SCHOLARCATALYST: A BENCHMARK FOR RETRIEVING PAPERS THAT INSPIRE NEW RESEARCH

Anonymous authors
Paper under double-blind review

ABSTRACT

What makes great scientists great? Even as AI systems start to make progress on open problems, scientists remain far ahead of them at sensing which prior idea, buried in an ever-growing archive of research, a new problem needs. To study this skill, we draw on researchers who know firsthand which earlier work advanced their completed projects, with papers serving as pointers to the ideas within. Using our automated pipeline that makes author annotation scalable, we build **SCHOLARCATALYST** by having 184 lead authors of 207 recent computer science papers label which candidates did or could have advanced their project, each with a detailed rationale. We introduce a *retrieval task* with author-provided judgments: given an initial research question, retrieve these papers from only the literature available when the project began. Agentic search does no better than embedding retrieval (0.42 vs. 0.48 Recall@20) despite calling that same retriever as a tool. Even an agent built on Claude Fable 5.1, which may have seen the completed papers during training, reaches only 0.51 R@20. These results highlight the need for new training recipes that equip models with expert intuition for searching broad corpora. We envision SCHOLARCATALYST as a step toward scientific agents that can take a half-formed idea and point to the prior research it needs.



Figure 1: **SCHOLARCATALYST** asks whether a system can retrieve prior papers that inspire progress on a research project. Such a paper may share little semantic similarity with the question yet offer an insight that advances the project (*right*), while topically related work may not (*left*).

1 INTRODUCTION

To invent is to discern, to choose. — Henri Poincaré

Scientific research is inherently cumulative, with each project building on previous results and ideas. Current AI systems for research mainly accelerate progress on an already well-defined subproblem, where the remaining work is closer to engineering. Deciding which existing idea a new problem should build on is still mostly left to human researchers. An insightful prior paper can turn a vague question into a well-defined problem, for instance by revealing a limitation that motivates a new direction or suggesting an approach worth adapting, thereby shaping and accelerating—***catalyzing***—subsequent research. Making this choice well is a defining ability of great scientists, and a critical

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

Table 2: **Comparison with literature retrieval benchmarks.** SCHOLARCATALYST combines queries that precede a project's solution (*Pre-discovery query*), direct author annotation (*Author annotation*), and relevance judgment beyond what is published in the completed source paper (*Undocumented relevance*).

Benchmark	Relevance	Pre-discovery query	Author annotation	Undocumented relevance	Queries $\lvert\mathcal{Q}\rvert$	Corpus $\lvert\mathcal{P}\rvert$
SciFact (Wadden et al., 2020)	Claim support	✗	✗	✗	1.4K	5.2K
DORIS-MAE (Wang et al., 2023)	Multi-aspect relevance	✗	✗	✗	100	100
ScholarQABench (Asai et al., 2024)	Claim support	✗	✗	✗	2,967	45M
LitSearch (Ajith et al., 2024)	Citation relation	✗	✓	✗	597	64K
MIR (Garikaparthi et al., 2025)	Methodological citation intent	✓	✗	✗	139	4.7K
ScholarCatalyst (ours)	Author-judged inspiration	✓	✓	✓	894	191K

† LitSearch asks authors to write 246 of its 597 queries to retrieve their own papers.

part of what researchers call *research taste*. We ask whether models can make that decision through a concrete task: given an early-stage research question, find its *catalyst papers*, by which we mean prior papers whose ideas did or could have meaningfully advanced the project (Figure 1).

Evaluating this retrieval task requires ground truth about which papers were catalysts for a particular project. Published papers record the outcomes of completed projects, but the ideation that shaped them is largely undocumented, and bibliographies rarely distinguish the sources that provided non-obvious, critical insights from the broader set of cited works. Existing scientific retrieval benchmarks (Table 2) evaluate facet-level similarity or relevance to a stated information need (Wadden et al., 2020; Wang et al., 2023; Ajith et al., 2024; Asai et al., 2024) or derive positives from citation-intent annotations for queries that withhold the eventual method (Garikaparthi et al., 2025). These signals do not capture whether a retrieved paper could have inspired an early-stage project, especially when it was not cited by the eventual paper.

We introduce SCHOLARCATALYST, a literature inspiration retrieval benchmark grounded in researchers' firsthand knowledge of their own projects. 184 researchers who led 207 recent computer science projects verified 894 research questions as they stood before each project's key findings, then labeled which prior papers did or could have advanced their work and explained why, including papers they had not encountered at the time. Given a question, a system must rank these papers near the top of a 191K-paper corpus restricted to work published before the project, using models whose knowledge cutoffs predate every source paper. This knowledge is rarely documented, and we collect it efficiently and at scale through an automated pipeline that only needs a paper's arXiv ID to produce author-reviewable draft data. Because any newly published paper can become a fresh instance, the pipeline keeps SCHOLARCATALYST ahead of model training cutoffs and lets the benchmark grow with the literature.

Across sparse, dense, and multi-vector retrievers and LLM search agents, the strongest system recovers only 48% of gold (i.e., author-credited) papers in its top 20. We find that topical similarity is a weak signal: catalyst papers are no more similar to the query than related papers the authors rejected. Every search an agent issues still goes through a similarity-based retriever, so repeated querying does not help (42% vs. 48% for the retriever alone). We also evaluate Claude Fable 5.1 as a lenient upper reference because its training data may include the source papers or secondary sources describing them, and even with this advantage it misses nearly half of the gold papers in its top 20. More broadly, our results point to the need for retrieval models with *an expert-level sense of which ideas matter*, and SCHOLARCATALYST offers a way to measure progress toward them.

2 RELATED WORK

Scientific discovery using LLMs. Accelerating science is a major opportunity for LLM-based agents, which increasingly can propose relevant hypotheses and research ideas (Wang et al., 2024a;b; Liu et al., 2025; 2026). We deliberately ground our task in existing *papers*. Unlike a newly generated idea, which cannot be verified without running it (Si et al., 2025; 2026), a published paper comes with a record of what worked, why, and where it fell short. The same grounding is what lets us collect ground truth, since authors can judge existing papers against what actually advanced their project. Existing scientific-discovery systems likewise draw on prior papers, either starting from a seed or target paper and exploiting its citation graph or metadata (Wang et al., 2024a; Liu et al., 2025),

while others search a task-specific candidate corpus constructed in advance (Wang et al., 2024b; Liu et al., 2026). Recent work studies how LLMs can generate scientific insights by combining existing ideas (He-Yueya et al., 2026), further motivating the need to evaluate which prior work agents choose to draw on. SCHOLARCATALYST directly addresses this evaluation gap by testing whether scientific agents can retrieve insightful prior ideas for an early-stage research question, guided by project authors' firsthand experience of what helped their research.

Scientific literature retrieval. Existing scientific retrieval benchmarks evaluate how well systems find evidence for claims (Wadden et al., 2020; 2022), retrieve documents for specified facets (Mysore et al., 2021; Wang et al., 2023), and locate papers given document context or explicit information needs (Ajith et al., 2024; Asai et al., 2024; He et al., 2025; Park et al., 2025; Shao et al., 2025). Whereas existing benchmarks provide a retrieval objective that is explicit in the query, we pose an open-ended problem that requires inferring the latent scientific relevance of papers to an early-stage research idea. MIR (Garikaparthi et al., 2025) comes closest, since its queries also withhold the project's eventual method, but it derives positives from citation-intent annotations indicating methodological influence (Lauscher et al., 2022). Rather than inferring relevance from the published record, we elicit directly from paper authors otherwise undocumented data about the research process behind it, including early-stage research questions and relevance judgments. To our knowledge, SCHOLARCATALYST is the first scientific literature retrieval benchmark grounded in authors' firsthand accounts of their research process.

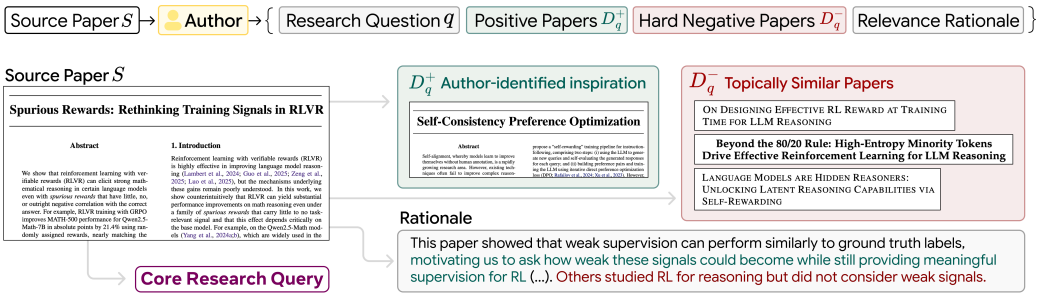
Information retrieval systems. Lexical retrievers (Robertson & Zaragoza, 2009) and embedding retrievers rank corpus papers through term overlap or learned semantic similarity (Karpukhin et al., 2020; Khattab & Zaharia, 2020; Zhang et al., 2025b; Shanbhogue et al., 2026). For a given query, these methods rank the corpus once and do not adapt the search to the returned papers. Search agents use LLM reasoning to iteratively formulate queries and invoke these retrievers as tools over multiple turns, enabling more adaptive search for complex information-seeking tasks (Yao et al., 2022; Jin et al., 2025; Li et al., 2025). Deep research systems extend this approach with subquestion decomposition, reading, and reflection over longer horizons (Team et al., 2025; Zhang et al., 2025a). However, these agents still access the corpus through a retriever's top-k candidates at each step. Recent work instead gives agents direct access to the raw corpus through terminal tools such as grep and file reads, letting the language model determine how to search and interpret documents (Li et al., 2026; Sen et al., 2026). Whether such agentic search approaches can retrieve prior work whose scientific inspiration is latent rather than lexically explicit remains unclear.

3 SCHOLARCATALYST: EVALUATING SCIENTIFIC LITERATURE RETRIEVAL WITH AUTHORS' FIRSTHAND ACCOUNTS

SCHOLARCATALYST evaluates whether retrieval systems can identify *catalyst papers*, prior work whose ideas could inspire and help advance an early stage of research. To obtain firsthand judgments about inspiration, we ask project authors which prior papers did or could have helped advance their project and why, knowledge that is largely absent from the public record. We first define the task of retrieving useful prior work, including its query types, retrieval corpus, and evaluation metrics (Section 3.1). We then describe our pipeline that supports effective and efficient data collection (Section 3.2) and the subsequent author validation and revision process for finalizing the benchmark instances (Section 3.3).

3.1 TASK DEFINITION

Each retrieval task instance $\left(q, \mathcal{D}_q^+, \mathcal{D}_q^-\right)$ is associated with a completed source paper S. Given a retrieval query q, the task is to rank papers from the corpus $\mathcal{P}_{<S}$ such that papers in $\mathcal{D}_q^+$ are ranked highly. To reflect the information available before the project's outcome is known, we define each component as follows. Each query q is a short paragraph that describes the state of a research area as the authors of S saw it before their project and ends with the open question they set out to answer, without revealing the eventual solution. We consider two query types that reflect how an early-stage project develops from a broad central research question into particular subfields, with a useful paper potentially determining the next step at either stage. A **core research query** (**CoreQ**) asks



CoreQ Reinforcement learning with verifiable rewards has proven highly effective at improving mathematical reasoning in large language models. There have been methods showing that using a majority-voted answer as the ground-truth labels can also help improve the model performance. Therefore, we wondered if we can use that to help improve model performance, and if so, how much noise in the majority-voted labels can it tolerate.

SubQ **(Unsupervised RL)** Previous work derives unsupervised reward signals from self-consistency or majority voting rather than ground-truth labels. How much noise can this method tolerate when such signals are unreliable in practice?

SubQ **(Policy optimization dynamics)** Without clipping bias, random rewards should yield no training signal. Can clipping make the objective asymmetric and produce improvement even under random rewards?

Figure 3: **An example SCHOLARCATALYST instance.** An author of the source paper S provides a research question q, expressed as a **CoreQ** or **SubQ**, along with positive and hard negative papers $\mathcal{D}_q^+, \mathcal{D}_q^-$ and a rationale for these judgments.

which prior ideas could help address the project's central research question. A **subfield-specific query** (**SubQ**) approaches the same research question from a particular research subfield, asking which prior ideas could help advance the project from that perspective. Figure 3 shows an example SCHOLARCATALYST instance $(q, \mathcal{D}_q^+, \mathcal{D}_q^-)$, derived from source paper S.

The retrieval corpus $\mathcal{P}_{<S}$ contains only papers published before S. This temporal cutoff (Yetisgen-Yildiz & Pratt, 2008; 2009; Halawi et al., 2024) approximates the literature available during the research process. External web access is disabled to prevent retrieval of subsequent writing about the project, and the source paper S is withheld from the retriever.

For each query q, the positive paper set $\mathcal{D}_q^+ \subseteq \mathcal{P}_{<S}$ contains papers that authors judge to offer inspiration for pursuing q. The hard-negative set $\mathcal{D}_q^- \subseteq \mathcal{P}_{<S} \setminus \mathcal{D}_q^+$ contains papers that the authors reviewed and rejected. Every such paper is topically related to q, since it was either cited in S or ranked highly for q by a retriever. We evaluate retrieval performance using Recall@N, which measures the fraction of positive papers retrieved in the top N results, and normalized discounted cumulative gain (nDCG@N) (Järvelin & Kekäläinen, 2002), which additionally rewards ranking positive papers higher.

3.2 OUR AUTOMATED PIPELINE FOR EFFICIENT DATA COLLECTION

We build an automated data construction pipeline that *only requires a source paper's arXiv ID* and produces the materials required for author annotation. The pipeline let us collect author-validated data at scale. It also makes it easy to refresh SCHOLARCATALYST with newly published papers as older instances go stale due to leakage into newer models' training data.

Source-paper and corpus processing. For each source paper S, we use the arXiv API[1] to obtain metadata and source files, parse the full text into a structured representation, and resolve bibliography entries to canonical scholarly records. The resulting one-hop citation graph links inline citations to the referenced papers' metadata and abstracts. Our retrieval corpus $\mathcal{P}$ contains 190,896 papers. It combines references resolved from our source papers with 181K additional arXiv papers from 2020–2024 across major computer-science categories. Scoping $\mathcal{P}$ to computer science and the source papers' publication years keeps the corpus relevant to SCHOLARCATALYST without indexing

[1] https://info.arxiv.org/help/api/index.html

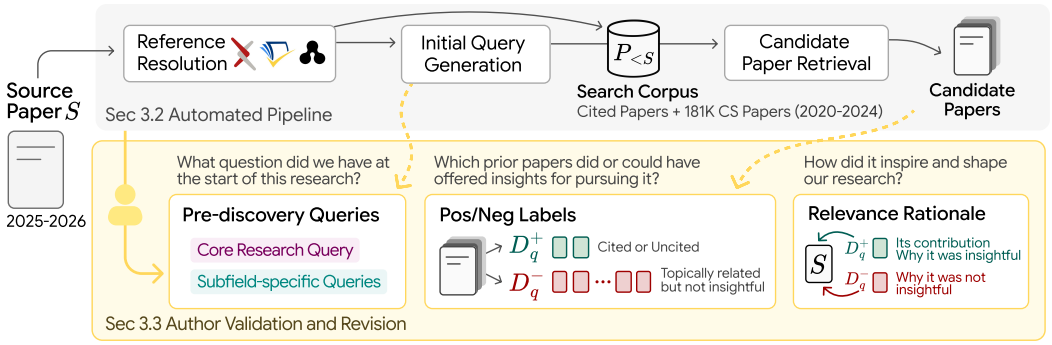


Figure 4: **SCHOLARCATALYST construction process.** Our automated data-collection pipeline constructs a retrieval corpus and author-review materials, including structured paper data and candidate papers (Section 3.2), which authors review through our interface to validate and revise queries, judge candidate-paper relevance, and provide rationales (Section 3.3).

Table 5: **Dataset statistics.** For each query type, we report the number of queries, the average number of positive papers ($|\mathcal{D}_q^+|$) and hard negatives ($|\mathcal{D}_q^-|$), and the average query length in tokens.

Query type	Queries	Avg. $\lvert\mathcal{D}_q^+\rvert$	Cited / Uncited Pos. (%)	Avg. $\lvert\mathcal{D}_q^-\rvert$	Avg. Length
Core Research Query	207	3.19	95.5 / 4.5	10.01	131.8
Subfield-specific Query	687	6.09	56.4 / 43.6	6.88	63.5

all of arXiv, which now exceeds 3 million articles (arXiv, 2026). We show detailed corpus statistics in Section A.2.

Pre-annotation and candidate construction. We use Gemini 3.1 Pro to generate pre-annotations from the structured source paper S and its bibliography. These include one **CoreQ**, several **SubQ**s, and rationales explaining the proposed relevance of cited papers. We additionally retrieve uncited candidates with BM25 and Qwen3-Embedding-8B (Robertson & Zaragoza, 2009; Zhang et al., 2025b). Gemini 3.6 Flash (Google DeepMind, 2026b) reranks the pooled candidates for scientific relevance, and the pipeline retains 10 candidates per query to balance coverage of potential false negatives with a manageable author-review workload.

The resulting pre-annotations and candidates are loaded into a web interface for author review (Section 3.3). We describe the implementation details of our pipeline in Section D.

3.3 COLLECTING RELEVANCE SUPERVISION FROM AUTHORS

We collect 207 projects from 184 researchers, each describing the intellectual path behind a *source paper* S in computer science (CS) and artificial intelligence (AI), published in 2025–2026. Each project contributes one **CoreQ** task instance and one or more **SubQ** task instances, each of the form $\left(q, \mathcal{D}_q^+, \mathcal{D}_q^-\right)$, together with a rationale for the relevance judgments.

1. **Author recruitment.** We recruit participants in two ways. We invite first authors of oral, spotlight, and award papers at major AI conferences in 2025–2026[2]. We also reach out personally and issue an open call to first, co-first, and corresponding authors of any paper accepted at a major AI conference since 2025. More than 100 of the 207 source papers received an oral, spotlight, or award distinction. Full details of the author interface and recruitment procedures are provided in Appendix E.1.
2. **Author validation and revision.** Through a web interface, authors provide an account of the research underlying their source paper S and assess the relevance of candidate papers, as shown in Figure 4. Starting from the pre-annotation of Section 3.2, they can validate, revise, or rewrite any component to reflect their own account of the research (Table 13 shows one validated query

[2]ICLR 2026, ICML 2026, NeurIPS 2025, ACL 2026, CVPR 2026, CoLM 2025, and CoRL 2025.

set). For each query q, authors label every candidate from Section 3.2 as offering an insightful idea for pursuing q or not, which yields the positives $\mathcal{D}_q^+$ and hard negatives $\mathcal{D}_q^-$, and they explain each judgment. Together, these author-validated task instances constitute the benchmark's relevance supervision.

3. **Post-processing.** We manually review all finalized task instances for consistency with the task definition, paying attention to whether queries describe the research question from an early-stage perspective. Of the 894 queries, 764 (85.5%) pass this review unchanged. In the rest, we reword a phrase that gives away the eventual solution or treats the work as complete, keeping the authors' meaning and voice and changing nothing else.

4 Evaluation and Analysis

Our evaluation corpus of 191K computer-science papers is far smaller than the three million papers on arXiv (arXiv, 2026), among which researchers are able to find work that inspires a project. Retrieval in this corpus is therefore an easier version of the problem researchers actually face. In this section, we ask: can current retrieval systems identify the *catalyst papers* that inspired a project from among the many topically related papers in the corpus? If not, where do they fail?

4.1 Experimental Setup

We evaluate lexical, multi-vector, dense, and agentic retrieval. We include BM25 (Robertson & Zaragoza, 2009) as a lexical baseline, since it performs strongly against dense models on BEIR (Thakur et al., 2021) and LongEmbed (Zhu et al., 2024), and LateOn (Sourty et al., 2026) as a multi-vector baseline. We evaluate seven dense retrievers. Qwen3-Embedding-4B and 8B (Zhang et al., 2025b) and Gemini Embedding 2 (Shanbhogue et al., 2026) are general-purpose models. SPECTER2 (Singh et al., 2022) and OpenScholar (Asai et al., 2024) are trained on scientific documents. ReasonEmbed (Chen et al., 2025) and Inf-Retriever-v1-Pro (Yao et al., 2025) are optimized for reasoning-intensive retrieval. Appendix C.1 provides extended results for further embedding retrievers.

