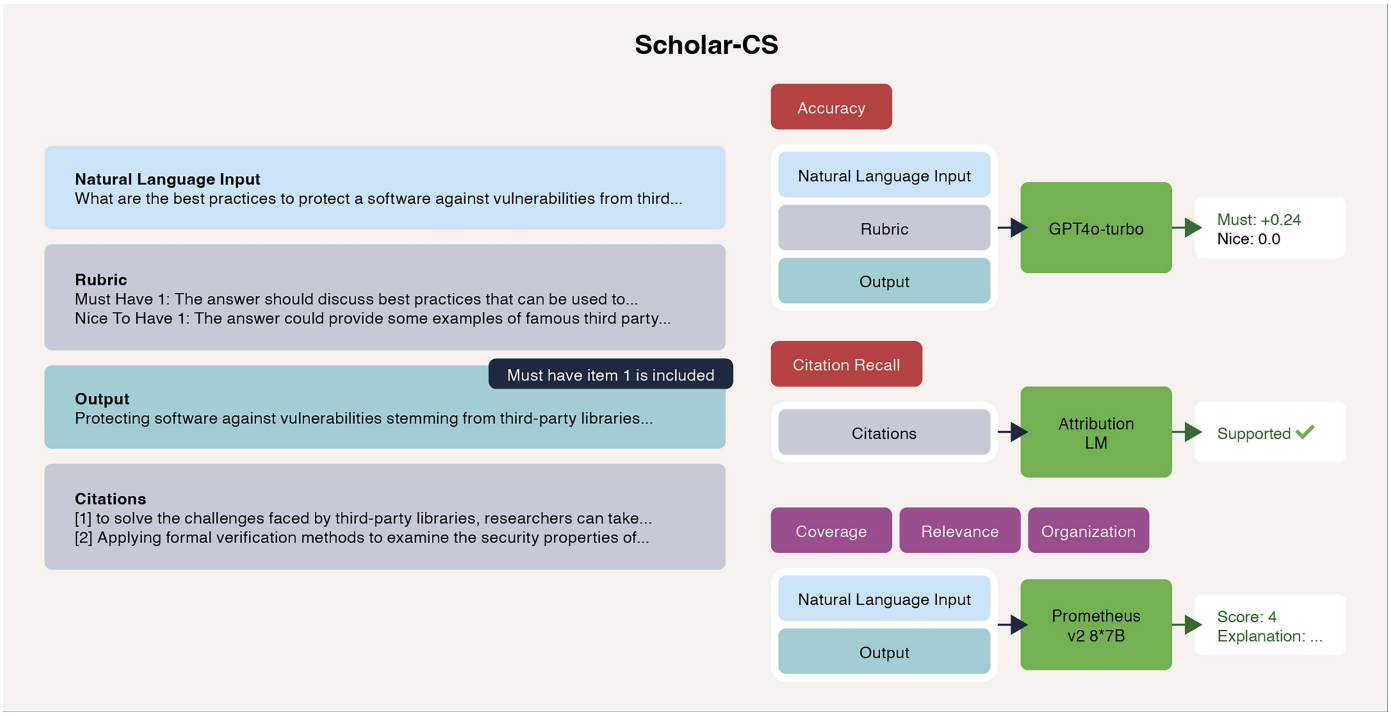
Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s41586-025-10072-4.

Correspondence and requests for materials should be addressed to Hannaneh Hajishirzi.

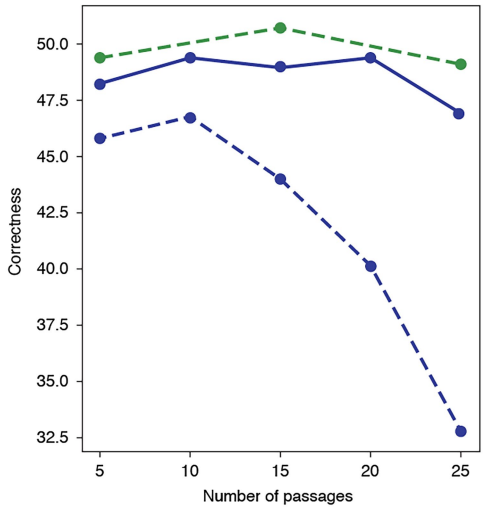
Peer review information *Nature* thanks Shubham Agarwal, Pascal Sun and the other, anonymous, reviewer(s) for their contribution to the peer review of this work.