We evaluate three search agents that represent common designs in the literature. The grep agent follows DCI-Agent-Lite (Li et al., 2026) and searches titles and abstracts with regular expressions. The tool-calling agent follows the search-then-select structure for literature search (He et al., 2025; Pan et al., 2026). The deep research agent adopts the workflow of deep research systems (OpenAI, 2025; Zhang et al., 2025a): it breaks the query into subquestions and iterates over search, reading, and synthesis. The latter two agents retrieve candidates for their search queries with Gemini Embedding 2. We use GPT-4.1 (OpenAI, 2026a) and o3 (OpenAI, 2026e) as backbone models since they are among the strongest models with a knowledge cutoff (June 2024) before all source papers. To compare against a recent frontier model, we also evaluate Claude Fable 5.1 (Anthropic, 2026), which may have seen most source papers during training (June 2026 cutoff). Following previous work (Thakur et al., 2021; Su et al., 2025; Tchuindjo et al., 2026), we report nDCG@20 and Recall@k ($k = 5, 20, 100$). Full experimental details are provided in Section B.

> Our main setting is retrieval from a **temporally filtered local corpus**. We disallow web search. Agent backbones must have a knowledge cutoff before every source paper (all published 2025 or later). Models past this cutoff serve only as points of comparison (‡ in Table 6).

4.2 Main Results

All systems struggle to retrieve catalyst papers. Current retrievers miss most of the papers that inspired a project (Table 6). General-purpose dense retrievers perform best, yet place 39% of author-identified inspiration papers in the top 20 for **CoreQ** and 51% for **SubQ**. Scientific-document retrievers (SPECTER2, OpenScholar) trail the general-purpose models by at least 16 points on **CoreQ** and 27 points on **SubQ** at Recall@20. BM25 and LateOn trail by 15–16 points on **CoreQ** and 18–25 points on **SubQ**. Together, these results suggest that domain-specific training on citation signals does not by itself solve this task. Among the pre-cutoff systems in our main experiment, **CoreQ** is harder than **SubQ**, consistent with its longer queries and smaller, almost entirely cited positive

Table 6: **Retrieval performance on SCHOLARCATALYST across both query types.** nDCG@20 and Recall@k (k=5, 20, 100) on SCHOLARCATALYST for both query types. Bold marks the best per column among the rows under the rule. Cells are shaded relative to Qwen3-Emb-8B, with green for higher values and red for lower values.

	Core Research Query (n=207)				Subfield-specific Query (n=687)			
Model	nDCG@20	R@5	R@20	R@100	nDCG@20	R@5	R@20	R@100
BM25	0.16	0.12	0.23	0.39	0.21	0.12	0.26	0.45
LateOn 0.1B	0.16	0.12	0.24	0.37	0.27	0.16	0.33	0.55
SPECTER2	0.13	0.10	0.20	0.38	0.18	0.11	0.23	0.42
OpenScholar	0.16	0.13	0.23	0.40	0.19	0.12	0.24	0.46
Qwen3-Emb-4B	0.25	0.18	**0.39**	0.59	0.35	0.21	0.44	0.69
Qwen3-Emb-8B	0.26	**0.21**	0.37	0.58	**0.41**	**0.25**	**0.51**	**0.73**
Gemini-Emb-2	**0.27**	**0.21**	0.38	**0.62**	0.38	0.24	0.46	0.70
ReasonEmbed	0.23	0.18	0.33	0.53	0.33	0.19	0.40	0.65
Inf-Retriever-v1-Pro	0.26	0.20	0.37	0.53	0.31	0.19	0.38	0.63
GPT-4.1 + Grep Agent	0.05	0.04	0.06	0.56†	0.09	0.07	0.09	0.66†
GPT-4.1 + Tool-calling Agent	0.25	0.19	0.37	0.60†	0.37	0.24	0.43	0.70†
o3 + Deep research Agent	0.24	**0.21**	0.33	0.59†	0.33	0.22	0.38	0.69†
Post-cutoff model (comparison only)‡								
Fable 5.1 + Grep Agent	0.41	0.36	0.52	0.67†	0.42	0.28	0.48	0.74†
Fable 5.1 + Tool-calling Agent	0.36	0.29	0.51	0.68†	0.41	0.27	0.49	0.71†
Fable 5.1 + Deep research Agent	0.40	0.35	0.53	0.69†	0.42	0.28	0.50	0.73†

† We backfill agent rankings with papers seen during search and Gemini-Emb-2 results (Appendix B.3).
‡ We evaluate Claude Fable 5.1 (June 2026 knowledge cutoff) as a stronger recent model for comparison.

sets (Table 5). We show later that reranking with an oracle candidate pool recovers part of this gap (Section 4.4).

Agentic search does not improve over embedding retrieval. We compare LLM search agents built on pre-cutoff models (GPT-4.1 and o3) against the strongest embedding model (Table 6). Grep-based search, despite its recent success as a lightweight alternative to embedding retrieval (Li et al., 2026; Sen et al., 2026), performs substantially worse. It finds only 8% of gold query–paper pairs during search, compared with 46% for the tool-calling agent (Figure 7).

We attribute this gap to *candidate coverage*, since regardless of how a model reasons, an agent can interact with the corpus only through its search queries. As the corpus scales, this constraint becomes more important because an agent can inspect only a small fraction of the corpus. Consistent with this account, agent recall rises with backbone capability yet remains close to the embedding retriever's even for the strongest model (Figure 8). If coverage is the limiting factor, progress will require embedding retrievers that surface inspiration papers in the first place and agents that build on those results to guide further exploration. We report Fable 5.1 separately because it falls outside our knowledge cutoff, and we analyze its results, together with agent failure cases and retrieval rationales, in Sections C.3 and C.6.

4.3 ANALYSIS OF AUTHOR-ANNOTATED CATALYST PAPERS

Inspiration relations are diverse and not captured by similarity. Prompting an LLM judge to classify each of the 663 key-inspiration rationales under a nine-type taxonomy (Appendix A.3), we find that authors most often credit a technique they adapted (*method*), empirical findings that supported a direction (*evidence*), an idea extended to a new setting (*generalization*), or a limitation that motivated a new approach (*limitation*), with more than one type applying to 55.8% of rationales (Figure 20). Similarity to the query does not identify these papers: measuring lexical similarity as the fraction of unique query words in a candidate's title and abstract, and semantic similarity as cosine similarity under Qwen3-Embedding-8B (Zhang et al., 2025b), hard negatives are at least as similar to the query as positives on both measures (Table 17). This also limits agentic search, where

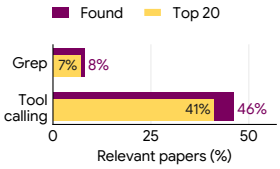


Figure 7: **Grep rarely surfaces gold papers.** Share of gold papers that each GPT-4.1 agent encounters during search (Found) and returns in its final top 20.

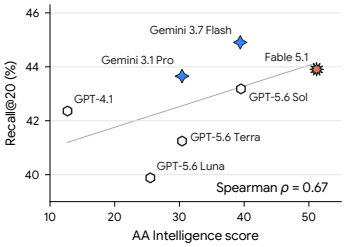


Figure 8: **Agent recall rises with backbone capability.** Tool-calling-agent Recall@20, plotted against general capability score.

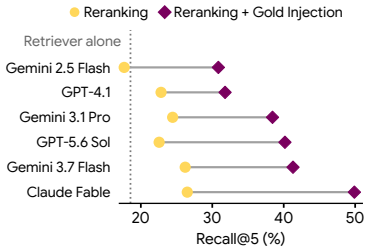


Figure 9: **Stronger rerankers gain more from gold injection.** Recall@5 after reranking the retriever's top 20, with and without missing gold papers swapped in.

Table 10: **Retrieval under different source-paper access conditions.** Recall@20 of the Gemini 3.7 Flash tool-calling agent on the 203-query split.

Setting	CoreQ	SubQ
Agent w/ corpus search	0.39	0.47
Agent w/ web search	0.38	0.47
Agent w/ title and abstract	0.50	0.49
Agent w/ source bibliography	0.74	0.57
Closed-book (no corpus)	0.31	0.24

Table 11: **Query rewriting does not consistently improve retrieval.** Recall@20 with Gemini-Embedding-2 on 100 queries, with GPT-4.1 rewrites.

Method	CoreQ	SubQ
Original query	0.40	0.42
Single-Query Expansion	0.44 (+0.04)	0.37 (−0.05)
Multi-Query Generation	0.37 (−0.03)	0.41 (−0.01)
HyDE (Mean-Pool)	0.23 (−0.17)	0.22 (−0.20)
HyDE (Multi-Query)	0.18 (−0.22)	0.20 (−0.22)

grep and embedding tools construct the candidate pool before an agent can reason about which papers could inspire a project.

The published record does not reveal which papers inspired a project. Source-paper citations provide an incomplete signal of inspiration. 43.6% of **SubQ** positives are not cited by the source paper, and 67.8% of those share no references with it. Among cited papers, citation-intent labels do not identify inspirations: only 46.2% of **CoreQ** and 34.0% of **SubQ** positives carry a `uses` or `extends` label (Lauscher et al., 2022). Even given the entire finished source paper in addition to its bibliography, LLMs fail to identify catalyst papers. Gemini 3.1 Pro (Google DeepMind, 2026a) proposes candidate inspiration papers that cover only 32.2% of **CoreQ** positives on average and none of the key inspirations in 28.0% of instances. Section A.3 and Table 18 give further details, including an instance with both uncited positives and cited hard negatives.

4.4 Analysis of Factors Behind Retrieval Performance

Stronger rerankers benefit more from better candidate coverage. To separate retrieval coverage from recognition performance, we apply LLM-based reranking to the top-K candidates from Gemini-Embedding-2 on a randomly sampled subset of 50 **CoreQ** and 50 **SubQ** queries. We compare reranking with and without gold injection, which replaces negative candidates with positive papers missing from the set while keeping the pool size fixed. Stronger rerankers gain more in Recall@5 from gold injection (Figure 9). The growing gap suggests that candidate coverage limits how much we can benefit from stronger rerankers: their ability to select useful papers matters only when those papers enter the pool. Better retrieval could therefore yield larger gains as rerankers improve. Reranking prompts and procedures are provided in Section C.2.

Agents benefit from source context but not from fuller reading. We vary two kinds of agent context: information about the source paper and how much of each retrieved paper the agent reads. With a Gemini 3.7 Flash (Google DeepMind, 2026c) (cutoff March 2026) tool-calling agent on the 203-query cutoff split (Table 10), the source title and abstract raise Recall@20 by 0.11 for **CoreQ** and 0.02 for **SubQ**, while web search changes it by at most 0.01. The source bibliography, which contains 95% of **CoreQ** gold papers, raises it by 0.35. Without search tools, the model given the title and abstract reaches only 0.31 **CoreQ** Recall@20, below corpus search alone, so source context

helps by guiding search rather than through memory. In contrast, letting a GPT-4.1 tool-calling agent read beyond abstracts on 50 queries per type changes Recall@20 by at most 0.04 (Figure 36), and the best **SubQ** policy still trails the retriever (0.50 versus 0.55). Together, these results suggest that the bottleneck lies in *knowing what to search for* rather than in understanding the papers already found, consistent with the candidate-coverage explanation above. Source information is used only in this analysis, and Sections B.3 and C.5 give details.

Query rewriting does not consistently improve retrieval. We test whether an LLM can improve retrieval by reasoning over the original research question to generate single-query expansions, multiple queries, or hypothetical abstracts (Gao et al., 2023). We evaluate these strategies on 50 queries of each type, using GPT-4.1 for generation and Gemini-Embedding-2 for retrieval. Neither single-query expansion nor multi-query generation yields a meaningful improvement, while both HyDE variants substantially reduce recall (Table 11). If the retriever's misses were a matter of phrasing, an LLM that reads the research question should be able to recover some of them by rewording. That none of these strategies does so suggests the embedding space itself does not place inspiration papers near their queries, consistent with the similarity analysis in Section 4.3. We show prompts and implementation details in Section C.4.

5 Conclusion

We introduced ScholarCatalyst, a benchmark for one ingredient of research ability: identifying prior papers whose ideas could help a project move forward. Drawing on 184 researchers' own accounts of what shaped their work, we capture judgments that the published record largely omits. Current systems recover only a fraction of the papers that authors credit, and agentic search does no better than embedding retrieval, largely because limited candidate coverage keeps many useful papers out of reach.

Limitations. ScholarCatalyst covers computer science papers from 2025–2026 with uneven representation across research areas (Table 14), so our findings may not transfer to other fields. Our labels are retrospective judgments by paper authors and are thus subject to hindsight. A more fundamental open question is the skyline, i.e., the **performance ceiling** of this task. As a point of reference, the human reference accuracy on ImageNet (94.9% top-5 (Russakovsky et al., 2015)) was later exceeded, and reannotation studies found that subsequent gains increasingly reflected the idiosyncrasies of the original labels (Beyer et al., 2020; Shankar et al., 2020). Catalyst paper retrieval is harder to audit this way, since relabeling by anyone other than the authors would be costly and would measure a different target, namely what an informed outsider can infer. Still, a preliminary comparison on one paper with three author-annotators in ScholarCatalyst found that coauthors marking only about seven positives per thread recovered 43–60% of the first author's labels at 84–88% precision. This is well above the best system's overall R@5 of 0.24, which suggests considerable room for improvement.

Outlook. Recent work on **reasoning-intensive retrieval** evaluates relevance relations that require reasoning to recognize (Xiao et al., 2024; Su et al., 2025; Tchuindjo et al., 2026). ScholarCatalyst pushes further, requiring judgments of how prior ideas could advance a research project. We see two milestones ahead. The first is *expert-level* retrieval, matching a domain expert's judgment within their own field. The second is retrieval *beyond any individual expert*: being able to exercise that level of intuition across a breadth of fields no single researcher can follow. This breadth may be where such systems help most, since nearly half of the papers our authors credit (49.5%) come from outside their project's subject area, and 43.6% of subfield-specific positives were never cited in the source paper. Researchers may therefore be limited more by what they have seen than by how well they judge it. The full literature makes this gap far larger than our benchmark shows: Semantic Scholar alone indexes over 225M papers (Kinney et al., 2023), compared with 190k in our corpus. What stands between current systems and both milestones is *an expert-level sense of which ideas matter*, and we hope ScholarCatalyst offers a starting point for developing it. If **undiscovered public knowledge** (Swanson, 1986) is as common as our results imply, models capable of expert judgment across fields could unlock a large backlog of progress hidden in plain sight.

Ethics Statement

This work collects data from researchers describing their own published projects. All contributors participated as unpaid volunteers. All contributing researchers consented to the public release of their responses for future research, including the evaluation or training of AI systems for scientific discovery. They also consented to edits that preserve the intended meaning of their responses. The released data contains no personal information beyond each contributor's account of a published paper. Contributors could choose whether to be acknowledged in the public release of the benchmark, and we will credit only those who opted in.

The resulting labels may reflect individual perspectives and have uneven representation across research areas. They are also retrospective and subject to hindsight. Queries begin as model-generated proposals, and uncited candidates are drawn from a retrieved pool, so the labels may also carry biases of the models that built them, though we did our best to limit this by pooling candidates from different kinds of retrievers and letting authors rewrite any part of a pre-annotation. We therefore present the labels as evidence of inspiration to one project, and they are not intended to measure a paper's overall scientific value or impact.

Reproducibility Statement

Section 3.1 defines the task and metrics, and Section 3.3 describes data collection, with the recruitment procedure and author interface in Section E.1 and the automatic data construction pipeline and corpus construction in Section D. Section 4.1 gives the evaluation protocol, including the temporal cutoff and the withholding of source papers. Retriever configurations are listed in Table 23, agent implementations and prompts are in Section B.3, and the post-cutoff leakage analysis is in Section C.3. We provide an anonymized supplementary archive containing a sample of benchmark instances, the data schema, and evaluation code.

AI Use Statement

We used generative AI tools to generate and analyze data, in addition to polishing our writing. LLMs generated the pre-annotations that authors then validated, revised, or rewrote, and reranked the candidate papers shown to authors (Section 3.3 and Section D). We use an LLM judge to classify the 663 author rationales under our inspiration taxonomy (Section A.3).

References

Anirudh Ajith, Mengzhou Xia, Alexis Chevalier, Tanya Goyal, Danqi Chen, and Tianyu Gao. Litsearch: A retrieval benchmark for scientific literature search. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 15068–15083, 2024.

Anthropic. System card: Claude Fable 5.1 & Claude Mythos 5.1. https://www.anthropic.com/claude-fable-5-1-mythos-5-1-system-card, September 2026. Accessed September 15, 2026.

Artificial Analysis. Artificial analysis intelligence index v4.3. https://artificialanalysis.ai/models, 2026. Per-model scores read on September 16, 2026.

arXiv. arXiv now hosts over 3 million articles. https://blog.arxiv.org/2026/07/09/arxiv-now-hosts-over-3-million-articles/, 2026.

Akari Asai, Jacqueline He, Rulin Shao, Weijia Shi, Amanpreet Singh, Joseph Chee Chang, Kyle Lo, Luca Soldaini, Sergey Feldman, Mike D'arcy, et al. Openscholar: Synthesizing scientific literature with retrieval-augmented lms. *arXiv preprint arXiv:2411.14199*, 2024.

Lucas Beyer, Olivier J. Hénaff, Alexander Kolesnikov, Xiaohua Zhai, and Aäron van den Oord. Are we done with imagenet?, 2020. URL https://arxiv.org/abs/2006.07159.

Jianlyu Chen, Junwei Lan, Chaofan Li, Defu Lian, and Zheng Liu. Reasonembed: Enhanced text embeddings for reasoning-intensive document retrieval. *arXiv preprint arXiv:2510.08252*, 2025.

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, et al. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*, 2025.

Gordon V Cormack, Charles LA Clarke, and Stefan Buettcher. Reciprocal rank fusion outperforms condorcet and individual rank learning methods. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, pp. 758–759, 2009.

Charles Frankston, A. Jonathan R. Godfrey, Shamsi Brinn, Alison Hofer, and Mark Nazzaro. Html papers on arxiv: Why it's important, and how we made it happen. *arXiv preprint arXiv:2402.08954*, 2024.

Luyu Gao, Xueguang Ma, Jimmy Lin, and Jamie Callan. Precise zero-shot dense retrieval without relevance labels. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1762–1777, 2023.

Aniketh Garikaparthi, Manasi Patwardhan, Aditya Sanjiv Kanade, Aman Hassan, Lovekesh Vig, and Arman Cohan. Mir: Methodology inspiration retrieval for scientific research problems. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 28614–28659, 2025.

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

Google DeepMind. Gemini 3.6 Flash: Model card, 2026b. URL https://deepmind.google/models/model-cards/gemini-3-6-flash/.

Google DeepMind. Gemini 3.7 Flash: Model card, 2026c. URL https://deepmind.google/models/model-cards/gemini-3-7-flash/.

Danny Halawi, Fred Zhang, Chen Yueh-Han, and Jacob Steinhardt. Approaching human-level forecasting with language models, 2024. URL https://arxiv.org/abs/2402.18563.

Yichen He, Guanhua Huang, Peiyuan Feng, Yuan Lin, Yuchen Zhang, Hang Li, et al. Pasa: An llm agent for comprehensive academic paper search. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 11663–11679, 2025.

Joy He-Yueya, Anikait Singh, Ge Gao, Michael Y Li, Sherry Yang, Chelsea Finn, Emma Brunskill, and Noah D Goodman. Giants: Generative insight anticipation from scientific literature. *arXiv preprint arXiv:2604.09793*, 2026.

Kalervo Järvelin and Jaana Kekäläinen. Cumulated gain-based evaluation of ir techniques. *ACM Trans. Inf. Syst.*, 20:422–446, 2002. URL https://api.semanticscholar.org/CorpusID:1981391.

Bowen Jin, Hansi Zeng, Zhenrui Yue, Jinsung Yoon, Sercan Arik, Dong Wang, Hamed Zamani, and Jiawei Han. Search-r1: Training llms to reason and leverage search engines with reinforcement learning. *arXiv preprint arXiv:2503.09516*, 2025.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 conference on empirical methods in natural language processing (EMNLP)*, pp. 6769–6781, 2020.

Omar Khattab and Matei Zaharia. Colbert: Efficient and effective passage search via contextualized late interaction over bert. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*, pp. 39–48, 2020.

Rodney Kinney, Chloe Anastasiades, Russell Authur, Iz Beltagy, Jonathan Bragg, Alexandra Buraczynski, Isabel Cachola, Stefan Candra, Yoganand Chandrasekhar, Arman Cohan, et al. The semantic scholar open data platform. *arXiv preprint arXiv:2301.10140*, 2023.

Anne Lauscher, Brandon Ko, Bailey Kuehl, Sophie Johnson, Arman Cohan, David Jurgens, and Kyle Lo. Multicite: Modeling realistic citations requires moving beyond the single-sentence single-label setting. In *Proceedings of the 2022 conference of the North American chapter of the association for computational linguistics: Human language technologies*, pp. 1875–1889, 2022.

Xiaoxi Li, Guanting Dong, Jiajie Jin, Yuyao Zhang, Yujia Zhou, Yutao Zhu, Peitian Zhang, and Zhicheng Dou. Search-o1: Agentic search-enhanced large reasoning models. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 5420–5438, 2025.

Zehan Li, Xin Zhang, Yanzhao Zhang, Dingkun Long, Pengjun Xie, and Meishan Zhang. Towards general text embeddings with multi-stage contrastive learning. *arXiv preprint arXiv:2308.03281*, 2023.

Zhuofeng Li, Haoxiang Zhang, Cong Wei, Pan Lu, Ping Nie, Yi Lu, Yuyang Bai, Shangbin Feng, Hangxiao Zhu, Ming Zhong, Yuyu Zhang, Jianwen Xie, Yejin Choi, James Zou, Jiawei Han, Wenhu Chen, Jimmy Lin, Dongfu Jiang, and Yu Zhang. Beyond semantic similarity: Rethinking retrieval for agentic search via direct corpus interaction. *arXiv preprint arXiv:2605.05242*, 2026.

Yan Liu, Zonglin Yang, Soujanya Poria, Thanh-Son Nguyen, and Erik Cambria. Harnessing large language models for scientific novelty detection. *arXiv preprint arXiv:2505.24615*, 2025.

Yujie Liu, Zonglin Yang, Tong Xie, Jinjie Ni, Ben Gao, Yuqiang Li, Shixiang Tang, Wanli Ouyang, Erik Cambria, and Dongzhan Zhou. Researchbench: Benchmarking llms in scientific discovery via inspiration-based task decomposition. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 13187–13207, 2026.

Kyle Lo, Lucy Lu Wang, Mark Neumann, Rodney Kinney, and Daniel S Weld. S2orc: The semantic scholar open research corpus. In *Proceedings of the 58th annual meeting of the association for computational linguistics*, pp. 4969–4983, 2020.

Meixiu Long, Duolin Sun, Dan Yang, Junjie Wang, Yue Shen, Jian Wang, Peng Wei, Jinjie Gu, and Jiahai Wang. Diver: A multi-stage approach for reasoning-intensive information retrieval, 2025. URL https://arxiv.org/abs/2508.07995.

Sheshera Mysore, Tim O'Gorman, Andrew McCallum, and Hamed Zamani. Csfcube–a test collection of computer science research articles for faceted query by example. *arXiv preprint arXiv:2103.12906*, 2021.

NVIDIA. Llama NV-Embed-Reasoning-3B. https://huggingface.co/nvidia/llama-nv-embed-reasoning-3b, 2026. Hugging Face model card. Accessed: 2026-09-25.

OpenAI. Introducing deep research. https://openai.com/index/introducing-deep-research/, 2025. February 2, 2025; accessed September 19, 2026.

OpenAI. GPT-4.1 model. https://developers.openai.com/api/docs/models/gpt-4.1, 2026a. Accessed September 15, 2026.

OpenAI. GPT-5.6 Luna model. https://developers.openai.com/api/docs/models/gpt-5.6-luna, 2026b. Accessed September 16, 2026.

OpenAI. GPT-5.6 Sol model. https://developers.openai.com/api/docs/models/gpt-5.6-sol, 2026c. Accessed September 16, 2026.

OpenAI. GPT-5.6 Terra model. https://developers.openai.com/api/docs/models/gpt-5.6-terra, 2026d. Accessed September 16, 2026.

OpenAI. o3 model. https://developers.openai.com/api/docs/models/o3, 2026e. Accessed September 15, 2026.

Malte Ostendorff, Nils Rethmeier, Isabelle Augenstein, Bela Gipp, and Georg Rehm. Neighborhood contrastive learning for scientific document representations with citation embeddings. In *Proceedings of the 2022 conference on empirical methods in natural language processing*, pp. 11670–11688, 2022.

Tingyue Pan, Jie Ouyang, Mingyue Cheng, Qingchuan Li, Zirui Liu, Daoyu Wang, Mingfan Pan, Shuo Yu, Qi Liu, and Enhong Chen. Paperscout: An autonomous agent for academic paper search with process-aware sequence-level policy optimization. *arXiv preprint arXiv:2601.10029*, 2026.

Sangwoo Park, Jinheon Baek, Soyeong Jeong, and Sung Ju Hwang. Chain of retrieval: Multi-aspect iterative search expansion and post-order search aggregation for full paper retrieval. *arXiv preprint arXiv:2507.10057*, 2025.

Jason Priem, Heather Piwowar, and Richard Orr. Openalex: A fully-open index of scholarly works, authors, venues, institutions, and concepts. *arXiv preprint arXiv:2205.01833*, 2022.

Stephen Robertson and Hugo Zaragoza. The probabilistic relevance framework: Bm25 and beyond. *Foundations and trends® in information retrieval*, 4(1-2):1–174, 2009.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. Imagenet large scale visual recognition challenge, 2015. URL https://arxiv.org/abs/1409.0575.

Sahil Sen, Akhil Kasturi, Elias Lumer, Anmol Gulati, and Vamse Kumar Subbiah. Is grep all you need? how agent harnesses reshape agentic search. *arXiv preprint arXiv:2605.15184*, 2026.

Madhuri Shanbhogue, Zhe Li, Shanfeng Zhang, Gustavo Hernández Ábrego, Shih-Cheng Huang, Aashi Jain, Daniel Salz, Sonam Goenka, Chaitra Hegde, Ji Ma, Feiyang Chen, Jiaxing Wu, Tanmaya Dabral, Babak Samari, Kevin Poulet, Daniel Cer, Kaifeng Chen, Paul Suganathan, Hui Hui, Jovan Andonov, Philippe Schlattner, Jay Han, Iftekhar Naim, Wing Lowe, Vladimir Pchelin, Albert Yang, Yi-Ting Chen, Zhongli Ding, Grace Zhang, Georg Heigold, Yichang Chen, Antoine Reveillon, Brendan Mccloskey, Wenlei Zhou, Dahun Kim, Rui Meng, Emma Wang, Jack Zheng, Halley Fede, Zhen Yang, Keegan Mosley, Brian Potetz, Sahil Dua, Henrique Schechter Vera, Shen Gao, Hesen Zhang, Andreas Hess, Hengxuan Ying, Alberto Montes, Karan Gill, Min Choi, Sebastian Russo, Anja Hauth, Jinhyuk Lee, Michael Boratko, Megan Barnes, Vikram Rao, Claudiu Musat, Cyril Allauzen, Ehsan Variani, Shankar Kumar, Tom Bagby, Junyi Jiao, Yang Gu, Tengxin Li, Ayush Agrawal, Roberto Santana, Dev Nath, Stephen Karukas, Shuoxuan Han, Lucia Loher, Alice Twu, Nidhi Vyas, Siddharth Bhai, Frank Palma Gomez, Wangyuan Zhang, Chaoren Liu, Jizheng Yang, Steve Qiu, Shijie Zhang, Sujay Kulkarni, Sascha Rothe, Sean Nakamoto, Raphael Hoffmann, Zach Gleicher, Yunhsuan Sung, Qin Yin, Tom Duerig, and Mojtaba Seyedhosseini. Gemini embedding 2: A native multimodal embedding model from gemini, 2026. URL https://arxiv.org/abs/2605.27295.

Vaishaal Shankar, Rebecca Roelofs, Horia Mania, Alex Fang, Benjamin Recht, and Ludwig Schmidt. Evaluating machine accuracy on ImageNet. In Hal Daumé III and Aarti Singh (eds.), *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 8634–8644. PMLR, 13–18 Jul 2020. URL https://proceedings.mlr.press/v119/shankar20c.html.

Chenyang Shao, Fengli Xu, and Yong Li. Scinet: Evaluating ai agents in relation-aware scientific literature retrieval. *arXiv preprint arXiv:2601.03260*, 2025.

Chenglei Si, Diyi Yang, and Tatsunori Hashimoto. Can llms generate novel research ideas? a large-scale human study with 100+ nlp researchers. In Y. Yue, A. Garg, N. Peng, F. Sha, and R. Yu (eds.), *International Conference on Learning Representations*, volume 2025, pp. 94003–94092, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/ea94957d81b1c1caf87ef5319fa6b467-Paper-Conference.pdf.

Chenglei Si, Tatsunori Hashimoto, and Diyi Yang. The ideation-execution gap: Execution outcomes of llm-generated versus human research ideas. In *International conference on learning representations*, volume 2026, pp. 74195–74240, 2026.

Amanpreet Singh, Mike D'Arcy, Arman Cohan, Doug Downey, and Sergey Feldman. Scirepeval: A multi-format benchmark for scientific document representations. In *Conference on Empirical Methods in Natural Language Processing*, 2022. URL https://api.semanticscholar.org/CorpusID:254018137.

Raphaël Sourty, Antoine Chaffin, Paulo Roberto Moura Junior, and Amélie Chatelain. Denseon with the lateon: Fully open dense and late-interaction models for multilingual, long-context, and code search. *arXiv preprint arXiv:2607.27178*, 2026.

Hongjin Su, Howard Yen, Mengzhou Xia, Weijia Shi, Niklas Muennighoff, Han-yu Wang, Liu Haisu, Quan Shi, Zachary Siegel, Michael Tang, et al. Bright: A realistic and challenging benchmark for reasoning-intensive retrieval. In *International Conference on Learning Representations*, volume 2025, pp. 48941–48991, 2025.

Don R Swanson. Undiscovered public knowledge. *The Library Quarterly*, 56(2):103–118, 1986.

Diane Tchuindjo, Devavrat Shah, and Omar Khattab. Obliq-bench: Exposing overlooked bottlenecks in modern retrievers with latent and implicit queries. *arXiv preprint arXiv:2605.06235*, 2026.

Tongyi DeepResearch Team, Baixuan Li, Bo Zhang, Dingchu Zhang, Fei Huang, Guangyu Li, Guoxin Chen, Huifeng Yin, Jialong Wu, Jingren Zhou, et al. Tongyi deepresearch technical report. *arXiv preprint arXiv:2510.24701*, 2025.

Nandan Thakur, Nils Reimers, Andreas Rücklé, Abhishek Srivastava, and Iryna Gurevych. Beir: A heterogenous benchmark for zero-shot evaluation of information retrieval models, 2021. URL https://arxiv.org/abs/2104.08663.

David Wadden, Shanchuan Lin, Kyle Lo, Lucy Lu Wang, Madeleine van Zuylen, Arman Cohan, and Hannaneh Hajishirzi. Fact or fiction: Verifying scientific claims. In *Proceedings of the 2020 conference on empirical methods in natural language processing (EMNLP)*, pp. 7534–7550, 2020.

David Wadden, Kyle Lo, Bailey Kuehl, Arman Cohan, Iz Beltagy, Lucy Lu Wang, and Hannaneh Hajishirzi. Scifact-open: Towards open-domain scientific claim verification. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pp. 4719–4734, 2022.

Jianyou Andre Wang, Kaicheng Wang, Xiaoyue Wang, Prudhviraj Naidu, Leon Bergen, and Ramamohan Paturi. Scientific document retrieval using multi-level aspect-based queries. *Advances in Neural Information Processing Systems*, 36:38404–38419, 2023.

Qingyun Wang, Doug Downey, Heng Ji, and Tom Hope. Scimon: Scientific inspiration machines optimized for novelty. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 279–299, 2024a.

Wenxiao Wang, Lihui Gu, Liye Zhang, Yunxiang Luo, Yi Dai, Chen Shen, Liang Xie, Binbin Lin, Xiaofei He, and Jieping Ye. Scipip: An llm-based scientific paper idea proposer. *arXiv preprint arXiv:2410.23166*, 2024b.

Chenghao Xiao, G Thomas Hudson, and Noura Al Moubayed. Rar-b: Reasoning as retrieval benchmark. *arXiv preprint arXiv:2404.06347*, 2024.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. React: Synergizing reasoning and acting in language models. *arXiv preprint arXiv:2210.03629*, 2022.

Yichen Yao, Jiahe Wan, Yuxin Hong, Mengna Zhang, Junhan Yang, Zhouyu Jiang, Qing Xu, Kuan Lu, Yinghui Xu, Wei Chu, Emma Wang, and Yuan Qi. Inf-x-retriever. https://yaoyichen.github.io/INF-X-Retriever, 2025. Project website.

Meliha Yetisgen-Yildiz and Wanda Pratt. Evaluation of literature-based discovery systems. In *Literature-based discovery*, pp. 101–113. Springer, 2008.

Meliha Yetisgen-Yildiz and Wanda Pratt. A new evaluation methodology for literature-based discovery systems. *Journal of biomedical informatics*, 42(4):633–643, 2009.

Wenlin Zhang, Xiaopeng Li, Yingyi Zhang, Pengyue Jia, Yichao Wang, Huifeng Guo, Yong Liu, and Xiangyu Zhao. Deep research: A survey of autonomous research agents. *arXiv preprint arXiv:2508.12752*, 2025a.

Yanzhao Zhang, Mingxin Li, Dingkun Long, Xin Zhang, Huan Lin, Baosong Yang, Pengjun Xie, An Yang, Dayiheng Liu, Junyang Lin, et al. Qwen3 embedding: Advancing text embedding and reranking through foundation models. *arXiv preprint arXiv:2506.05176*, 2025b.

Dawei Zhu, Liang Wang, Nan Yang, Yifan Song, Wenhao Wu, Furu Wei, and Sujian Li. Longembed: Extending embedding models for long context retrieval. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 802–816, 2024.

APPENDIX

A **The SCHOLARCATALYST Benchmark** 17
- A.1 Benchmark Instance 17
- A.2 Dataset Statistics 17
- A.3 Qualitative Analysis of Inspiration Relations 18

B **Experimental Setup** 19
- B.1 Model Configurations 19
- B.2 Evaluation Setup for Pre-discovery Retrieval 19
- B.3 Agentic Search Baselines 19

C **Additional Analyses** 21
- C.1 Extended Results for Embedding Retrievers 21
- C.2 LLM Reranking 21
- C.3 Post-cutoff Backbones and Benchmark Leakage 22
- C.4 Query Generation Methods 23
- C.5 Reading Full Papers versus Abstracts 24
- C.6 Analysis of Agent Search Failures and Rationale Explanations 24
- C.7 Cost and Search Effort of the Agents 25

D **Automated Data Construction Pipeline** 25
- D.1 Source-paper Processing 25
- D.2 Pre-annotation Generation 27
- D.3 Candidate Retrieval 27
- D.4 Author Validation of the Automated Pipeline 27

E **Author Data Collection** 28
- E.1 Author Recruitment 28
- E.2 Annotation Interface 28

F **Prompts** 28
- F.1 Agent Prompts 28
- F.2 Reranking and Query Generation Prompts 30
- F.3 Pre-annotation Construction Prompts 31

A THE SCHOLARCATALYST BENCHMARK

A.1 BENCHMARK INSTANCE

SCHOLARCATALYST evaluates whether a retrieval system can find prior work whose ideas would have been insightful for pursuing a research project, rather than papers that are merely topically similar. Each instance is anchored in a completed source paper and pairs its central research question and distinct research directions with author-validated positive papers, topically related hard negatives, and a temporally restricted search corpus. The 207 author-validated instances contribute 894 queries: 207 core research queries and 687 subfield-specific queries. This appendix first defines the information in one benchmark instance (Section A.1), then reports the composition of the instances and corpus (Section A.2), and finally examines the kinds of inspiration relations the benchmark captures (Section A.3).

Table 12: Components of a SCHOLARCATALYST instance.

Component	Information
Source paper (S)	The completed paper defining the instance. Withheld from the retriever, so its solution and bibliography cannot be used to find prior work.
Search corpus ($\mathcal{P}_{<S}$)	Papers published before S (temporal cutoff).
Core research query **CoreQ**	Asks which prior ideas could help address the paper's central research question.
Subfield-specific query **SubQ**	Asks which prior ideas from one specific research direction could help advance the paper.
Positive papers ($\mathcal{D}_q^+$)	Papers in $\mathcal{P}_{<S}$ that authors judge to offer an insight that could inspire work on q.
Hard negatives ($\mathcal{D}_q^-$)	Papers in $\mathcal{P}_{<S} \setminus \mathcal{D}_q^+$ that are topically related to q but judged not to offer such an idea.
Rationale	Every query carries a rationale for its positive set.

A project's overarching question and its subfield-specific questions need not draw on the same prior work. The *core research query* can identify ideas that organize the project as a whole, whereas *subfield-specific queries* surface papers relevant to particular methodological or empirical directions. Their overlap and divergence reveal complementary forms of inspiration within a single project. Table 13 shows what this looks like for one full instance.

A.2 DATASET STATISTICS

The author-validated benchmark contains 207 source papers and 894 queries concentrated in machine learning and adjacent AI fields. Table 14 reports each source paper's primary arXiv category,[3] where *Other* comprises stat.ML, cs.SE, cs.IR, cs.MA, cs.HC, cond-mat.mtrl-sci, cs.CE, q-bio.NC, cs.IT, cs.GR, and cs.NE. The author-validated set comprises 894 queries (207 central, 687 directional). Table 5 summarizes the query and corpus statistics.

The retrieval corpus contains 190,896 papers: resolved references cited by our source papers, together with a pool of additional arXiv papers drawn from five main computer science domains and published between 2020 and 2024. Table 15 reports the primary arXiv category of every corpus paper, and Table 16 reports its publication year.

Of the 190,896 corpus papers, 189,296 (99.2%) are sourced from arXiv[4] directly. The remaining 1,600 papers (0.8%) come from Semantic Scholar (S2)[5] and OpenAlex (OA) (Priem et al., 2022), used only when a cited paper could not be found on arXiv itself. Because the corpus includes every resolved reference our source papers cite, it also ends up containing a small number of papers published in 2025 or later, past the nominal collection window between 2020 and 2024. This does

[3] arXiv category taxonomy: https://arxiv.org/category_taxonomy

[4] https://arxiv.org

[5] https://www.semanticscholar.org/product/api

not cause temporal leakage: at evaluation time, we restrict each query's search to only the papers published no later than that query's own source paper.

A.3 Qualitative Analysis of Inspiration Relations

This section provides further detail on the analysis of author-identified inspiration in Section 4.3 and presents additional qualitative examples. To examine the inspirations captured by ScholarCatalyst beyond topical relevance, we qualitatively analyze author-provided rationales for catalyst papers. We first introduce a multi-label taxonomy of inspiration relations, and then present case studies illustrating cross-subfield transfer, the context-dependent role of prior work, and the imperfect correspondence between citation and inspiration. The authors' accounts offer a rare view of the research process that finished papers seldom document: which prior ideas researchers found useful, and why. We find that these judgments are diverse in kind and unrelated to similarity, and that neither citations nor an LLM reading the finished paper can recover them.

Inspiration relations are diverse and not captured by similarity. Prompting an LLM judge to classify each of the 663 key-inspiration rationales under a nine-type taxonomy, we find that authors most often credit a technique they adapted (*method*), empirical findings that supported a direction (*evidence*), an idea extended to a new setting (*generalization*), or a limitation that motivated a new approach (*limitation*), with more than one type applying to 55.8% of rationales (Figure 20). Similarity to the query does not identify these papers: measuring lexical similarity as the fraction of unique query words in a candidate's title and abstract, and semantic similarity as cosine similarity under Qwen3-Embedding-8B (Zhang et al., 2025b), hard negatives are at least as similar to the query as positives on both measures (Table 17). This also limits agentic search, where grep and embedding tools construct the candidate pool before an agent can reason about which papers could inspire a project.

Citation Status Is an Imperfect Proxy for Inspiration. Citation status and author-identified inspiration can diverge in both directions. Table 18 illustrates one source-paper instance in which two of four **CoreQ** positives are uncited, while two cited papers are hard negatives. The uncited positives informed the source paper's exploration and offline-learning mechanisms, whereas the cited hard negatives are topically adjacent but did not shape its approach. This pattern also appears at scale: 43.6% of **SubQ** positives are not cited by the source paper, and 67.8% of those share no references with it. Even among cited papers, citation intent does not single out inspirations: only 46.2% of **CoreQ** and 34.0% of **SubQ** positives carry a `uses` or `extends` label (Lauscher et al., 2022).