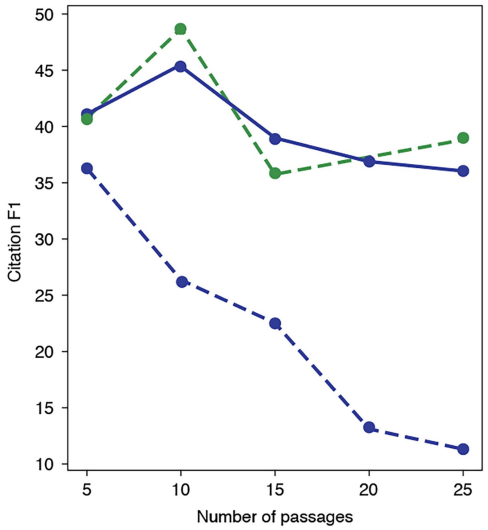
Extended Data Fig. 1 | Detailed overview of the OpenScholar inference (top) and training pipeline (bottom). At inference time, given an input x, OpenScholar first uses a retriever to identify relevant papers from a specialized data store (OSDS) and then uses a reranker to refine and identify the top N retrieved documents. The retrieved output is then passed to the LM, which generates both (1) an initial response y_0 and (2) self-feedback f_1. By incorporating its feedback, the LM iteratively refines its output a pre-defined number of times. Subsequently, a LM (1) generates an initial response y_0, (2) generates self-feedback on the initial output and (3) incorporates feedback (f_i) to generate a revised response y_1. The LM repeats the process until all feedback is incorporated. To train a smaller yet competitive 8B LM, we generate high-quality training data using this inference-time pipeline, followed by data filtering and mixing. We use our new OpenScholar retriever, continue pre-training on the OSDS for the retriever, followed by OpenScholar reranker, fine-tuned on synthetically generated data initialized from the BGE reranker, for the reranker component. For OpenScholar-8B, we use Llama 3.1 8B as the generator and for OpenScholar-GPT-4o, we use GPT-4o as the LM.



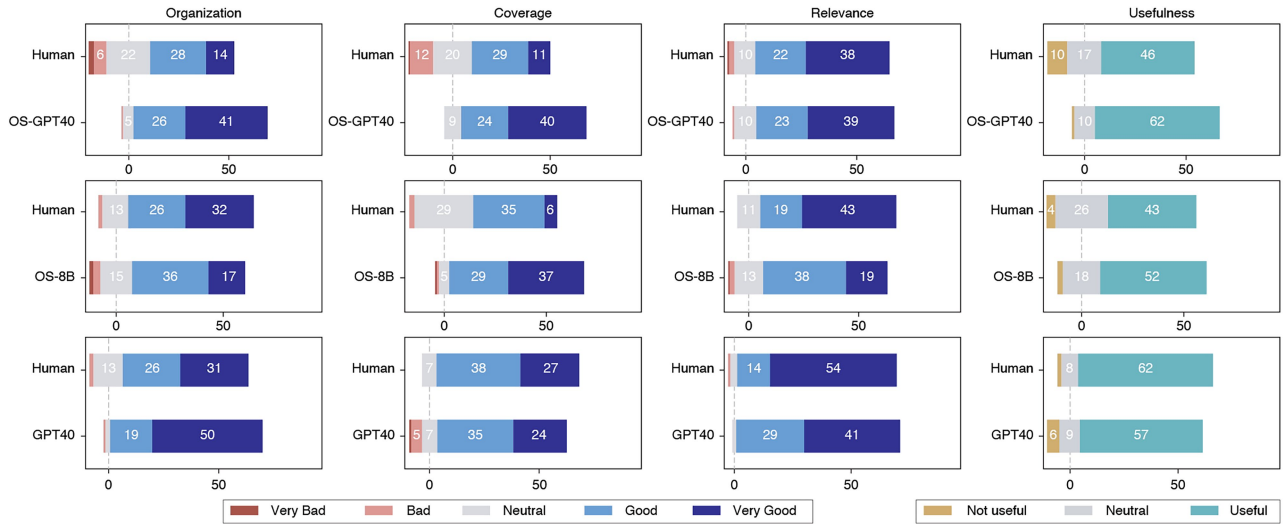
Extended Data Fig. 2 | Example Scholar-CS question, rubric and evaluation pipeline. Scholar-CS consists of 100 questions and an average of 4.4 expert-written rubrics to be satisfied. Our ScholarQABench evaluation pipeline evaluates aspects such as correctness and citation accuracy.