Diversity of Inspiration Relations. Prior work inspires new research through diverse mechanisms, including direct method transfer, generalization to a new setting, limitations of existing work, and conceptual reframing. To characterize *how* a key inspiration paper shaped a target work, we define a nine-type taxonomy, given in Table 19, and prompt an LLM judge to label each core-research-query key-inspiration rationale with every type it clearly expresses. To capture the ways a single paper can shape a project, we allow each rationale to receive multiple type labels. The rationales receive 1.69 labels on average, with 55.8% receiving two or more and 12.7% receiving three or more. This pattern shows that inspiration often operates through several mechanisms at once.

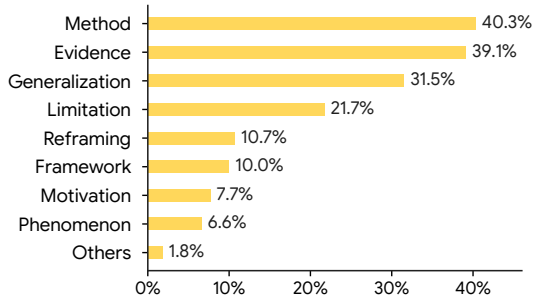


Figure 20: **Distribution of inspiration types.** Share of key-inspiration rationales labeled with each type. Percentages sum to more than 100% because a rationale can receive multiple types.

Context-Dependent Roles of the Same Source Paper. A prior paper does not have a fixed role across research projects. It may offer a transferable method for one target, motivate a different line of inquiry for another, and provide little value for a third despite topical overlap. Its usefulness depends on the question a project is trying to answer. Table 22 shows one key catalyst paper, selected as a **CoreQ** positive by three unrelated source papers, each applying it in a different way.

B Experimental Setup

This appendix specifies the model configurations, agentic baselines, and evaluation conditions used in our experiments. Supplementary analyses are in Section C, and all prompts are in Section F.

B.1 Model Configurations

Table 23 summarizes the retrieval models used in ScholarCatalyst, including their architectures, input limits, and whether they receive a task-specific instruction. Resource links point to the corresponding checkpoint, API, or implementation. Table 24 lists the language-model backbones used for agentic search and LLM reranking. GPT-4.1 and o3 run under the knowledge-cutoff rule of Table 6; the other backbones serve the reranking, backbone-swap, and leakage analyses, and Claude Fable 5.1 also appears in the rows of Table 6 marked ‡.

For Qwen3-Embedding models, we prepend each query with the following task instruction:

```
Given a scientific research question, retrieve papers that could inspire or
contribute to answering the question, including relevant background knowledge,
methods, and ideas.
```

B.2 Evaluation Setup for Pre-discovery Retrieval

Our main setting is retrieval from a temporally filtered local corpus. For each query, we exclude corpus papers published after its source paper, so that the retriever can access only literature available when the project began. We disallow web search. Table 25 reports the resulting searchable corpus size per query. Although the temporal cutoff removes later work, the searchable corpus remains large, with a median of 190,574 papers across both query types.

B.3 Agentic Search Baselines

Shared setup. Under the main setting, agents search a temporally filtered local corpus of paper titles and abstracts, with no access to the web or the source paper, and their backbones have knowledge cutoffs preceding all source papers. The corpus-search tool returns the ten papers the retriever ranks highest under the temporal cutoff. All calls use temperature 0, except GPT-5.6 backbones (Section C.3), which set a reasoning effort instead, and Claude Fable 5.1, which uses its default thinking setting.

Agents return rankings of different lengths, so we score each on a depth-100 list. The list starts with the agent's ranking, adds up to 25 papers the agent saw but did not rank, in discovery order, and fills the rest with the retriever ranking of the original query, without duplicates or the source paper. The cap of 25 keeps an agent that sees many papers but ranks few from displacing the retriever back-fill. R@5 and R@20 thus come from the agent's own ranking unless it holds fewer than k papers, which happens mainly for the grep agent (440 of 894 queries at R@20, against 10 and 95 for the tool-calling and deep research agents, and at most 93 for any Claude Fable 5.1 row). Trajectory recall is the share of gold papers among the papers an agent sees during its run.

Tool-calling agent. The agent combines the ReAct function-calling pattern (Yao et al., 2022), the five-round budget of PaperScout (Pan et al., 2026), and the search-then-select design of PaSa (He et al., 2025). It first searches the query as written, then runs up to 5 rounds, each one backbone reply with any number of search calls. Returned papers enter a pool of at most 60. The backbone then scores each pooled paper from 0 to 10 from its title and abstract, by how much its ideas could help a researcher pursue the query, and the scores sorted with ties in discovery order give the ranking. We do not run PaSa or PaperScout as released, because both search the web and PaperScout releases no

checkpoint of its trained policy, whose base model has no vendor-stated knowledge cutoff. We omit their citation-expansion step because the corpus has no citation graph resolved to corpus papers.

Deep research agent. The agent follows the plan, search, read, and synthesize procedure of deep research systems (OpenAI, 2025; Zhang et al., 2025a), with the same search tool and o3 (OpenAI, 2026e) as backbone. It splits the query into at most 6 sub-questions, searches the query as written, then makes up to 20 searches, 10 reads, and a pool of 150 papers. A read returns the paper's title and abstract. After every 5 calls, a reflection step asks which sub-questions remain uncovered. A final synthesis call ranks up to 50 pooled papers by usefulness, and this list is the ranking, with no per-paper judge.

Grep agent. We re-implement DCI-Agent-Lite (Li et al., 2026), which searches the raw corpus using shell commands rather than a retriever. The released harness has read and shell tools and a 300-turn budget; ours has one shell tool, 20 rounds, and results truncated to 4,000 characters. The corpus is given as monthly files with one paper per line (identifier, title, abstract, date), and the prompt asks the agent to search abstracts with regular expressions and return an ordered list of identifiers. The backbone is GPT-4.1.

Claude Fable 5.1 rows. We also run all three agents with Claude Fable 5.1 (Anthropic, 2026), which compares the procedures using a single backbone. Its June 2026 knowledge cutoff follows all but 6 of the 207 source papers, so these rows fall outside the rule of Section 4.1 and are marked in Table 6. The model's safety classifiers sometimes decline a call, which we treat as an empty reply. Declines occur on 47 of the 894 queries for the tool-calling agent, 95 for the deep research agent, and 33 for the grep agent. About half of the deep research agent's declines are labeled as requests for the model's reasoning, almost all at or right after the reflection step; all others are labeled biology or cyber-security risk. A declined judge call ranks the paper after all scored papers, a declined loop turn ends the search, and a declined synthesis leaves the deep research agent with only the seen papers and the back-fill.

Analysis conditions. The remaining conditions relax the main setting and are used only for analysis in Section C.3, not for direct comparison with the query-only agents. The last three give extra information to the tool-calling agent with Gemini 3.7 Flash (Google DeepMind, 2026c), and all are back-filled to depth 100 as above.

- **Closed-book (recall-from-memory).** A backbone may know the source papers' bibliographies from training, so part of its agent's recall could come from memory. To measure this, the backbone receives the query and the source paper's title and abstract, has no tools, and names up to 30 prior papers. Each title is resolved to the top hit of a retriever search under the temporal cutoff, with no similarity threshold, and we drop duplicates, titles whose search returns nothing, and the source paper.
- **Source paper.** This measures how much the query-only restriction costs, since a real user would know more than the query. The agent's first message includes the source paper's title and abstract.
- **Web search.** This tests whether web access, where the source paper and its citations may appear, can substitute for the local corpus. The agent also gets a web-search tool (ten results per call) and a page-reading tool (pages truncated to 6,000 characters, at most five per query), and must call a tool in its first two rounds. A web result enters the pool only if its address contains an arXiv identifier or its normalized title matches a corpus paper. Other results are shown to the agent but not pooled.
- **Bibliography.** This gives an upper bound on what any leak of the answer could achieve. We map the source paper's bibliography, taken from its arXiv HTML version, to corpus papers by arXiv identifier or exact title (46 of 54 entries on average). The leak upper-bound row of Table 32 adds these papers to the agent's pool and prompt, and the bibliography-alone row ranks them in citation order without a model.

C Additional Analyses

This appendix gives details and supplementary results for our extended comparison of embedding retrievers and for the analyses of LLM reranking, knowledge cutoffs, query generation, full-text reading, and agent search failures.

C.1 Extended Results for Embedding Retrievers

This section reports extended results for a broader set of recent embedding retrievers, including models trained on scientific documents and models optimized for reasoning-intensive retrieval. Recently, several embedding retrievers have been developed for reasoning-intensive retrieval, where identifying relevant documents for complex queries requires reasoning beyond surface-level semantic matching (Xiao et al., 2024; Su et al., 2025; Tchuindjo et al., 2026). We evaluate SciNCL (Ostendorff et al., 2022), OpenScholar (Asai et al., 2024), GTE-Qwen2-7B (Li et al., 2023), ReasonEmbed (Chen et al., 2025), INF-X-Retriever (Yao et al., 2025), Llama NV-Embed-Reasoning-3B (NVIDIA, 2026), and DIVER (Long et al., 2025). Interestingly, none of the reasoning-optimized models improves over the strongest general-purpose embeddings. Their lack of gains suggests that current objectives for reasoning-intensive retrieval do not adequately capture the relationships that make a prior paper useful for advancing a research project. As discussed in Section 5, closing this gap may require an expert-level sense of which ideas matter, enabling retrieval of useful connections across fields beyond the scope of any individual expert.

C.2 LLM Reranking

We test whether an LLM can reorder the retriever's candidates to place gold papers higher, using two procedures over candidate pools from Gemini-Embedding-2. The **listwise reranker** gives the top 20 candidates to a single LLM call, which returns them in ranked order. The **tournament reranker** handles larger pools by ranking batches of $b = 20$ papers and promoting the top $k = 4$ of each batch to the next round, until one batch remains and its ranking is the final order. Both use the prompt in Section F.2. We additionally extend the evaluation to candidate pools of $K \in \{20, 40, 80, 160\}$; Table 27 and Figure 28 report standard and oracle reranking results across these pool sizes.

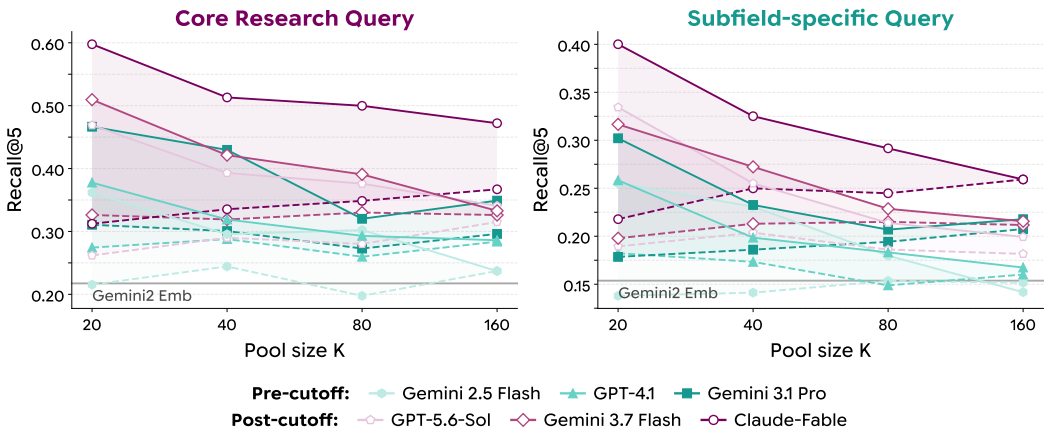


Figure 28: **Reranking performance as the candidate pool grows.** Recall@5 for six LLM backbones across candidate pools from Gemini-Emb-2. Solid curves show oracle reranking and dashed curves show standard reranking. The gray horizontal line marks retriever-only performance.

These results suggest that reranking requires recognizing which prior papers in a retrieved candidate set contain ideas that may inspire the project. Reranking performance generally improves with the backbone's reasoning capability. Claude Fable 5.1 is the most recent backbone we evaluate and achieves the strongest results. Weaker models provide little benefit and can even underperform the retriever-only baseline on SubQ queries.

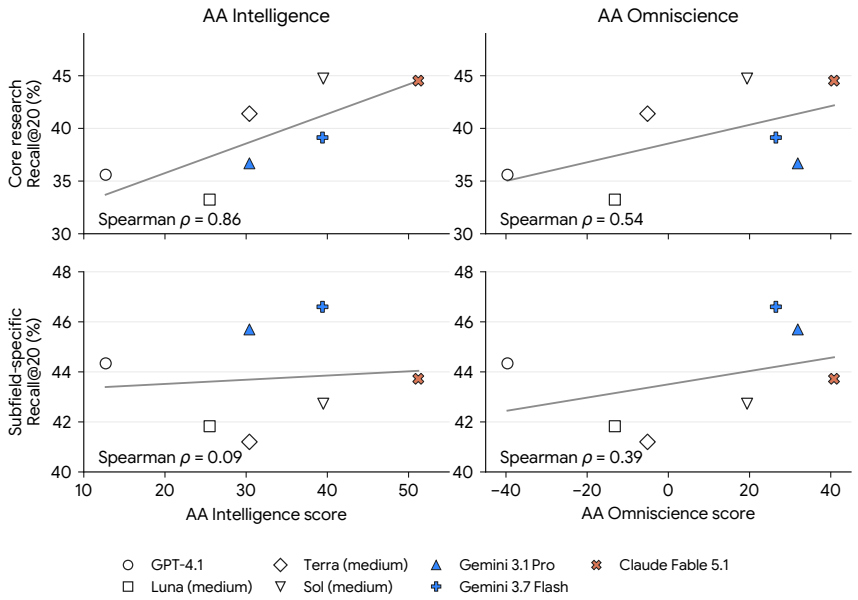


Figure 29: **Backbone capability tracks core research query recall but not subfield-specific query recall.** Recall@20 of the tool-calling agent with seven backbones on the post-cutoff split, with both date slices pooled, plotted against the Artificial Analysis Intelligence and Omniscience scores (Artificial Analysis, 2026). The top row shows the 46 **CoreQ** and the bottom row the 157 **SubQ**. Spearman ρ with the Intelligence Index is 0.86 at the **CoreQ** and 0.09 at the **SubQ**. GPT-5.6 Luna, Terra, and Sol use medium reasoning effort. Lines are least-squares fits, and y-axis ranges differ between rows.

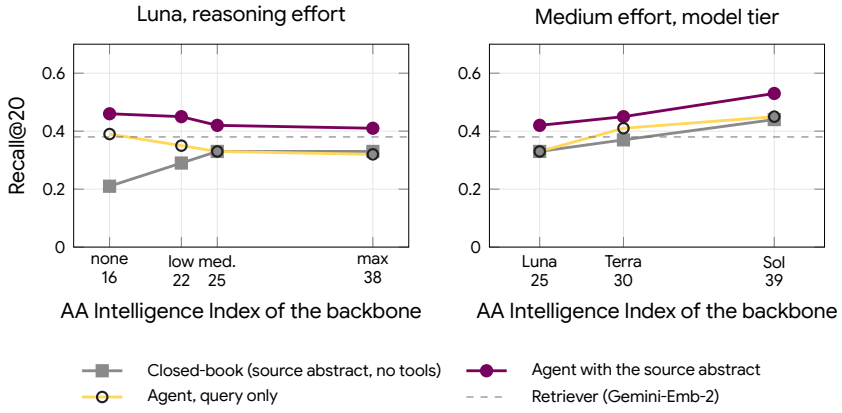


Figure 30: **With the same source context, search helps weaker backbones more than memory alone.** **CoreQ** R@20 over the 46 **CoreQ** of the post-cutoff split, with both date slices pooled. All variants share the February 2026 knowledge cutoff (OpenAI, 2026b;c;d). Filled marks show the tool-calling agent with the source title and abstract, hollow marks show the query-only agent, and closed-book marks show papers named from memory with the same source context. The left panel varies Luna's reasoning effort, and the right panel compares Luna, Terra, and Sol at medium effort. The dashed line marks Gemini-Emb-2 at 0.38.

C.3 Post-cutoff Backbones and Benchmark Leakage

Setup. Our main experimental setup keeps every backbone's knowledge cutoff before every source paper and gives the agent neither the source paper nor web access. This section asks what happens when these restrictions are lifted: whether newer backbones gain from having seen the source papers,

and how much the source paper, the web, or its bibliography would help. We use two slices of the author-written set, matched by primary arXiv category. The pre-cutoff slice contains 23 source papers published between February 12 and December 23, 2025, with two cs.SE slots filled by cs.CE and cs.MA papers. The post-cutoff slice contains 23 source papers published between May 8 and August 14, 2026, four months later. A backbone whose cutoff falls between the slices can have seen the first but not the second.

The pre-cutoff slice is harder for every system (core research query retriever R@100 of 0.49 against 0.63), so we compare each system with the retriever within the same slice. Table 31 evaluates five backbones and their closed-book counterparts, and Table 32 varies the availability of access to source papers, the web, and the bibliography for Gemini 3.7 Flash.

The cutoff split shows no sign that stronger backbones gain from having seen the source papers. With GPT-5.6 Sol and Claude Fable 5.1, the tool-calling agent exceeds the retriever at core research query R@20 on both slices, while the other three backbones stay close to it (Table 31). Because every system scores higher on the post-cutoff slice, we compare each backbone's margin over the retriever within a slice. For GPT-5.6 Sol, whose cutoff falls between the slices, the margin is the same on both (0.07 and 0.06), so it shows no gain from having seen the earlier papers. Claude Fable 5.1 cannot be tested this way because its cutoff falls inside the post-cutoff slice. Closed-book recall rises with capability, but this fits memorization and general capability equally well, so it does not settle the question. At the subfield-specific query level, no agent or closed-book row exceeds the retriever on either slice.

With the same source context, search adds more to weak backbones than to strong ones. Figure 30 holds the February 2026 cutoff fixed across the GPT-5.6 family and varies capability through Luna's reasoning effort and the model tier. Given the source abstract, the agent beats recall from memory by 0.25 for Luna at no reasoning effort, and the gap shrinks to within noise for the stronger runs. Raising Luna's effort helps memory and hurts the query-only agent, while a stronger tier helps both. Figure 29 shows a related pattern across seven backbones: capability tracks core research query recall but not subfield-specific query recall.

The source abstract helps, web search does not, and the bibliography nearly gives away the answer. With Gemini 3.7 Flash, the source title and abstract raise core research query R@20 by 0.11 on both slices (Table 32). Web search leaves recall almost unchanged, even though the source paper appears in the web results for 85 of the 203 queries. The bibliography alone contains nearly all gold papers (R@100 above 0.9). The main protocol therefore withholds the source paper, its bibliography, and the web.

C.4 Query Generation Methods

These methods test whether rewriting the research question with an LLM brings retrieval closer to the gold papers, which often use different vocabulary from the question. All use GPT 4.1, with the prompts in Section F.2 as the system message and the research question as the user message.

- **Single-query expansion.** The model's output, the question plus a short continuation, replaces the question as the retrieval query.
- **Multi-query generation.** The model returns up to five augmented questions, each used as a separate retrieval query. Their top-1,000 rankings are fused with reciprocal rank fusion (k=60, Cormack et al. (2009)).
- **HyDE** (Gao et al., 2023). The model writes five hypothetical abstracts of prior papers, and each is used as a retrieval query, with the rankings fused as above. Its user message is the question prefixed with `Research question:`.

Tables 33 and 34 show generated outputs for the same **CoreQ** instance from the fixed **CoreQ**/**SubQ** subset. For this instance, dense retrieval on the original question recovers a gold paper that the generated queries do not recover in their top 100 results. Table 33 shows the original question alongside single- and multi-query generation. Table 34 shows three hypothetical abstracts generated by HyDE for the same question.



Figure 36: **Reading corpus papers beyond their abstracts adds little.** Recall@20 of the GPT-4.1 tool-calling agent under six read policies and abstract-only reading, on 50 queries of each type from the full-text subset. The top bar is the retriever alone, the second the agent with no read tool.

C.5 Reading Full Papers versus Abstracts

Agents in the main setting see only titles and abstracts. An abstract states a paper's main result but often omits other critical details, so a gold paper could look unrelated to the question from its abstract alone. Indexing full text is not a simple fix, since embedding-based retrievers handle long documents poorly (Zhu et al., 2024) and a paper's body far exceeds the input length they are trained on. We therefore keep search over abstracts and instead let the agent read more of the papers it finds, testing whether that extra text helps it judge and pursue candidates.

Full-text subset. Full text is available for 180,399 corpus papers, converted from arXiv HTML, LaTeX source, or PDF and split by section heading. A query qualifies when all of its gold papers have full text. A source paper enters the subset when its **CoreQ** qualifies and at least one of its **SubQ** does, giving 158 **CoreQ** and 403 **SubQ**. We sample 50 queries of each type, separate from the 100 queries used in the other analyses. The search tool and the back-fill are restricted to the full-text papers. Reads share the five-round budget with searches. All other settings follow that of the tool-calling agent.

Read policies. Four policies add to the abstract the introduction, the method sections, the bibliography, or both the introduction and the bibliography. A fifth returns the full paper, and a sixth lets the agent request one section from a list returned by a second tool. Introductions are matched by their heading, methods by headings containing method, approach, model, framework, algorithm, or architecture, and bibliographies by references or bibliography. If no heading matches, the tool returns the abstract. Each read is capped at 12,000 characters, about 41% of the body at the median for papers read in full, and an included bibliography has its own 6,000-character cap.

C.6 Analysis of Agent Search Failures and Rationale Explanations

We analyze *why* agentic search fails to improve over retrieval, studying three possible causes in turn:

- **Coverage.** We measure how many gold papers agents find outside the retriever's candidates.
- **Ranking.** We trace three search trajectories to see where gold papers are found and lost.
- **Recognition.** We check whether agents explain the gold papers they rank highly the same way the authors do.

Coverage: agents rarely find gold papers beyond the retriever's pool. Table 37 splits gold papers by two outcomes, whether Gemini-Emb-2 retrieves them in its top 100 and whether each agent ranks them in its own top 20. The retriever's top 100 covers 70% of gold papers, and the agents recover almost none of the rest. About half of gold papers are returned by no search during an agent's run. The first search, which uses the query as written, accounts for about two-thirds of the gold papers an agent sees, and follow-up searches add few. The grep agent, which writes its own search commands instead of using the retriever, keeps only 8% of the gold papers the retriever covers.

Ranking: agents drop many of the gold papers they do find. Among the gold papers in the retriever's top 100, the tool-calling and deep research agents each keep 57% and 48% in their top

20. Tables 38 to 40 trace one agent's searches and final ranking on each of three core research query queries, marking gold papers wherever a search returns them. The grep agent retrieves no gold papers in any of the three.

Case A. Ranking failure. The retriever ranks two gold papers first and fifth, and both agents retrieve them in their first search. The tool-calling agent keeps only one in its ranking, and the deep research agent keeps neither.

Case B. Follow-up recovery. A follow-up search by both agents finds a gold paper that the retriever ranks beyond 100.

Case C. Query drift. The retriever's top 30 contains three of the four gold papers. The agents' follow-up searches narrow to a subtopic and miss them, and the deep research agent leaves the one gold paper it does retrieve out of its ranking.

Recognition: agents usually explain gold papers the way the authors do. For the gold papers an agent does rank highly, we check whether its backbone understands why they matter. For each of the 238 core research query gold papers in the tool-calling agent's top 20, GPT-4.1 explains from the query, title, and abstract how the paper could help pursue the query, and a GPT-4.1 judge compares this explanation with the author's rationale. Of the 219 pairs whose author rationale is long enough to compare, 76% describe the same relationship, and almost all others describe a different but valid one (Table 41). Because the same model writes and judges, this agreement is an upper estimate. The analysis also shows only that the backbone can explain a paper's relevance once given it, not that this understanding is why the paper entered the ranking.

C.7 Cost and Search Effort of the Agents

Table 42 reports the search effort, API cost and time per query of each agent in Table 6. With GPT-4.1 and o3, the grep and deep research agents carry a growing context through 14 to 20 tool calls per query and cost about eight times as much as the tool-calling agent, which makes few searches and scores each pooled paper in a short, separate call. With Claude Fable 5.1 the tool-calling agent issues more than twice as many searches within the same five rounds, and every agent costs more than a dollar per query. Times are medians over queries run in parallel and include API latency and retries, so they vary with provider load.

D Automated Data Construction Pipeline

We construct ScholarCatalyst through an end-to-end pipeline that transforms raw arXiv papers into benchmark instances for inspiration literature retrieval. Starting from recent computer science papers, the pipeline constructs a collection of source papers, reconstructs benchmark annotations, assembles a retrieval corpus with a time cutoff, and augments benchmark instances through candidate retrieval and validation. Figure 4 summarizes the overall construction pipeline.

Existing publicly available scientific corpus (Lo et al., 2020) and kaggle arXiv datasets[6] do not provide the combination of recency, full-text representation, bibliography information, and citation relations required by our setting. We therefore build directly from raw arXiv data, processing each paper into a structured representation with explicit citation links that supports all subsequent stages of benchmark construction.

D.1 Source-paper Processing

We enumerate candidate source papers from recent oral and spotlight lists for recent AI conferences, match each accepted paper to its arXiv preprint by title, and exclude surveys and review articles. For each paper, we obtain a structured representation from the arXiv HTML rendering when available and otherwise by converting the corresponding LaTeX source using LaTeXML[7], which better preserves document structure and citation information than PDF-based processing (Frankston et al., 2024). Because some papers cannot be reliably converted due to source formatting or parsing failures, we apply quality filtering to remove papers with incomplete text or citation structure. The

[6] https://www.kaggle.com/datasets/sumitm004/arxiv-scientific-research-papers-dataset
[7] LaTeXML project: https://math.nist.gov/~BMiller/LaTeXML/

Table 38: **Case A. Ranking failure. The deep research agent (o3) retrieves two of the four gold papers in its first search, one of them again in its second, and lists neither in its final ranking. Each gold paper's position in the search result is in parentheses.**

`[[query]] CoT is the main tool used to tackle reasoning problems. There has been much work done investigating the theoretical capabilities of CoT in models from an expressive perspective...`

Call	Search query	Positive papers returned
1	`[[query]]`	Lower Bounds for Chain-of-Thought… (#1); Let Me Think!… (#5)
2	`multi-agent chain-of-thought language models communication complexity limits`	Lower Bounds for Chain-of-Thought… (#2)
3	`multi-agent debate large language models theory communication complexity`	none

Calls 4–22 return no additional gold papers. The final ranking includes neither of the two gold papers found.

Table 39: **Case B. Follow-up recovery. The deep research agent (o3) finds a gold paper the retriever ranks beyond 100 with its second search and places it second in its final ranking. Each gold paper's position in the search result is in parentheses.**

`[[query]] Diffusion models are slow because they repeatedly perform denoising over many timesteps. Dynamic caching methods speed up this process by measuring the distance between intermediate features...`

Call	Search query	Positive papers returned
1	`[[query]]`	Timestep Embedding Tells… (#9)
2	`spectral evolution diffusion models low frequency first generative modeling`	A Fourier Space Perspective on Diffusion Models (#1)

Calls 3–13 return no additional gold papers. The final ranking places *A Fourier Space Perspective on Diffusion Models* second and *Timestep Embedding Tells...* thirteenth.

Table 40: **Case C. Query drift. The retriever's top 30 holds three of the four gold papers, but the tool-calling agent (GPT-4.1) drifts to generic transfer learning and never sees any of them. Each gold paper's position in the search result is in parentheses.**

`[[query]] LLM agents struggle to put past experience to use in complex scenarios. There has been a lot of recent work along similar lines that jointly optimizes memory or reasoning banks together with the model weight...`

Call	Search query	Positive papers returned
1	`[[query]]`	none
2	`meta-learning approaches for generalization in LLM agents`	none
3	`transfer learning in reinforcement learning agents with memory`	none
4	`representation learning for transferable skills in AI agents`	none
5	`case-based reasoning and analogical transfer in AI agents`	none

Call 6 returns no gold papers. The final ranking contains no gold papers.

resulting full text is converted into a structured representation in which inline citations are replaced with explicit citation markers linked to bibliography entries, providing a citation-grounded representation for all downstream stages.

We resolve every cited reference to a canonical scholarly record using the arXiv[8], Semantic Scholar[9], and OpenAlex (Priem et al., 2022) APIs based on the bibliography text. Citation parsing, inconsistent bibliography formatting, and search engine limitations occasionally prevent reliable resolution of the intended paper. We retain only source papers for which at least 70% of cited references are successfully resolved, and discard the rest before they proceed to the remaining pipeline stages, ensuring sufficient citation coverage for annotation reconstruction.

D.2 Pre-annotation Generation

From the structured source paper, we automatically reconstruct the benchmark instances. The model (Gemini 3.1 pro) reads the full paper together with its bibliography to recover the motivating research question (query q, one **CoreQ** and several **SubQ**s), inspiration papers (positive papers D_q^+) for each query. It further generates detailed rationales explaining how each selected paper contributed to the resulting research. These annotations form the benchmark instances used throughout the evaluation. This reconstruction uses the prompts provided in Appendix F.3.

D.3 Candidate Retrieval

We augment the benchmark beyond the paper's cited bibliography by recovering insightful papers that were not cited. For each retrieval query, we retrieve the top-15 candidate papers from the corpus using both sparse (BM25 (Robertson & Zaragoza, 2009)) and dense (Qwen3 8B embeddings (Zhang et al., 2025b)) retrieval systems, deduplicate them, and rerank with a reasoning model (Gemini-3.6-flash) that scores each candidate for scientific relevance to the query rather than mere topical similarity. We retain the top 10 for author review, where authors mark inspiring papers as positives and closely related but non-inspiring papers as hard negatives.

Retrieval corpus. We construct a retrieval corpus containing 190,896 papers. The corpus includes resolved references cited by our source papers together with a pool of 181K additional arXiv papers published between 2020 and 2024, drawn from five main computer science domains (cs.CL, cs.CV, cs.LG, cs.RO, and cs.AI), each restricted to papers with at least 1 citation. To construct a realistic literature pool beyond the cited literature, we estimate the citation distribution of recent ICML, ICLR, and NeurIPS oral and highlight papers and use it to determine the publication-year and category distributions of additional corpus papers. At retrieval time, we additionally apply a per-query temporal cutoff based on the query's source-paper publication date, restricting candidate documents to those published no later than the query paper itself. This precludes temporal leakage, ensuring that no candidate postdates the query paper it is retrieved for.

The prompts used in the pipeline are in Section F.3.

D.4 Author Validation of the Automated Pipeline

Evaluation against author judgments. The final author-written set results from authors validating and revising the pipeline's reconstructions, enabling us to measure how closely the pipeline's initial outputs align with author judgments. In total, 184 researchers completed the validation form for 207 source-paper instances, together reviewing 687 subfields. Table 43 summarizes the validation results across all annotation stages.

Authors rated the reconstructed **CoreQ** as accurate or mostly accurate for 98.1% of the 207 instances (60.9% fully accurate). The same holds for individual subfields: across all 687 subfields, the reconstructed **SubQ** was rated accurate or mostly accurate 95.9% of the time (64.9% fully accurate). The pipeline's reconstructed questions thus capture what authors recall motivating their work, both for a paper as a whole and for each of its research directions.

[8] https://arxiv.org

[9] https://www.semanticscholar.org/product/api

For **SubQ**s, the candidate step shows the value of pooling. Authors retained 85.1% of originally cited candidates as positives, making citations a high-yield starting point. More importantly, authors judged 35.1% of pooled, non-cited candidates—papers absent from the source paper's bibliography—as genuine inspirations.

Finally, we evaluate the pipeline's key-inspiration nomination for **CoreQ**s. For each paper's **CoreQ**, the pipeline selects cited papers as likely key inspirations. We compare these nominations with the cited papers that the author ultimately labels positive.

Model-generated dataset. Given this fidelity, we apply the pipeline to 250 additional papers to form the model-reconstructed set, yielding 250 central queries and 863 directional queries.

E Author Data Collection

E.1 Author Recruitment

We recruit first-author, co-first-author, and corresponding authors of highlighted papers (oral, spotlight, and award) at recent iterations of ICLR, ICML, NeurIPS, CVPR, CoLM, CoRL, and ACL, together with institution-wide public recruitment open to any first, co-first, or corresponding author of a paper accepted at a major AI conference from 2025 onward. Participants were invited through direct email and public calls. The recruitment page in Figure 44, and the annotation interface in Figure 45. In total, 184 authors contributed 207 author-written instances.

E.2 Annotation Interface

We developed a web-based interface through which authors annotate their own research projects. For each source paper, authors write or revise a research question intended to reflect their pre-solution information state. They organize the question into distinct research subfields and identify prior papers that provided meaningful scientific inspiration for each direction. Authors then review additional candidate papers, labeling papers that provided meaningful inspiration as positives and plausible but non-inspirational papers as hard negatives. For each judgment, they provide a rationale explaining how the paper did or did not contribute to the development of the target work. The resulting author-written instances contain research questions, positive and hard-negative labels, and open-ended rationales.

F Prompts

F.1 Agent Prompts

Prompt for the Tool-Calling Agent Search Loop

```
You are a research agent. Given an early-stage research question, find prior
papers whose ideas could genuinely help a researcher pursue it, not papers that
are merely on the same topic. You have at most 5 rounds of tool calls. Call
search with different angles of the question. Do not assume or reveal a
solution to the question. When you have enough candidates, reply with the
single word DONE.
```

Prompt for the Tool-Calling Agent Candidate Judge

```
Research question:
<research question>

Candidate paper:
Title: <title>
Abstract: <abstract>
```

```
Would this paper's ideas meaningfully help a researcher pursue the research
question, beyond being topically related? Reply with one integer from 0 (no) to
10 (certainly), nothing else.
```

Prompt for the Deep Research Agent Search Loop

```
You are a research agent. Given an early-stage research question, find prior
papers whose ideas could genuinely help a researcher pursue it, not papers that
are merely on the same topic. Work through the sub-questions of your plan: call
search with different angles (20 searches at most) and read on promising papers
to see their full abstract (10 reads at most). Do not assume or reveal a
solution to the question. When the plan is covered, reply without calling a
tool.
```

Prompt for Deep Research Planning

```
Research question:
<research question>

Split this question into at most 6 sub-questions a literature search should
answer to find prior work whose ideas could help pursue it. Reply with a JSON
array of strings and nothing else.
```

Prompt for Deep Research Reflection

```
Reflection. The plan was:
<the plan, one sub-question per line>

Which sub-questions are still uncovered by the papers found so far, and what
will you search or read next? Reply in a few sentences, without calling a tool.

Continue with the searches and reads your reflection names.
```

Prompt for Deep Research Synthesis

```
Papers found (<pool size>):
[<id>] <title>
...

Research question:
<research question>

From these papers only, pick the ones whose ideas would meaningfully help a
researcher pursue the question, most useful first, at most 50. Reply with a
JSON array of their ids and nothing else.
```

Prompt for the Grep Agent

```
You are a research agent searching a local paper corpus for prior work whose
ideas could genuinely help a researcher pursue an early-stage research
question, not papers that are merely on the same topic.
The current directory holds the corpus as files named YYYY-MM.jsonl, one paper
per line as JSON with the fields id, title, text (the abstract) and published;
only papers the researcher could have read are present. Search the abstracts,
not only the titles. Use bash with ripgrep, for example
```

```
  rg -i 'phrase one|phrase two' *.jsonl | head -c 4000        # matching papers
  with id, title and abstract
  rg -i -c 'phrase' *.jsonl | sort -t: -k2 -nr | head          # which months
  have the most matches
  rg -i -o '"id": "[^"]+", "title": "[^"]{0,120}' 2023-*.jsonl | rg -i 'phrase'
  | head -30   # compact id + title list
Try 5 to 10 different angles, including specific technical terms, method names
and problem formulations that a relevant paper would use even if the question
does not. Read the abstracts of at least 20 candidates before answering; a
paper counts only if you have seen its abstract. You have at most 20 commands.
Do not assume or reveal a solution to the question. When done, reply with a
JSON array of paper ids, most useful first, and nothing else.
```

F.2 Reranking and Query Generation Prompts

Prompt for Listwise and Tournament Reranking

```
[System]
You are an expert at scientific literature retrieval.
Your goal is to identify prior papers that could help a researcher pursue the
given research question.
Given a research question and a list of candidate papers, rank ALL papers from
most to least likely to be useful for advancing this research question.
Rank a paper higher when its scientific contribution would meaningfully inform
how a researcher thinks about or approaches this question.
Rank it lower when it is merely topically related and does not provide insight
for this research.
Output only a JSON array of paper IDs in ranked order, most relevant first.
Example: ["id_a", "id_b", "id_c"]

[User]
Query: <research question>

Papers:
id=<paper_id_1> title=<title_1>
abstract=<abstract_1>

id=<paper_id_2> title=<title_2>
abstract=<abstract_2>

...
```

Prompt for Single-Query Expansion

```
You are helping a researcher find prior scientific papers that could help them
pursue an early-stage research question. Augment the researcher's question to
make retrieval more effective. Keep the original question text unchanged, and
add a short continuation that makes the search more specific, using concrete
terms that prior work in this area would use. Do not change or remove any part
of the original question. Do not introduce a different research direction or a
solution. Write natural prose only, no keyword lists. Output the original
question followed by the continuation.
```

Prompt for Multi-Query Generation

```
You are helping a researcher find prior scientific papers that could help them
pursue an early-stage research question. Generate up to 5 augmented versions of
the researcher's question. Each version must keep the original question text
unchanged and add a short continuation that makes the search more specific.
Make the continuations diverse: each should point toward a distinct plausible
line of prior work that could inform the same research situation. Do not
introduce a different research direction or a solution. Write natural prose
only, no keyword lists. Output one augmented question per line and nothing
else.
```

Prompt for HyDE Hypothetical Abstracts

```
You are helping a researcher find the prior work that inspired a new research
idea.

Below is a research question written by the authors of a paper before they
solved it. Your job is NOT to answer the question. Your job is to imagine the
earlier papers that the authors might have read and drawn on when forming this
question.

Write 5 hypothetical abstracts of such prior papers. Each abstract should:
- Read like a real paper abstract (title + 120-180 words): problem, method, key
finding.
- Describe a paper that already existed before the question was written, so it
must not solve the question itself. It contributes one idea, tool, observation,
or framing that the authors could have borrowed.
- Come from a different angle than the others. Cover at least one adjacent
field or task where the same underlying idea appears under different
terminology.
- Use the vocabulary that paper's own community would use, not the vocabulary
of the question.

Do not name real papers, authors, or years. Do not mention the research
question or the target paper. Output only the abstracts.

Output format (JSON):
[
  {"title": "...", "abstract": "..."},
  ...
]
```

F.3 Pre-annotation Construction Prompts

This section details the exact prompts used in our pre-annotation construction pipeline. We show the prompt for reconstructing the research question and key inspiration papers, for identifying meaningful subfield direction, and for extracting key citations from each subfield.