Extended Data Fig. 3 | Effect of context length on Scholar-CS correctness. Analysis of how varying the number of retrieved passages (top *N*) affects rubric scores (correctness) for standard RAG and OpenScholar using Llama 3.1 8B and our trained 8B model.



Extended Data Fig. 4 | Effect of context length. Analysis of how varying the number of retrieved passages (top *N*) affects citation F1 for standard RAG and OpenScholar using Llama 3.1 8B and our trained 8B model.



Extended Data Fig. 5 | Fine-grained expert evaluations of human and model-written answers. Score distributions from expert raters comparing expert-written answers with GPT-4o, OpenScholar-8B and OpenScholar-GPT-4o on Scholar-Multi. The panels show histograms for organization, coverage and relevance on a five-point scale, as well as relative win/tie/lose rates against human answers. OpenScholar-GPT-4o and OpenScholar-8B are preferred over expert answers in most cases, largely because of higher coverage and depth, whereas GPT-4o without retrieval exhibits limited coverage and lower overall usefulness despite strong fluency.

Extended Data Table 1 | Overview of ScholarQABench datasets and evaluation protocols

Dataset	Task Format	Discipline	Size	Evaluation	Avg. $\|x\|$	Avg. $\|y\|$
Single-paper						
SciFact (Wadden et al. 2020)	Claim → Label (True or False)	Biomedicine	208	Acc., Cite	12.4	1.0
PubMed QA (Jin et al. 2019)	Question → Answer (Yes, No)	Biomedicine	843	Acc., Cite	12.8	1.0
QASA (Lee et al. 2023)	Question → Answer (Long-form)	Computer Science	1,375	Acc., Cite	14.2	31.3
Multi-paper						
SCHOLAR-CS	Question → Answer† (Long-form)	Computer Science	100	Rub., Cite	17.4	–
SCHOLAR-BIO	Question → Answer* (Long-form)	Biomedicine	1,451	CIte	17.6	–
SCHOLAR-NEURO	Question → Answer* (Long-form)	Neuroscience	1,308	Cite	16.8	–
SCHOLAR-MULTI	Question → Answer (Long-form)	Computer Science, Physics, Biomedicine	108	Cite LLM, Exp.	19.6	245.6

The top three rows show single-paper datasets taken from previous datasets. The bottom four rows are new datasets, which we constructed by recruiting PhD-level experts. Answer* indicates that the dataset comes with questions only and Answer† indicates that the answer will be evaluated on the basis of human-annotated rubrics. The evaluation columns correspond to the multifaceted evaluations in the Methods ('Metrics and evaluation protocols'). 'Multi-paper' indicates that the task requires several papers to answer. To evaluate response correctness, we use 'Acc.' for single-paper tasks (SciFact, PubMedQA, QASA), which correspond to the primary metrics in the original datasets (that is, accuracy for SciFact and PubMedQA and ROUGE-L for QASA) and 'Rub.' (rubric accuracy) for Scholar-CS. For Scholar-Multi, we assess relevance, organization and coverage using both a LLM judge (aggregated as 'LLM') and expert annotators (aggregated as 'Exp.'). Avg. |x| and Avg. |y| denote the average token length of input and output reference answers (where applicable), respectively.

Extended Data Table 2 | Ablations of training and inference of OpenScholar

	SCHOLAR-CS	
	Rub	Cite
OS-8B	51.1	47.9
- training	49.4	42.3
- reranking	49.6	28.2
- feedback	50.5	41.4
- attribution	49.3	44.0
(a) OSDS only	49.1	32.5
(b) S2 only	47.9	39.1
(c) Web only	45.9	12.6
(a)+(b)+(c)	49.6	47.6
OS-GPT4o	57.7	39.5
- reranking	52.4	22.9
- feedback	55.1	31.0
- attribution	55.6	30.6

Quantitative ablations of OpenScholar components on Scholar-CS. The table reports rubric accuracy and citation F1 for OpenScholar-8B and OpenScholar-GPT-4o, along with variants that remove training, reranking, self-feedback or attribution, as well as retrieval-only variants using OSDS only, Semantic Scholar only or web-only search. Removing reranking or attribution leads to the largest decreases in citation F1, web-only retrieval performs worst overall and combining dense retrieval, Semantic Scholar and web sources yields the strongest factuality and citation support.