Prompt for Research Question and Key Inspiration Reconstruction

```
You are an expert research analyst.

Given a research paper and its bibliography, your task is to reconstruct the
research question the authors were pursuing before they developed the solution
presented in the paper, and identify the cited prior paper(s) whose
contribution
most directly inspired the target paper's central contribution.
```

```
<paper>
{paper_text}
</paper>

<bibliography>
{bibliography}
</bibliography>

Each citation in the paper is marked inline as `[CITE:<key>]` and listed in the
bibliography above. Use the exact `<key>` string when referring to citations in
your output.

**Task:**
1) Identify the line of research this paper belongs to and reconstruct the
research context that led to the authors' research question. Describe this from
the authors' own perspective rather than as a neutral outside summary. Use only
information that would have been available before this work was conducted.
2) Write the research question the authors would have asked from this context,
before they knew how to solve it. Briefly describe only the research context
that
naturally led to the research question, then end with exactly one central
research question.
3) Given this research question and the target paper, identify the cited prior
paper(s) whose contribution most directly inspired the target paper's central
contribution. A key inspiration paper is not simply the most relevant,
representative, or highly cited prior work. Instead, it is the prior work whose
contribution most directly inspired the target paper's central contribution.

**Research Question Requirements:**
- Reconstruct only the research context available before the work was
conducted.
- Reflect the authors' research perspective rather than a broad field-level
question. Briefly describe only the scientific problem and its limitations that
naturally led to the research question.
- End with exactly one research question in question form.
**Avoid:**
- Revealing or implying the paper's eventual answer, solution, proposed
paradigm,
or high-level approach.
- Using the paper's own methods, systems, datasets, benchmarks, findings, or
other solution-specific concepts.
- Using terminology, names, or phrases introduced by the target paper itself.

**Key Inspiration Requirements:**
**Selection Criteria:**
- Select the cited prior paper(s) using the exact bibliography key(s).
- Select only the cited paper(s) whose contribution most directly inspired the
target paper's central contribution.
**Rationale Requirements:**
- Briefly explain the selected paper's main contribution.
- Explain what idea from this work informed the target paper's central
contribution and how it influenced that contribution.
- Explain why this work should be regarded as a primary source of inspiration.
- Focus on explaining the scientific connection between the selected work and
the
target paper, rather than summarizing either paper independently.
**Avoid:**
- Selecting papers simply because they are representative, foundational, or
highly
cited unless they directly inspired the target paper's central contribution.
- Referring to the selected work by its citation key, title, or author names.
Instead, describe it through its contribution and scientific role.

**CRITICAL:**
```

```
- Every `cite_key` must exactly match a key listed in the bibliography.
- NEVER invent, modify, shorten, normalize, paraphrase, or partially copy a
`cite_key`.
- Write as continuous natural language rather than keywords, concept lists,
paper
titles, or multiple questions.
- Prefer clear, direct sentences. Do not use em dashes or semicolons.

**Return a JSON object:**
{
  "research_question": "a short paragraph describing the research context that
led to the research question, ending with exactly one research question",
  "key_inspiration": [
    {
      "cite_key": "the exact bibliography key",
      "rationale": "a dense paragraph explaining the selected paper's main
contribution, what idea from this work informed the target paper's central
contribution, how this idea influenced that contribution, and why this work
should be regarded as a primary source of inspiration."
    }
  ]
}

Return ONLY the JSON object, nothing else.
```

Source Paper
Hi Robot: Open-Ended Instruction Following with Hierarchical Vision-Language-Action Models (cs.RO)

Core Research Query
Robots working in human-centric settings receive language that is far messier than the atomic commands studied in language-conditioned imitation learning. A line of work trains policies to map instructions such as 'pick up the coke can' to actions and generalizes well within that regime, but real requests carry constraints, preferences, and mid-execution corrections ("that's not trash", "I'm allergic to pickles") whose meaning depends on what the robot is currently seeing and doing. A separate line of work uses LLMs and VLMs off the shelf to decompose long-horizon tasks over predefined or hand-designed skills. These systems parse richer language but inherit the limited dexterity of their skill libraries and rarely react to a user who interrupts mid-task. Both lines leave a gap between the semantics a system can interpret and the physical behavior it can actually produce, and it is not obvious whether that gap is best closed by making language interpretation more capable or by making dexterous learned policies more promptable. How can a robot with dexterous, learned manipulation skills interpret open-ended prompts and real-time human feedback, grounded in its own observations, and act on them while the task is underway?

Subfield-Specific Query 1
Vision-language-action models fine-tune pretrained VLMs to emit low-level continuous actions conditioned on images and a language instruction, and have shown strong dexterity and some language generalization on real hardware. However, their instruction supervision comes from demonstration annotations that are short and literal ("put the cup on the plate"), so it is unclear how much of their language-following ability survives when a prompt carries constraints, negations, preferences, or a correction issued partway through a rollout. What language can an end-to-end VLA actually be steered by at execution time, and where does its instruction-following break down as prompts depart from the literal commands seen in its training data?

Subfield-Specific Query 2
A line of work uses pretrained LLMs and VLMs to reason over a scene and break a long-horizon goal into a sequence of calls to predefined or hand-designed skills. These models bring broad semantic and world knowledge, enough to infer that a paper cup is trash while a reusable one is a dish, or that "something sweet" means candy, but they have never seen the robot's observations, its skill repertoire, or what its execution actually looks like mid-episode. What limits an off-the-shelf foundation model's ability to serve as a robot's high-level reasoner: can it stay grounded in the robot's own visual stream and affordances across a long rollout, and does its output remain something the low-level controller can physically carry out?

Subfield-Specific Query 3
A line of work already lets users correct a robot while it acts. Some approaches edit trajectories from verbal feedback, others train recovery behaviors from simulator-constructed perturbations, and others route spoken corrections through a language channel between a high-level and a low-level policy. Each demonstrates that mid-execution language can redirect behavior, but each is bounded: by a single task prompt, by the specific correction phrases human annotators happened to produce, or by feedback that never depends on what the robot is currently seeing. Utterances like "that's not trash" or "leave it alone" are unresolvable from text alone. They refer to whatever is in the gripper right now. What determines the range of corrections such a system can absorb, and can a robot ground feedback in its live observations while also handling phrasings and situations that never appeared in its interaction data?

Table 13: **Example query set for one research project.** The core research query describes the project's overarching goal, while subfield-specific queries isolate the distinct directions through which it is pursued.

Table 14: Primary arXiv category of the author-validated instances.

Primary category	Count	%
cs.LG (Machine Learning)	66	31.9%
cs.CV (Computer Vision)	46	22.2%
cs.CL (Computation and Language)	38	18.4%
cs.AI (Artificial Intelligence)	23	11.1%
cs.RO (Robotics)	20	9.7%
Other	14	6.8%

Table 15: Primary arXiv category of the 190,896 corpus papers.

Primary category	Count	%
cs.CV (Computer Vision)	64,919	34.0%
cs.LG (Machine Learning)	57,151	29.9%
cs.CL (Computation and Language)	33,255	17.4%
cs.RO (Robotics)	18,922	9.9%
cs.AI (Artificial Intelligence)	10,933	5.7%
Other arXiv category	4,116	2.2%
Non-arXiv (S2, OA)	1,600	0.8%

Table 16: Publication year of the 190,896 corpus papers.

Year	Count	%
Before 2020	1,783	0.9%
2020	28,570	15.0%
2021	34,894	18.3%
2022	33,956	17.8%
2023	42,091	22.0%
2024	47,029	24.6%
2025 or later	2,573	1.3%

Table 17: **Inspiration papers are not more similar to queries than hard negatives.** Mean BM25 lexical similarity and Qwen-3-8B semantic similarity between each query and its positive papers ($\mathcal{D}_q^+$) or topically similar hard negatives ($\mathcal{D}_q^-$).

	BM25 Sim.		Emb. Sim.	
Query	$\mathcal{D}_q^+$	$\mathcal{D}_q^-$	$\mathcal{D}_q^+$	$\mathcal{D}_q^-$
CoreQ	28.7	29.2	64.5	66.2
SubQ	34.5	34.4	65.1	64.2

CoreQ. `How can RL agents leverage the temporal dynamics of spiking neural networks when randomly initialized networks fail to collect sequences long enough to expose those dynamics?`

	CoreQ Positive	CoreQ Hard Negative
Cited	Elucidating the theoretical underpinnings of surrogate gradient learning in SNNs; Neuromorphic Attitude Estimation and Control	Evolving Connectivity for Recurrent Spiking Neural Networks; Deep RL with Spiking Q-learning
Uncited	Jump-Start Reinforcement Learning; Revisiting the Minimalist Approach to Offline RL	8 further topically adjacent papers, including surrogate-gradient variants and neuromorphic control work

Author rationale for uncited positives. `Jump-Start Reinforcement Learning suggests using a secondary controller to bridge the warmup period, enabling the spiking network to learn from the start. Revisiting the Minimalist Approach to Offline RL supports combining secondary-controller demonstrations with the spiking actor's own TD3 rollouts.`

Table 18: **Citation Status Does Not Identify Inspiration.** For a CoreQ about leveraging temporal SNN dynamics in RL, citation status is crossed with the author-provided positive/hard-negative judgment. The highlighted rationale explains why the uncited positives were useful.

Table 19: **Taxonomy of inspiration types.** Definitions describe how a source work shaped the target work.

Inspiration type	Definition
Method	A specific method, mechanism, component, formulation, or principle from the source work is adopted or adapted.
Generalization	An idea, finding, or method from the source work is extended to a new domain, modality, task, setting, or broader scope.
Limitation	A limitation, failure mode, or unresolved issue in the source work motivates the target work to address it.
Framework	The target work builds on the source work's overall framework, model, or system rather than a specific component.
Phenomenon	An empirical phenomenon or observation in the source work motivates the target work to explain, characterize, or formalize it.
Reframing	An insight from the source work provides a new perspective, abstraction, or interpretation of the target problem.
Motivation	The source work motivates the research problem or direction pursued in the target work. This label is used only when the motivation is not better captured by limitation, phenomenon, or evidence.
Evidence	The source work provides evidence or an independent finding that supports the plausibility or direction of the target work without being directly transferred.

Source Paper Towards Understanding the Mechanisms of Classifier-Free Guidance (cs.CV) (NeurIPS 2025 Spotlight)

Core Research Question
Classifier-free guidance (CFG) is a highly effective inference-time guidance method that significantly improves sample quality and condition alignment in diffusion models, but its underlying mechanisms remain poorly understood. Existing theoretical analyses mainly study isotropic Gaussian mixtures or one-dimensional distributions, where much of the complex structure of natural images is absent. This made us wonder (i) whether we could construct a simplified model in which the effects of CFG on image structures become directly visible, hence capturing the underlying mechanism of CFG more faithfully than the simplified settings considered in prior theoretical analyses and (ii) whether the mechanisms identified in this simplified setting extend beyond itself and can help explain, at least partially explain the CFG's mechanism in real-world diffusion models?

Key Inspiration Paper: Abid, Zhang, Bagaria & Zou (2018). Exploring patterns enriched in a dataset with contrastive principal component analysis. *Nature Communications* 9, 2134.

Rationale

Contribution: This paper shows that the eigendirections associated with differences between the covariance structures of two datasets can reveal features that are distinctive to one dataset relative to the other. These directions, referred to as contrastive principal components, characterize prominent structures that distinguish the target data from the background data.

New idea I brought into my own research: When analyzing CFG in our simplified linear model, I found that the guidance term involves the difference between two covariance matrices: the covariance of the conditional distribution and that of the unconditional distribution (CPC guidance component in equation (12) of my paper). This suggested that the effect of CFG might be understood through the structure of this covariance difference. I then searched for prior work studying such differences between covariance structures and came across contrastive PCA. This connection suggested that the relevant eigendirections could identify structures that are particularly prominent in the conditional distribution relative to the unconditional distribution, which we later show contribute to the CFG's effect.

Why I expect that idea to work: The mathematical structure arising in our analysis of CFG closely resembles that underlying contrastive PCA: both are governed by differences between the covariance structures of two distributions. This similarity suggested that the interpretation of contrastive principal components could provide useful insight into how CFG emphasizes structures that distinguish the conditional distribution from the unconditional one.

Table 21: **Inspiration can cross research domains.** In the above example, contrastive PCA supplies the covariance-difference perspective used to analyze classifier-free guidance.

Key Inspiration Paper: Self-Consistency Improves Chain of Thought Reasoning in Language Models (cs.CL, arxiv:2203.11171)
Source Paper: Benefits and Limitations of Communication in Multi-Agent Reasoning (cs.MA, arxiv:2510.13903)
Role: naive baseline. Proposes multiple CoT with a voting mechanism at the end to improve performance. This work seemed to be the perfect naive baseline as the simplest "multi-agent" protocol where multiple agents think and do not communicate during their thinking process. Here all agents have all the context and think the full CoT, in contrast with multi-agent approaches where context is chunked and CoT is distributed. We used this as a baseline in our empirical experiments section. We expected this to be a good naive baseline as it does not adaptively allocate resources and models are given full context. These were the two main levers by which we conceived multi-agent reasoning to be superior.
Source Paper Self-Consistency for LLM-Based Motion Trajectory Generation and Verification (cs.CV, arxiv:2603.29301)
Role: generalization to a new modality. The key work that our approach extends from, we bring it to the visual domain, whereas this work only works for language-based tasks.
Source Paper Accelerated Test-Time Scaling with Model-Free Speculative Sampling (cs.CL, arxiv:2506.04708)
Role: structural premise to exploit. This paper proposes parallel test-time scaling, which is one of the core bases of our work. We aim to accelerate test-time scaling, along the setting that the given paper suggested. We expected the idea to work because speculative decoding is already known to accelerate decoding, and we found redundancy to leverage.

Table 22: **A single paper can inspire different projects in different ways.** In the above example, Self-Consistency serves as a baseline, a cross-modal generalization, and a systems premise across three source papers.

Table 23: **Retriever configurations used in SCHOLARCATALYST.** Configurations are reported from the official model documentation and checkpoint configuration files. Maximum lengths denote the reported input limits or recommended retrieval caps of each checkpoint or API. "Instruction" indicates whether we prepend the task-specific instruction to queries.

Model	Size	Arch.	Max.$\|Q\|$	Max.$\|D\|$	Instr.	Link
Sparse retriever						
`BM25` (Robertson & Zaragoza, 2009)	—	Sparse	∞	∞	No	
Multi-vector retriever						
`LateOn` (Sourty et al., 2026)	149M	ColBERT	32	300	No	
Scientific embedding retrievers						
`SPECTER2` (Singh et al., 2022)	110M	Encoder	512	512	No	
`SciNCL` (Ostendorff et al., 2022)	110M	Encoder	512	512	No	
`OpenScholar` (Asai et al., 2024)	110M	Encoder	512	512	No	
General-purpose embedding retrievers						
`Qwen3-Embedding-4B` (Zhang et al., 2025b)	4B	Decoder	32K	32K	Yes	
`Qwen3-Embedding-8B` (Zhang et al., 2025b)	8B	Decoder	32K	32K	Yes	
`Gemini Embedding 2` (Shanbhogue et al., 2026)	—	Encoder	8,192	8,192	No	
`GTE-Qwen2-7B-Instruct` (Li et al., 2023)	7B	Bi-decoder	32K	32K	No	
Reasoning-optimized embedding retrievers						
`Reason-Embed-Qwen3-8B` (Chen et al., 2025)	8B	Decoder	512	512	No	
`Inf-Retriever-v1-Pro` (Yao et al., 2025)	7B	Decoder	8,192	8,192	No	
`NV-Embed-Reasoning-3B` (NVIDIA, 2026)	3B	Bi-decoder	8,192	8,192	No	
`Diver-Retriever-4B` (Long et al., 2025)	4B	Decoder	40K	40K	No	

Table 24: **Language-model backbones used in SCHOLARCATALYST.** GPT-4.1 and o3 run under the knowledge-cutoff rule of Section 4.1. The other rows serve the reranking, backbone-swap and leakage analyses, and Claude Fable 5.1 also appears in the rows of Table 6 marked ‡.

Model	Knowledge Cutoff
Under the knowledge-cutoff rule	
`GPT-4.1` (OpenAI, 2026a)	June 2024
`o3` (OpenAI, 2026e)	June 2024
Analysis backbones	
`Gemini 2.5 Flash` (Comanici et al., 2025)	January 2025
`Gemini 3.1 Pro` (Google DeepMind, 2026a)	January 2025
`GPT-5.6 Luna` (OpenAI, 2026b)	February 2026
`GPT-5.6 Terra` (OpenAI, 2026d)	February 2026
`GPT-5.6 Sol` (OpenAI, 2026c)	February 2026
`Gemini 3.7 Flash` (Google DeepMind, 2026c)	March 2026
`Claude Fable 5.1` (Anthropic, 2026)	June 2026

Table 25: **Effective searchable corpus size per query after the temporal cutoff.**

	n	Mean	Median	Min	Max
Full corpus (no cutoff)			190,896		
CoreQ (cutoff)	207	190,306	190,574	188,526	190,896
SubQ (cutoff)	687	190,300	190,574	188,526	190,896
All queries (cutoff)	894	190,302	190,574	188,526	190,896

Table 26: **Extended results for embedding retrievers on SCHOLARCATALYST.** Recall@k (k=5, 20, 100) and nDCG@20 across both query types. This table includes all embedding retrievers evaluated in our experiments.

	Core Research Query				Subfield-specific Query			
Model	nDCG@20	R@5	R@20	R@100	nDCG@20	R@5	R@20	R@100
Sparse retriever								
BM25	0.16	0.12	0.23	0.39	0.21	0.12	0.26	0.45
Multi-vector retriever								
LateOn 0.1B	0.16	0.12	0.24	0.37	0.27	0.16	0.33	0.55
Scientific embedding retrievers								
SPECTER2	0.13	0.10	0.20	0.38	0.18	0.11	0.23	0.42
SciNCL	0.13	0.10	0.20	0.40	0.18	0.10	0.22	0.42
OpenScholar	0.16	0.13	0.23	0.40	0.19	0.12	0.24	0.46
General-purpose embedding retrievers								
GTE-Qwen2-7B	0.23	0.17	0.32	0.51	0.30	0.18	0.38	0.62
Qwen3-Emb-4B	0.25	0.18	**0.39**	0.59	0.35	0.21	0.44	0.69
Qwen3-Emb-8B	0.26	**0.21**	0.37	0.58	**0.41**	**0.25**	**0.51**	**0.73**
Gemini-Emb-2	**0.27**	**0.21**	0.38	**0.62**	0.38	0.24	0.46	0.70
Reasoning-optimized embedding retrievers								
NV-Embed-Reasoning-3B	0.23	0.18	0.32	0.51	0.32	0.20	0.40	0.63
Diver-Retriever-4B	0.26	0.19	0.36	0.56	0.37	0.23	0.45	0.70
Reason-Embed-Qwen3-8B	0.23	0.18	0.33	0.53	0.33	0.19	0.40	0.65
Inf-Retriever-v1-Pro	0.26	0.20	0.37	0.53	0.31	0.19	0.38	0.63

Table 27: **Reranking performance across candidate-pool sizes.** Recall@5 on 50 queries of each type after reranking Gemini-Emb-2's top-K candidates. Bold and underlining indicate the highest and second-highest values, respectively, within each condition and column.

		Core Research Query				Subfield-specific Query			
Verifier	Condition	K=20	K=40	K=80	K=160	K=20	K=40	K=80	K=160
GPT-4.1	Standard	0.27	0.29	0.26	0.28	0.18	0.17	0.15	0.16
	Oracle	0.38	0.32	0.29	0.29	0.26	0.20	0.18	0.17
Gemini 2.5 Flash	Standard	0.22	0.24	0.20	0.24	0.14	0.14	0.15	0.15
	Oracle	0.36	0.30	0.30	0.24	0.26	0.23	0.18	0.14
Gemini 3.1 Pro	Standard	0.31	0.30	0.27	0.30	0.18	0.19	0.19	0.21
	Oracle	0.47	0.43	0.32	0.35	0.30	0.23	0.21	0.22
Gemini 3.7 Flash	Standard	**0.33**	0.32	0.33	0.33	0.20	0.21	0.22	0.21
	Oracle	0.51	0.42	0.39	0.33	0.32	0.27	0.23	0.22
GPT-5.6 Sol	Standard	0.26	0.29	0.28	0.32	0.19	0.20	0.19	0.18
	Oracle	0.47	0.39	0.38	0.34	0.33	0.26	0.21	0.20
Claude Fable	Standard	0.31	**0.34**	**0.35**	**0.37**	**0.22**	**0.25**	**0.25**	**0.26**
	Oracle	**0.60**	**0.51**	**0.50**	**0.47**	**0.40**	**0.33**	**0.29**	**0.26**

Table 31: **GPT-5.6 Sol and Claude Fable 5.1 exceed Gemini-Emb-2 at CoreQ R@20 on both cutoff slices.** CoreQ Recall@k (k=5, 20, 100) and trajectory recall for the tool-calling agent with five backbones and their matching closed-book conditions. Gemini-Emb-2 is the retriever reference within each slice. Closed-book rows receive the source paper's title and abstract and name papers from memory. Agent rows receive the query only. Parentheses indicate vendor-stated knowledge cutoffs.

	Pre-cutoff slice				Post-cutoff slice			
System	R@5	R@20	R@100	Traj.	R@5	R@20	R@100	Traj.
Gemini-Emb-2 retriever	0.23	0.35	0.49		0.25	0.41	0.63	
Tool-calling agent								
GPT-4.1 (June 2024)	0.20	0.32	0.44	0.33	0.25	0.39	0.60	0.42
Gemini 3.1 Pro (January 2025)	0.21	0.36	0.47	0.40	0.26	0.37	0.53	0.45
GPT-5.6 Sol (February 2026)	0.21	0.42	0.50	0.49	0.25	0.47	0.64	0.58
Gemini 3.7 Flash (March 2026)	0.29	0.37	0.49	0.38	0.28	0.41	0.59	0.41
Claude Fable 5.1 (June 2026)	0.36	0.42	0.60	0.64	0.26	0.47	0.66	0.63
Closed-book, given the source title and abstract								
GPT-4.1	0.07	0.14	0.57	0.14	0.05	0.12	0.64	0.15
Gemini 3.1 Pro	0.20	0.29	0.61	0.31	0.18	0.26	0.72	0.26
GPT-5.6 Sol	0.30	0.40	0.64	0.40	0.40	0.48	0.81	0.51
Gemini 3.7 Flash	0.25	0.34	0.66	0.34	0.20	0.29	0.72	0.28
Claude Fable 5.1	0.36	0.55	0.66	0.56	0.44	0.54	0.80	0.55

Table 32: **Source-paper context improves CoreQ retrieval, whereas web search changes little.** CoreQ Recall@k (k=5, 20, 100) and trajectory recall of the Gemini 3.7 Flash tool-calling agent on the two cutoff slices. The local-corpus row is the main protocol. The remaining rows add the source paper's title and abstract, web search, or mapped source-bibliography entries. The bibliography-alone row ranks the same entries in citation order without a model.

	Pre-cutoff slice				Post-cutoff slice			
Condition	R@5	R@20	R@100	Traj.	R@5	R@20	R@100	Traj.
Local corpus only	0.29	0.37	0.49	0.38	0.28	0.41	0.59	0.41
+ source paper (title and abstract)	0.34	0.48	0.57	0.51	0.34	0.52	0.69	0.56
+ web search	0.29	0.34	0.48	0.37	0.28	0.42	0.61	0.44
+ source-paper bibliography	0.36	0.68	0.93	0.93	0.42	0.81	0.98	0.98
Source-paper bibliography alone	0.23	0.60	0.93	0.93	0.18	0.60	0.98	0.98

Table 33: **Generated queries for a CoreQ instance.** We show the original question, the single-query expansion, and five multi-query outputs.

Original query	Enhancing the reasoning capabilities of large language models traditionally involves supervised fine-tuning followed by reinforcement learning applied to reasoning-specific datasets. Recent advancements have shown that applying reinforcement learning directly to base models can autonomously elicit sophisticated reasoning behaviors without relying on supervised fine-tuning data. However, these methods still depend on rule-based verifiers, verified golden answers, or pre-trained reward models to generate reward signals during the reinforcement learning phase. Relying on such explicit feedback limits scalability in domains where ground-truth verification is costly or unavailable. Can we enhance the reasoning capability of LLMs in a fully unsupervised manner?
Single-query expansion	Enhancing the reasoning capabilities of large language models traditionally involves supervised fine-tuning followed by reinforcement learning applied to reasoning-specific datasets. Recent advancements have shown that applying reinforcement learning directly to base models can autonomously elicit sophisticated reasoning behaviors without relying on supervised fine-tuning data. However, these methods still depend on rule-based verifiers, verified golden answers, or pre-trained reward models to generate reward signals during the reinforcement learning phase. Relying on such explicit feedback limits scalability in domains where ground-truth verification is costly or unavailable. Can we enhance the reasoning capability of LLMs in a fully unsupervised manner? I am particularly interested in prior work exploring self-supervised objectives, intrinsic motivation, or unsupervised reward shaping techniques for language models, as well as approaches that leverage emergent properties or self-consistency signals to improve reasoning without external supervision.
Multi-query generation	(1) Enhancing the reasoning capability of LLMs in a fully unsupervised manner? I am particularly interested in approaches that leverage self-supervised objectives or intrinsic motivation signals to guide reasoning skill acquisition without any external reward models. (2) Enhancing the reasoning capability of LLMs in a fully unsupervised manner? I would like to explore whether emergent reasoning behaviors can be fostered through unsupervised pre-training on synthetic or automatically generated reasoning tasks. (3) Enhancing the reasoning capability of LLMs in a fully unsupervised manner? I am curious about prior work that investigates the use of mutual information maximization or contrastive learning techniques to improve reasoning without labeled data. (4) Enhancing the reasoning capability of LLMs in a fully unsupervised manner? I am seeking studies that examine the role of curriculum learning or self-play in enabling models to develop reasoning skills autonomously. (5) Enhancing the reasoning capability of LLMs in a fully unsupervised manner? I am interested in research that analyzes the emergence of reasoning abilities in large-scale language models trained solely on raw text corpora, without any explicit reasoning supervision.

Table 34: **HyDE-generated hypothetical abstracts.** We show three of the five abstracts generated for the same **CoreQ** instance as Table 33.

Original query	Enhancing the reasoning capabilities of large language models traditionally involves supervised fine-tuning followed by reinforcement learning applied to reasoning-specific datasets. Recent advancements have shown that applying reinforcement learning directly to base models can autonomously elicit sophisticated reasoning behaviors without relying on supervised fine-tuning data. However, these methods still depend on rule-based verifiers, verified golden answers, or pre-trained reward models to generate reward signals during the reinforcement learning phase. Relying on such explicit feedback limits scalability in domains where ground-truth verification is costly or unavailable. Can we enhance the reasoning capability of LLMs in a fully unsupervised manner?
Hypothetical abstract 1	Emergent Reasoning in Language Models via Self-Consistency Training. Recent work has demonstrated that large language models can exhibit improved reasoning abilities when trained to produce self-consistent outputs. In this study, we investigate a self-consistency training paradigm where models are encouraged to generate multiple solutions to a given problem and are rewarded for producing answers that agree with the majority of their own outputs. We apply this approach to arithmetic and commonsense reasoning tasks, showing that self-consistency training leads to modest improvements in accuracy without requiring external supervision or gold-standard answers. Our analysis suggests that self-consistency acts as a form of internal regularization, promoting the emergence of more robust reasoning patterns. These findings highlight the potential of leveraging a model's own outputs as a weak supervisory signal for enhancing reasoning capabilities.
Hypothetical abstract 2	Unsupervised Skill Acquisition in Reinforcement Learning via Intrinsic Motivation. Traditional reinforcement learning relies on extrinsic rewards provided by the environment, which can be sparse or unavailable in many real-world scenarios. We propose an unsupervised RL framework where agents are driven by intrinsic motivation signals, such as curiosity or novelty, to autonomously acquire diverse skills. Our method employs a self-supervised objective that rewards agents for exploring novel states and behaviors, without access to task-specific rewards or external supervision. Experiments in simulated environments demonstrate that agents trained with intrinsic motivation develop a repertoire of complex behaviors, including multi-step reasoning and problem-solving strategies. This work suggests that unsupervised RL can serve as a foundation for skill acquisition in domains where explicit reward signals are difficult to define.
Hypothetical abstract 3	Bootstrapping Commonsense Knowledge in Neural Language Models via Iterative Self-Labeling. Acquiring commonsense knowledge remains a challenge for neural language models, particularly in the absence of large-scale annotated datasets. We introduce an iterative self-labeling approach in which a language model generates candidate answers to commonsense questions and then refines its predictions by training on its own high-confidence outputs. Our method leverages confidence estimation to select pseudo-labels, enabling the model to bootstrap its knowledge without external supervision. We evaluate our approach on several commonsense reasoning benchmarks and observe consistent improvements over baseline models trained without self-labeling. Our results indicate that iterative self-labeling can be an effective strategy for enhancing the reasoning abilities of language models in a data-efficient and unsupervised manner.

Table 35: **Reading beyond abstracts changes agent recall little.** Recall@k (k=5, 20, 100), nDCG@20, and trajectory recall for GPT-4.1 with Gemini-Emb-2 search on 50 **CoreQ** and 50 **SubQ** from the 180,399-paper full-text subset. The table gives the full results behind Figure 36.

	Core Research Query				Subfield-specific Query				
Read policy	nDCG@20	R@5	R@20	R@100	nDCG@20	R@5	R@20	R@100	Traj.
Gemini-Emb-2 retriever	0.25	0.19	0.35	0.61	0.43	0.27	0.55	0.79	
Abstract only (no read tool)	0.25	0.22	0.35	0.58	0.41	0.32	0.46	0.76	0.37 / 0.52
Abstract + intro	0.26	0.22	0.34	0.61	0.41	0.29	0.48	0.78	0.36 / 0.51
Abstract + method	0.24	0.22	0.32	0.61	0.41	0.28	0.47	0.78	0.33 / 0.53
Abstract + bib	0.26	0.23	0.34	0.61	0.41	0.30	0.48	0.79	0.36 / 0.52
Abstract + intro + bib	0.25	0.23	0.34	0.60	0.42	0.30	0.49	0.77	0.36 / 0.53
Full paper	0.25	0.23	0.34	0.60	0.42	0.29	0.49	0.78	0.35 / 0.51
Sections chosen by the agent	0.26	0.23	0.35	0.60	0.42	0.30	0.50	0.77	0.36 / 0.53

Table 37: **Agents rarely recover positives beyond the retriever's pool.** Distribution of the 4,845 author-validated positives across 894 queries by whether Gemini-Emb-2 retrieves them in its top 100 and whether each agent ranks them in its top 20. Each of the first four columns reports the percentage of all positives in the corresponding retriever-agent outcome cell. The final two columns report the agent top-20 rate conditional on whether the retriever places the positive in its top 100.

	All positives (%)				**Agent top-20 rate (%)**	
	Retriever top 100		Outside top 100		Retriever top 100	Outside top 100
Agent	Top 20	Missed	Top 20	Missed	Top 20	Top 20
Tool-calling agent (GPT-4.1)	40%	30%	1%	29%	57%	4%
Deep research agent (o3)	34%	36%	2%	28%	48%	6%
Grep agent (GPT-4.1)	6%	64%	1%	29%	8%	4%

Table 41: **Agents usually explain a positive paper the way its author does.** For each core research query positive in the tool-calling agent's top 20, GPT-4.1 explains how the paper could help pursue the query, and a GPT-4.1 judge compares that explanation with the author's rationale. Of the 219 comparable pairs, 76% describe the same relationship, and almost all others give a different relationship that is still valid. Two examples below are judged the same and one different but valid. The judge's explanation is in gray. Bolding within the rationales is ours.

Verdict	Author's rationale	Backbone's rationale (GPT-4.1)
TTRL: Test-Time Reinforcement Learning		
Same relationship	This is the most significant paper that impacted my initial experiments. I was following their setup and used their code to run test-time training with unsupervised labels, but I found the **correctness of the labels seems to not matter** too much on the Qwen-Math models they used. [...]	The TTRL paper's main contribution is the demonstration that reinforcement learning can be effectively performed on reasoning tasks in large language models **using only unlabeled data**, by leveraging majority-voted answers as a proxy for ground-truth rewards. [...]
Judge: Both rationales identify that TTRL demonstrates the effectiveness of using majority-voted, potentially noisy, unsupervised labels as rewards for reinforcement learning in LLMs, directly informing the author's [...]		
Consent in Crisis: The Rapid Decline of the AI Data Commons		
Same relationship	The paper did a large-scale, **longitudinal audit** of the consent protocols for the web domains underlying AI training corpora, and pointed out there is a decline in data owners' willingness to contribute data to AI training. This paper made an important observation [...]	The key contribution of "Consent in Crisis: The Rapid Decline of the AI Data Commons" is its rigorous, **longitudinal audit** quantifying how quickly and extensively web data sources are restricting AI training use, particularly [...]
Judge: Both rationales identify the paper's empirical audit of declining data consent as foundational for understanding the real-world consequences of respecting opt-outs on LLM training, emphasizing the importance of the [...]		
DOME: Taming Diffusion Model into High-Fidelity Controllable Occupancy World Model		
Different, valid	One of the existing sota methods, with high reconstruction accuracy but **severe degradation of future prediction** performance, inspired us to think about the limitations of autoregressive correlation design.	The DOME paper's primary contribution is the introduction of a **spatial-temporal diffusion transformer** that directly models 3D occupancy evolution over time, bypassing the limitations of discrete tokenization and rigid geometric priors [...]
Judge: The author focuses on DOME's limitations in autoregressive design for future prediction, while the model highlights DOME's diffusion-based architecture and its advantages over tokenization and priors, both valid but [...]		

Table 42: **Search effort, cost, and latency of agent baselines.** All numerical columns except total cost report per-query values. Tool and backbone calls are means, while time is the median. Total cost is for all 894 queries. Costs use list prices and include prompt caching for Claude Fable 5.1. A tool call is a search, read, or shell command. Backbone calls include the tool-calling agent's per-paper scoring calls.

Agent	Backbone	Tool calls	Backbone calls	Cost ($)	Total cost ($)	Median time (s)
Grep	GPT-4.1	20.3	18.5	0.37	332	22
Tool-calling	GPT-4.1	5.8	44.3	0.05	42	25
Deep research	o3	14.3	18.3	0.39	345	60
Grep	Claude Fable 5.1	19.3	20.2	1.14	1,016	139
Tool-calling	Claude Fable 5.1	13.9	63.2	1.32	1,178	88
Deep research	Claude Fable 5.1	22.7	12.4	1.39	1,240	146

Table 43: Author validation results, computed on the final 207-paper, 687-subfield-specific query benchmark. *Reconstructed query quality* is rated per **CoreQ** or **SubQ**. *Candidate pool judgement* is rated per individual candidate document the author judged for an **SubQ**. *Key-inspiration selection vs. author* is evaluated only for **CoreQ**s, treating the pipeline's per-paper nomination as a binary prediction over every cited candidate the author judged for that paper.

Category	Metric	Value
Reconstructed query quality	**CoreQ** rated fully accurate	60.9%
	CoreQ rated accurate or mostly accurate	98.1%
	SubQ rated fully accurate	64.9%
	SubQ rated accurate or mostly accurate	95.9%
Candidate pool judgement	Original citation retained as positive	85.1%
	Pooled candidate promoted to positive	35.1%
Key-inspiration selection vs. author	Precision	58.2%
	Recall	26.0%
	Pipeline's full nominated set exactly equals the author's	9.2%

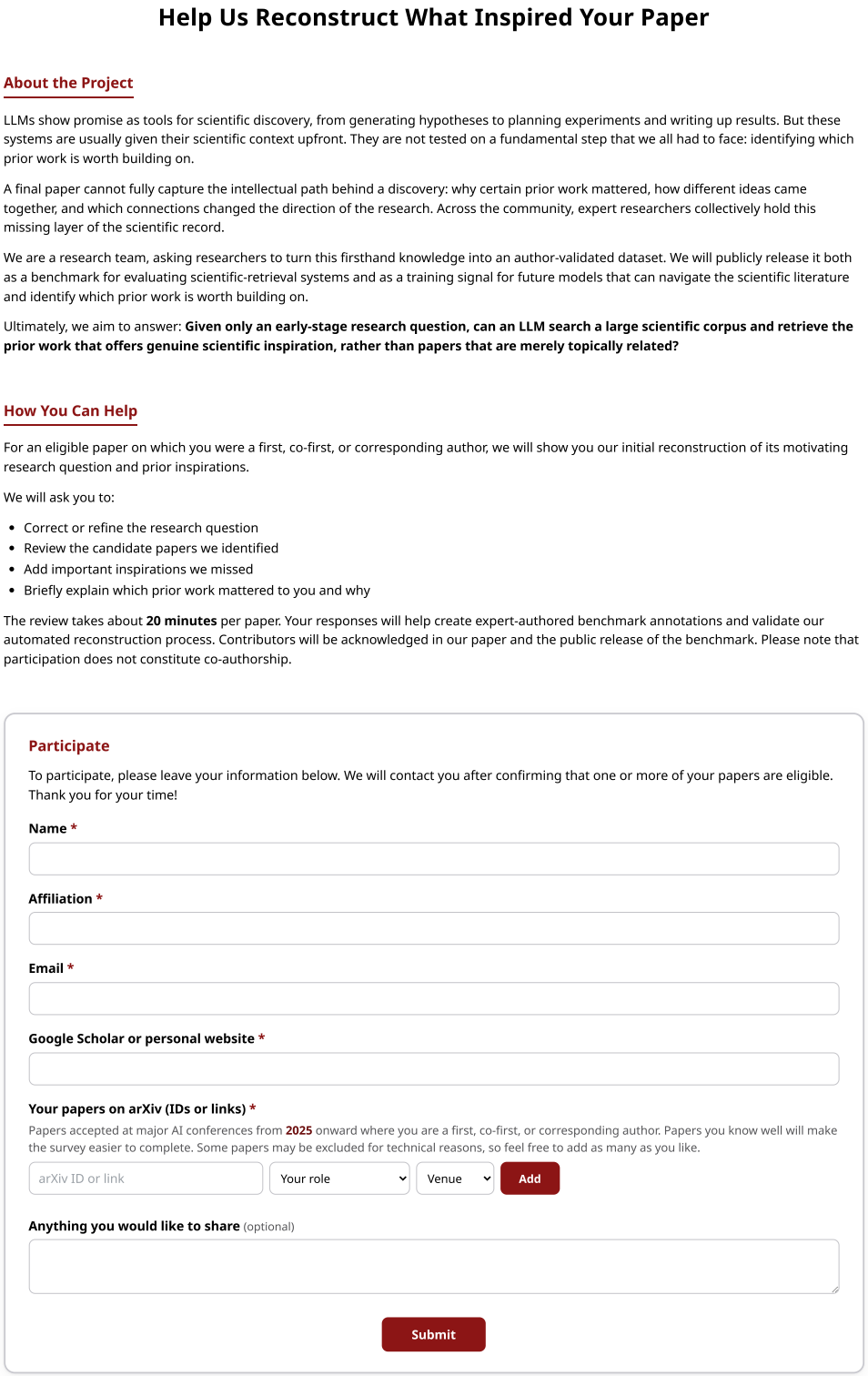


Figure 44: Recruitment page shown to participants, where eligible authors submit their arXiv papers accepted at major AI conferences from 2025 onward.

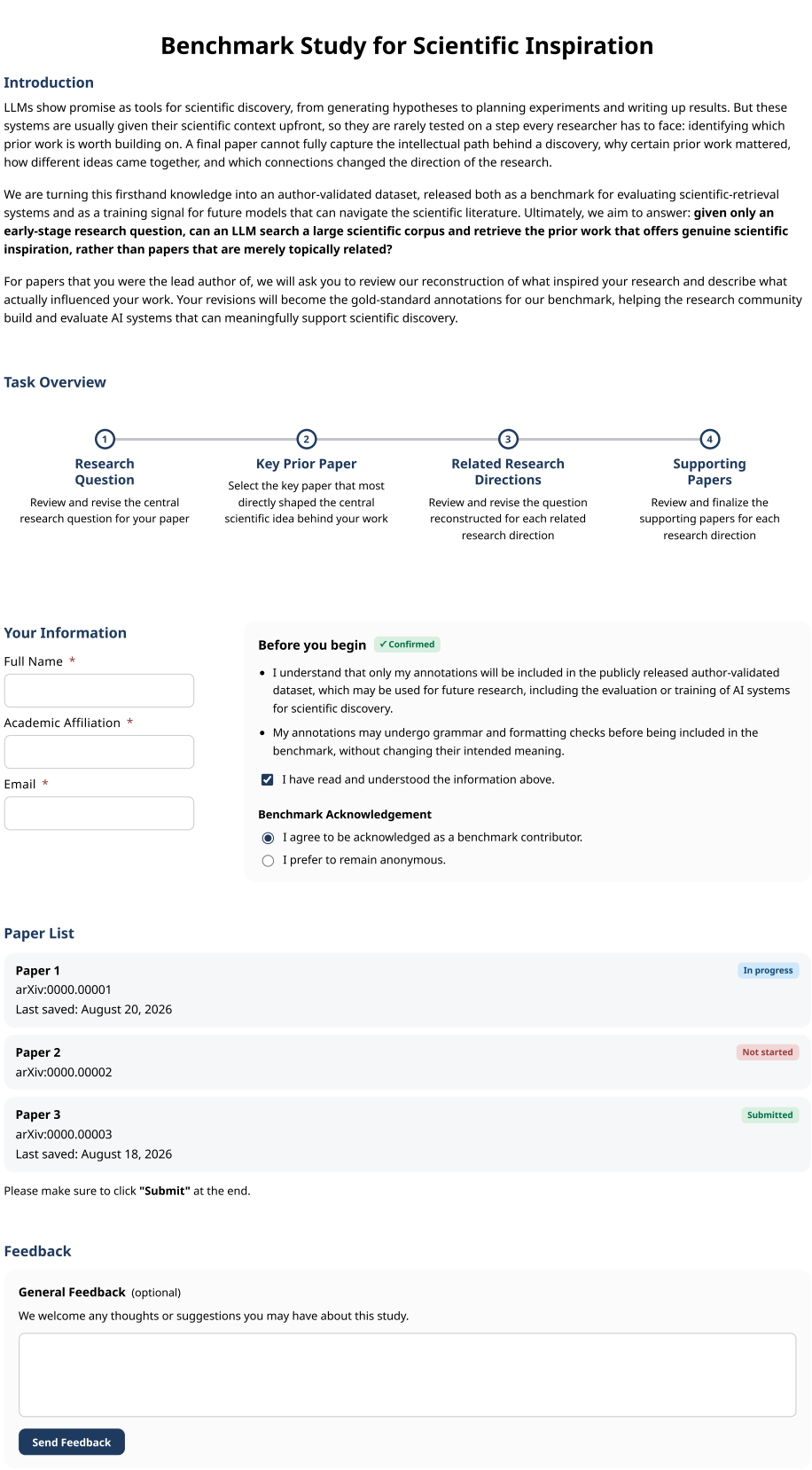


Figure 45: The web-based annotation interface presented to authors.

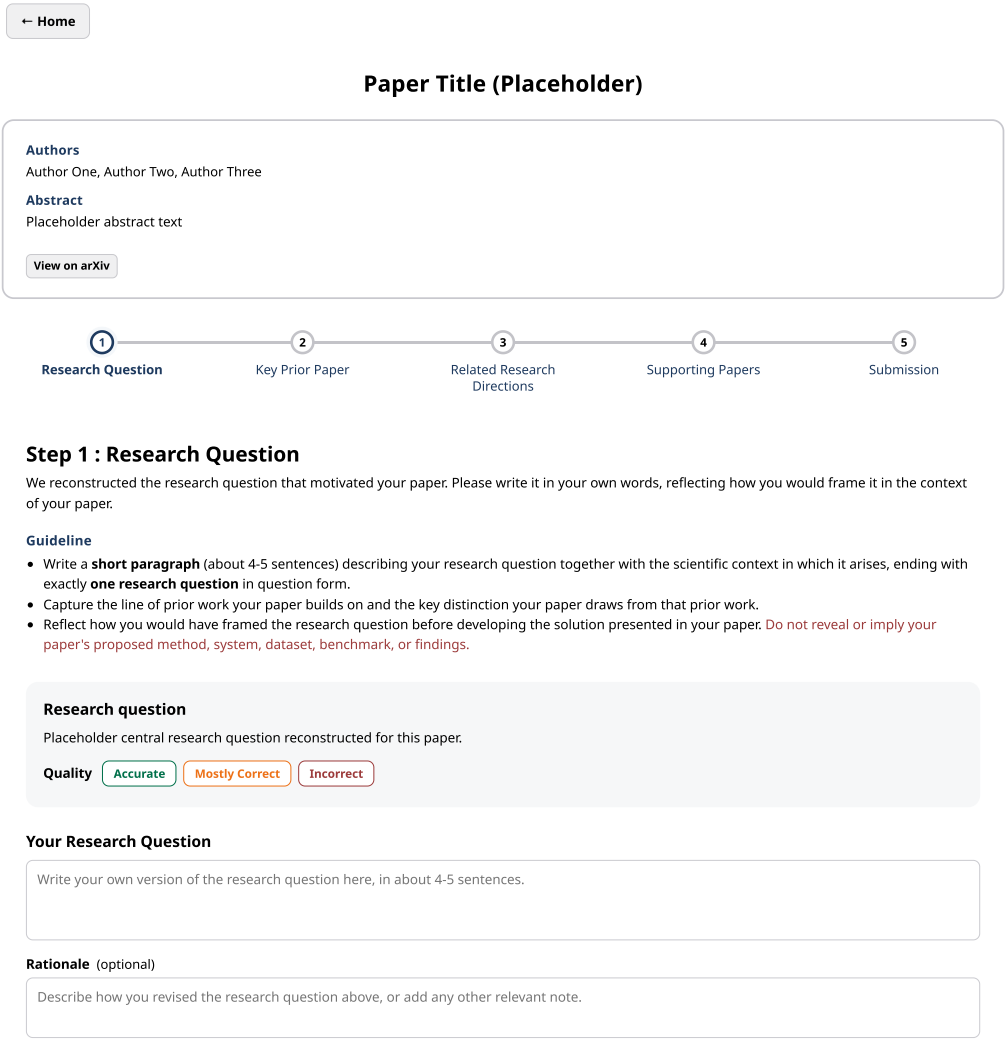


Figure 46: Author annotation, Step 1 (Research Question). Authors rate our reconstructed research question, rewrite it in their own words to reflect their pre-solution framing, and optionally add a rationale.

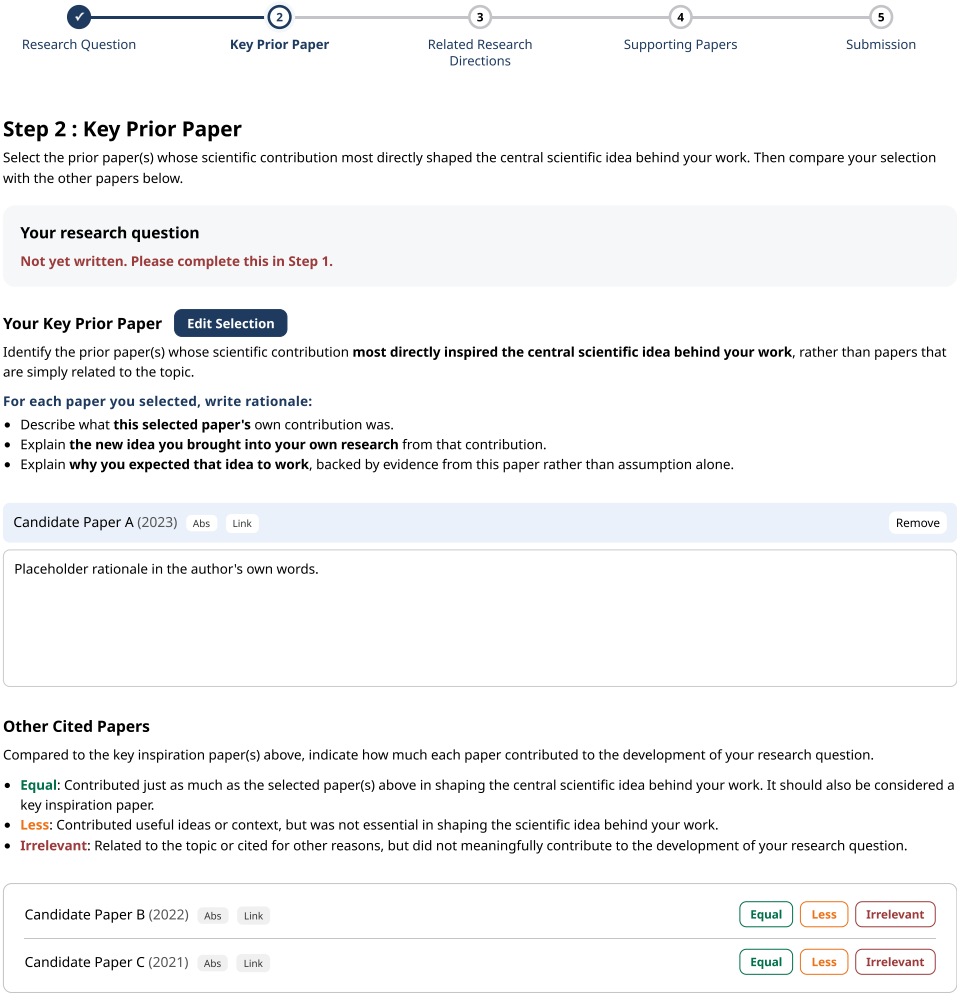


Figure 47: Author annotation, Step 2 (Key Prior Paper). Authors select the prior paper(s) whose contribution most directly shaped the central idea, write a rationale for each, and mark every other cited paper as an equal, lesser, or irrelevant influence.

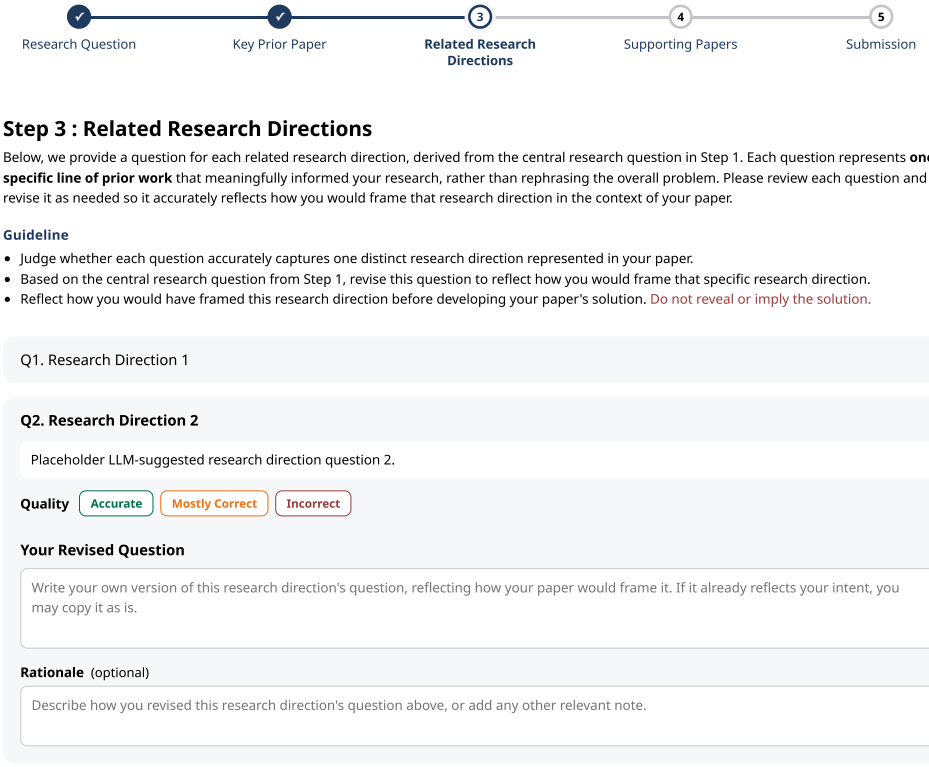


Figure 48: Author annotation, Step 3 (Related Research Directions). For each reconstructed research direction, authors rate its question, revise it to reflect how their paper frames that direction, and optionally add a rationale.

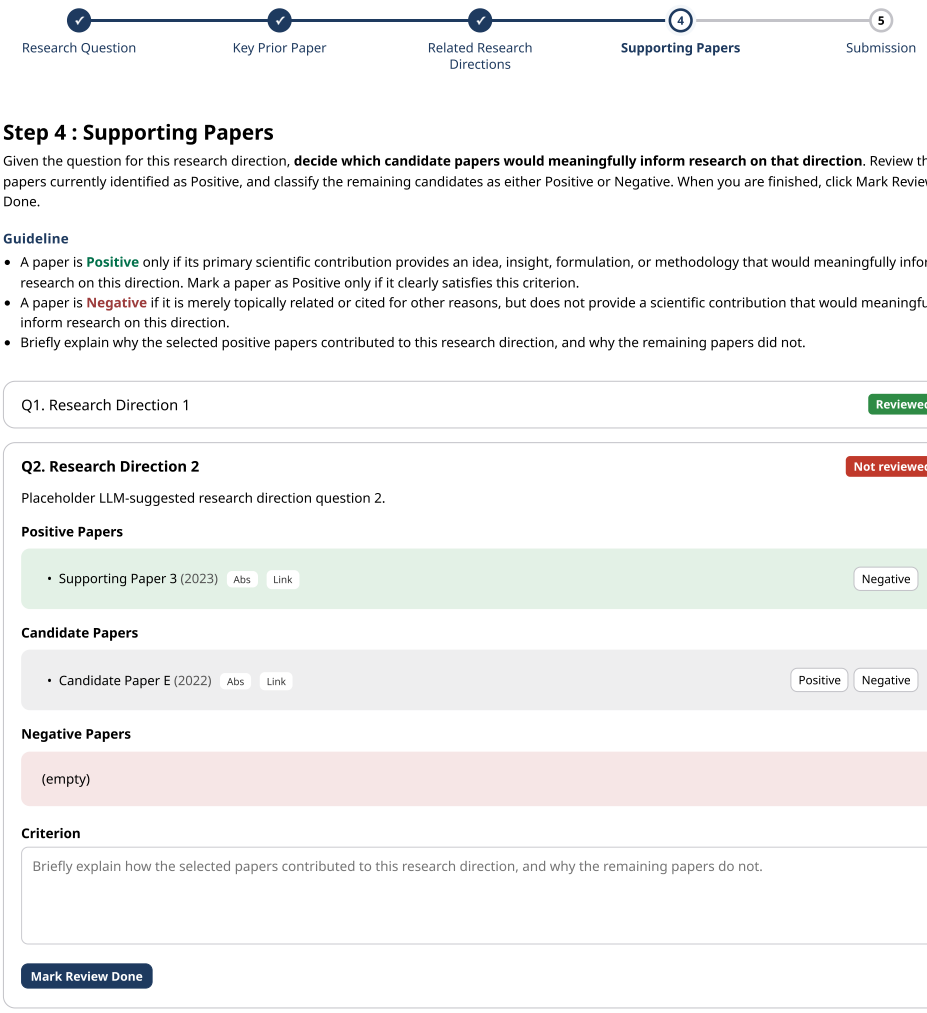


Figure 49: Author annotation, Step 4 (Supporting Papers). For each research direction, authors classify candidate papers as positive or negative according to whether the paper's contribution would meaningfully inform that direction, with a brief justification.

Prompt for Identifying Key Subfields

```
You are an expert research analyst. Given a research paper, identify the
distinct
inspiration threads underlying the paper and formulate a retrieval query for
each
identified thread.

<paper>
{paper_text}
</paper>

**Task**
Identify between 3 and 5 distinct inspiration threads reflected in how the
paper
discusses, organizes, and builds upon prior work.

An inspiration thread is one coherent line of prior work representing a
distinct
scientific direction that informs the development of the paper. Different
inspiration threads should correspond to distinct lines of prior work at a
comparable level of abstraction, rather than broad research directions and
their
finer grained subdirections.

For each inspiration thread:

1. Inspiration Thread Identification
- Identify the distinct inspiration threads reflected in the paper's discussion
of
prior work and motivation (e.g., the Introduction, Related Work, and other
relevant sections).
- Each inspiration thread should correspond to one coherent line of prior work
representing a distinct scientific direction.
- Define all inspiration threads at a comparable level of abstraction. Avoid
creating separate threads for closely related subdirections within the same
line
of prior work.
- Do not merge distinct lines of prior work simply because they belong to the
same
broader research area. Likewise, do not split a single line of prior work into
multiple threads because of method variants or incremental improvements.
- Assign a short, canonical name that reflects the shared scientific direction
represented by the thread. Prefer naming the underlying scientific direction
over
a representative paper, specific method, or implementation.

2. Thread Description
- Describe the scientific direction represented by the thread, the shared
scientific insight unifying the prior work belonging to it, and how this
direction
contributes to the development of the target paper.
- Focus on the scientific direction and shared scientific insight rather than
paper-specific methods, techniques, terminology, or implementations introduced
by
individual papers.
- Clearly distinguish this thread from the other inspiration threads by
explaining
what makes its scientific direction unique.

3. Thread-Specific Retrieval Query
- Generate a retrieval query by specializing the source paper's research
question
toward the scientific direction represented by this inspiration thread. The
query
```

```
should reflect how the authors would investigate this perspective in the
context
of their own research problem.
- The query should reflect how the authors framed this scientific perspective
within their overall research problem before developing the solution.
- Briefly describe the line of prior work represented by this perspective, then
end
with exactly one research question.
- The query should be specific enough to retrieve the prior work that most
substantially contributed to the target paper from this perspective, rather
than
broadly retrieving related work.
- Write the retrieval query as continuous natural language rather than
keywords,
concept lists, paper titles, or Boolean style search queries.
- Do not reveal or imply the paper's eventual solution, proposed paradigm, or
high-level approach.
- Do not mention methods, systems, datasets, benchmarks, coined terminology, or
other solution-specific concepts introduced by the paper itself.

**CRITICAL:**
- Write as continuous natural language rather than keywords, concept lists,
paper
titles, or multiple questions.
- Prefer clear, direct sentences. Do not use em dashes or semicolons.

**Return a JSON object with one field "threads" containing a list of thread
objects.**

**Format**
{
  "threads": [
    {
      "thread_name": "short, canonical name of the scientific direction
represented by this inspiration thread",
      "description": "the scientific direction represented by this inspiration
thread, the shared scientific insight represented by the thread, and the
scientific role this direction serves in relation to the paper's contribution",
      "query": "natural-language retrieval query for retrieving papers from
      this
inspiration thread"
    }
  ]
}
```

Prompt for Extracting Subfield-specific Inspiration

```
You are an expert research analyst.
Given a research paper, its bibliography, and a set of pre-identified
inspiration
threads, identify the key cited papers in the bibliography for each inspiration
thread.

<paper>
{paper_text}
</paper>

<bibliography>
{bibliography}
</bibliography>

<threads>
{threads_block}
```

```
</threads>

**Task**
For each inspiration thread, identify the cited papers that contributed most
substantially to the development of the target paper from the scientific
perspective represented by that thread.

**Guidelines:**
For each inspiration thread:
1. Key Inspiration Paper Selection
- Select only papers whose primary scientific contributions were essential to
the
development of the target paper within this perspective.
- Select at most 5 papers for this thread, choosing the ones whose primary
scientific contributions were most essential to the development of the target
paper within this perspective.
- Use the paper's discussion of prior work as the primary evidence for
selecting
key papers for inspiration threads.
- Avoid selecting a paper merely because it shares the thread's general topic.
Select it only if its specific contribution was essential to the target paper's
approach within this thread.
- Do not select papers that are cited primarily for background, comparison,
datasets, benchmarks, or implementation details unless they also made a core
scientific contribution within this perspective.
- Use the inline citation markers (e.g., [CITE:key]) together with the
surrounding
discussion to determine the scientific role of each cited paper.

2. Rationale
- Explain the shared scientific contribution represented by the selected
papers.
- Explain how this contribution was interpreted, adapted, extended, or
transformed
within the target paper.
- Explain why this shared contribution makes these papers the most important
prior
work from this scientific perspective, compared with other related cited
papers.
- Do not refer to the selected papers by their citation keys, titles, or author
names. Instead, describe their shared scientific contribution in natural
language.

**CRITICAL:**
- `cite_key` must exactly match a key listed in the bibliography.
- NEVER invent, modify, shorten, normalize, paraphrase, or partially copy a
`cite_key`.
- Write as continuous natural language rather than keywords, concept lists,
paper
titles, or multiple questions.
- Prefer clear, direct sentences. Do not use em dashes or semicolons.

**Return a JSON object:**
{
  "threads": [
    {
      "thread_name": "exact thread name as given",
      "cite_keys": ["key 1", "key 2", ..., "key n"],
      "rationale": "A dense paragraph explaining how the selected papers
collectively contributed to the target paper from the perspective represented
by
this thread, what key scientific contributions they provided, how those
contributions influenced the target paper, and why these papers were selected
over other related cited papers within this perspective."
```

```
    }
  ]
}

Return ONLY the JSON object, nothing else.
```

Prompt for Reranking Retrieved Candidates

```
You are an expert in scientific literature analysis. Your task is to rank
candidate papers according to how well they satisfy the scientific relevance
criterion below.

{query_block}

Reference Scientific Contribution:
{reference_rationale}

Scientific Relevance Criterion:
- Use the reference scientific contribution above to infer the scientific
contribution criterion for this question.
- Rank candidates higher if their primary scientific contributions provide
ideas, insights, formulations, or methodologies that satisfy this criterion and
would meaningfully inform research on this question.
- Rank candidates lower if they are merely topically related, address a similar
problem, use similar methods, datasets, or benchmarks, or belong to the same
broad research area without satisfying the scientific contribution criterion
represented above.
- Evaluate candidates based on their scientific contributions rather than their
topical similarity.

Candidates:
{listing}

Rank all candidate papers from most to least likely to satisfy the scientific
relevance criterion above.

Return only valid JSON in the following format:
{"ranked": [every candidate number, ordered from most to least suitable]}
```